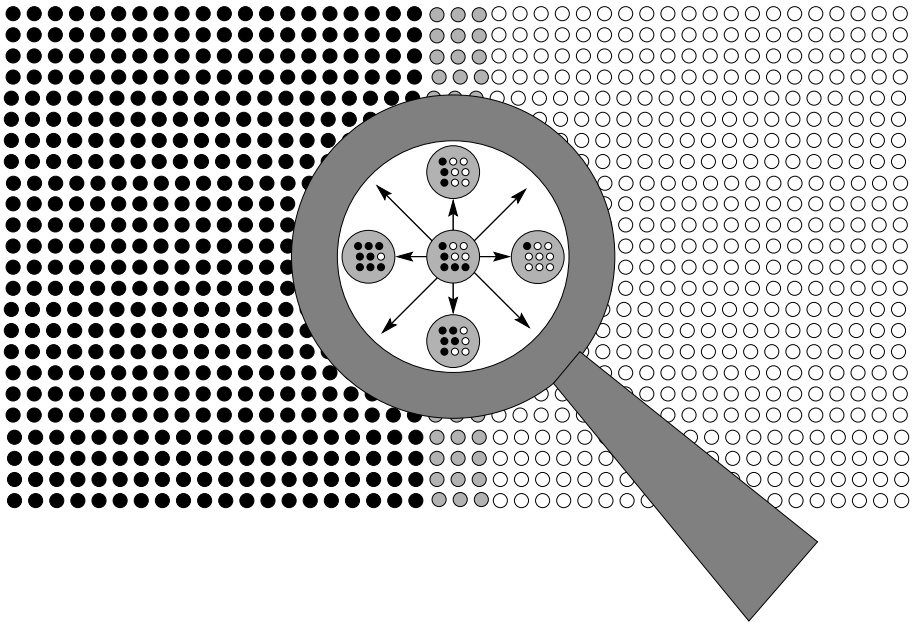
Dirk Kehrwald

Numerical Analysis of Immiscible Lattice BGK



Vom Fachbereich Mathematik der Universität Kaiserslautern zur Verleihung des akademischen Grades Doktor der Naturwissenschaften (Doctor rerum naturalium, Dr. rer. nat) genehmigte Dissertation

1. Gutachter: HDoz. Dr. Michael Junk
2. Gutachter: Dr. Li-Shi Luo

Datum der Disputation: 19. November 2002

D 386

Dirk Kehrwald
Abteilung Strömung in komplexen Strukturen
Fraunhofer-Institut für Techno- und Wirtschaftsmathematik
Postfach 3049
67653 Kaiserslautern
Germany

E-mail address: kehrwald@itwm.fraunhofer.de
URL: http://www.itwm.fraunhofer.de

2000 Mathematics Subject Classification. 35B25, 46F10, 53B50, 68U10, 68U20, 76D05, 76M20, 76M28, 76R50, 76T99.

Key words. Anisotropic diffusion, anti-diffusion, conservative volume tracking, consistency analysis, doctoral dissertation, diffusion, immiscible lattice BGK, interface tracking, lattice Boltzmann method, numerical analysis, PhD thesis, surface tension, two-phase flow, viscous transport schemes, weak formulation .

Abstract. The immiscible lattice BGK method for solving the two-phase incompressible Navier-Stokes equations is analysed in great detail. Equivalent moment analysis and local differential geometry are applied to examine how interface motion is determined and how surface tension effects can be included such that consistency to the two-phase incompressible Navier-Stokes equations can be expected. The results obtained from theoretical analysis are verified by numerical experiments. Since the intrinsic interface tracking scheme of immiscible lattice BGK is found to produce unsatisfactory results in two-dimensional simulations several approaches to improving it are discussed but all of them turn out to yield no substantial improvement. Furthermore, the intrinsic interface tracking scheme of immiscible lattice BGK is found to be closely connected to the well-known conservative volume tracking method. This result suggests to couple the conservative volume tracking method for determining interface motion with the Navier-Stokes solver of immiscible lattice BGK. Applied to simple flow fields, this coupled method yields much better results than plain immiscible lattice BGK.

Preface

This dissertation is the result of work carried out in the Mathematical Department of the University of Kaiserslautern between April 1999 and June 2002. During this time, my work was financially supported by the Deutsche Forschungsgemeinschaft (German Research Council) and I was allowed to use the resources of the Fraunhofer Institute for Industrial Mathematics (ITWM). All numerical experiments were implemented in Matlab or C on the Institute‘s Linux machines.

I want to take this opportunity to express my deep gratitude to HDoz. Dr. Michael Junk for his excellent supervision and his outstanding support during the preparation of this dissertation. Furthermore, I am very thankful to Dr. Irina Ginzburg, Dr. Li-Shi Luo, Prof. Dr. Dr. h. c. Helmut Neunzert, and Dr. Konrad Steiner for many helpful hints and fruitful discussions. I also want to thank Dr. Doris Reinel-Bitzer for making her lattice BGK code available to me, as well as the referees, HDoz. Dr. Michael Junk and Dr. Li-Shi Luo, for drawing my attention to a number of typographical errors in the original manuscript.

Most gratitude I owe to my girl-friend Dagmar Ernst and my family. Without their great love and their extensive support in all spheres of life, I had not been able to finish this thesis.

Kaiserslautern, 25th November 2002

Dirk Kehrwald

Contents

Contents . . . 5

Introduction . . . 7

Chapter 1. The lattice BGK equation . . . 9

1.1. Overview of the lattice BGK equation . . . 9
1.1.1. Description of the method . . . 9
1.1.2. Consistency analysis . . . 11
1.1.3. Alternative formulation . . . 14

1.2. Experimental investigation of moments . . . 14

Chapter 2. Modelling of two-phase flow . . . 21

2.1. Mathematical theory of incompressible two-phase flow . . . 21
2.1.1. Basic assumptions . . . 21
2.1.2. Derivation of the full-space equations . . . 22
2.1.3. Divergence form of the surface tension term . . . 25

2.2. Immiscible lattice BGK . . . 29
2.2.1. Original method . . . 29
2.2.2. Reformulation . . . 31

2.3. Interface tracking in immiscible lattice BGK . . . 35
2.3.1. Equivalent moment analysis . . . 35
2.3.2. Isolation . . . 39
2.3.3. Numerical investigation . . . 40

2.4. Modelling surface tension in immiscible lattice BGK . . . 41
2.4.1. Theoretical background . . . 42
2.4.2. Consistent modelling of surface tension . . . 47
2.4.3. Bubble test . . . 49

Chapter 3. Some numerical experiments . . . 53

3.1. A two-dimensional benchmark . . . 53

3.2. Plain ILBGK . . . 54

3.3. Anisotropic diffusion . . . 56
3.3.1. Theoretical Background . . . 57
3.3.2. Numerical algorithm . . . 61
3.3.3. Anisotropic diffusion instead of recolouring . . . 62
3.3.4. Anisotropic diffusion after recolouring . . . 64

3.4. Artificial surface tension . . . 66

Chapter 4. Numerical analysis of the viscous transport scheme . . . 77

4.1. Generalisation of the viscous transport scheme . . . 77

4.2. Stability of the general viscous transport scheme . . . 80

4.3. Numerical experiments with improved ILBGK interface tracking . . . 83

Chapter 5. Volume tracking lattice BGK 91
5.1. Conservative volume tracking 91
5.1.1. Interface reconstruction 91
5.1.2. Volume tracking 92
5.1.3. Numerical experiments with conservative volume tracking 95
5.2. ILBGK interface tracking versus conservative volume tracking 95
5.3. Combination of conservative volume tracking and lattice BGK 98

Conclusions 105

Appendix. Example of a smooth indicator function 107

Bibliography 113

Introduction

Lattice Boltzmann methods are relatively new and quite popular numerical methods for solving the incompressible Navier-Stokes equations of mathematical fluid dynamics in a very special way. Unlike conventional Navier-Stokes solvers, they do not approximate the equations directly but simulate fluid behaviour on a mesoscopic level and determine the solution of the incompressible Navier-Stokes equations by computing certain moments of the particle density. The lattice BGK equation is a particularly simple lattice Boltzmann method. Lattice Boltzmann has been developed as an advancement of the lattice gas method which models the motion of microscopic fluid particles in a very simple way. A detailed historic review is given at the beginning of Section 1.1.

The main advantage of lattice Boltzmann methods is the simplicity of the underlying mesoscopic equations in comparison to Navier-Stokes. Thus, it is natural to believe in lattice Boltzmann to outperform conventional methods of computational fluid dynamics in the simulation of complicated flow, such as flow in porous media or two-phase flow. In fact, there is a huge number of papers on lattice Boltzmann methods for two-phase flow, a small extract is listed at the beginning of Section 2.2. However, so far, rigorous numerical analysis of lattice Boltzmann methods for two-phase flow has hardly been performed. Nevertheless, such kind of analysis is a very reliable tool for understanding the behaviour of a given numerical scheme, so it is surely a good idea to apply it to lattice Boltzmann methods for two-phase flow as well.

Keeping to this idea we shall proceed as follows. In Chapter 1, we will introduce the lattice BGK method for simulating single-phase flow and show its consistency to the incompressible Navier-Stokes equations by so-called equivalent moment analysis. The validity of this analysis will be underlined by numerical experiments. Chapter 2 starts with the presentation of mathematical theory of two-phase flow. After this, we will introduce the immiscible lattice BGK method which is one of several lattice BGK schemes for two-phase flow. We will perform a detailed analysis of this method with particular focus on interface tracking and the treatment of surface tension effects. Additionally, we will isolate the intrinsic interface tracking method of immiscible lattice BGK and consider it as a stand-alone transport scheme. In Chapter 3, we will define a benchmark problem for transport schemes and then present results computed with ILBGK interface tracking. Those results will turn out to be quite unsatisfactory and thus, we will also discuss several ideas how the scheme could possibly be improved. Unfortunately, none of those ideas is to provide the desired result. Chapter 4 contains a detailed numerical analysis of ILBGK interface tracking. We will learn that this scheme is a special case of a parameter-dependent general method and consider numerical results for some reasonable choices of parameters. However, a careful choice of parameters does not generally improve the scheme. Therefore, a conservative volume tracking method will be presented in Chapter 5 as an alternative to the intrinsic interface tracking scheme of immiscible lattice BGK. Numerical results will be presented for conservative volume tracking as a stand-alone interface tracking scheme and in combination with the Navier-Stokes solver of immiscible lattice BGK. After this, we will draw some conclusions from the work we performed and finally, the Appendix will provide an interesting supplement to the theoretical studies in Chapter 2.

Let us now conclude this introduction with a few remarks on notation. In the following, we will print matrices and vectors in boldface and their respective components, as well as other scalars, in normalface, for example $A_{\alpha\beta}$ denotes the entry in the α-th line and β-th column of the matrix $\boldsymbol{A}$ and x_α stands for the α-th component of the vector $\boldsymbol{x}$. Especially, $\mathbf{0} = (0, \ldots, 0)^T$. Subscripts denoting components are separated from other lower indices by a semicolon, for example $c_{j;\alpha}$ represents the α-th component of the vector $\boldsymbol{c}_j$. Summation is implicitly assumed over repeated lowercase Greek but not uppercase Greek or Latin indices. Furthermore, we will use shortcuts of the form ∂_t for the derivative with respect to the scalar variable t and ∂_α for the derivative with respect to the α-th component of the position vector.

CHAPTER 1

The lattice BGK equation

This chapter introduces the lattice BGK method for solving the incompressible Navier-Stokes equations. The lattice BGK equation belongs to the family of lattice gas/lattice Boltzmann methods, see for example Benzi et al. [**3**], Chen and Doolen [**10**], Rothman and Zaleski [**67**], or Wolf-Gladrow [**81**] for a detailed overview of such methods. The main feature of lattice gas/lattice Boltzmann methods is that they do not approximate the Navier-Stokes equations directly. Instead, they simulate particle dynamics on a microscopic (lattice gas) respectively mesoscopic (lattice Boltzmann) scale and the solution of the macroscopic Navier-Stokes equations is obtained via averaging. In particular, the lattice BGK equation is a very simple lattice Boltzmann method.

Section 1.1 contains a short description of the lattice BGK method and a proof of its consistency to the incompressible Navier-Stokes equations. In Section 1.2, one of the basic prerequisites of the consistency proof, namely the assumption on the order of magnitude of some particular moments of the mesoscopic particle distribution function, is justified by numerical experiments.

1.1. Overview of the lattice BGK equation

In 1986, Frisch et al. [**17**] introduced the lattice gas method for the incompressible Navier-Stokes equations in two-dimensional space and in the same year d'Humières et al. [**38**] extended the method to three space dimensions. The lattice Boltzmann method was developed in 1988 by McNamara and Zanetti [**53**] in order to overcome an essential drawback of lattice gas, namely intrinsic statistical noise, and in 1989, Higuera and Jiménez [**33**] introduced a linearised version of lattice Boltzmann. In 1992, Chen et al. [**9**] and Qian et al. [**63**] independently proposed to apply the ideas of Bhatnagar et al. [**4**] to the lattice Boltzmann method, thus inventing the lattice BGK (LBGK) method. In 1997, Abe [**1**] as well as He and Luo [**31**] independently proved that the governing equations of LBGK can be derived directly from the continuous Boltzmann equation with BGK type collision operator and in 1999, Junk [**40**] showed the formal connection between the lattice Boltzmann equation and other kinetic schemes. It should also be noted that Luo [**51, 52**] derived a similar set of equations from the continuous Enskog equation and proved consistency of this model to the incompressible Navier-Stokes equations.

In the following, we will discuss the lattice BGK method (Section 1.1.1) and prove its consistency to the incompressible Navier-Stokes equations (Section 1.1.2).

1.1.1. Description of the method. In LBGK, we assume that a domain $\Omega \subset \mathbb{R}^d$ is covered by a regular lattice with equidistant nodes and we denote the set of lattice nodes with $\mathcal{X}$. At a given time t, we consider the particle density $f(t, \boldsymbol{x}; \boldsymbol{v})$ of fluid particles located at position $\boldsymbol{x} \in \mathcal{X}$ and moving with velocity $\boldsymbol{v} \in \mathcal{V}_q$, where $\mathcal{V}_q = \{\boldsymbol{c}_j : j = 0, \ldots, q-1\}$ with some $q \in \mathbb{N}$ contains only a finite number of velocity vectors. Given two functions $g_1, g_2 : \mathcal{V}_q \to \mathbb{R}$, we introduce the scalar product

$$\langle g_1, g_2 \rangle_{\boldsymbol{v}} = \sum_{\boldsymbol{v}} g_1(\boldsymbol{v}) g_2(\boldsymbol{v}).$$

Defining now a discrete moment M of order $m \in \mathbb{N}$ of f by

$$M(t, \boldsymbol{x}) = \langle f(t, \boldsymbol{x}; \boldsymbol{v}), P(\boldsymbol{v}) \rangle_{\boldsymbol{v}},$$

where P is a polynomial of degree m in $\boldsymbol{v}$, we can write mass density ρ as a zeroth order moment of f and momentum density $\rho \boldsymbol{u}$ as a first order moment of f, in particular

$$\rho(t, \boldsymbol{x}) = \langle f(t, \boldsymbol{x}; \boldsymbol{v}), 1 \rangle_{\boldsymbol{v}} \qquad \text{and} \qquad (\rho u_\alpha)(t, \boldsymbol{x}) = \langle f(t, \boldsymbol{x}; \boldsymbol{v}), v_\alpha \rangle_{\boldsymbol{v}}. \tag{1.1}$$

Flow velocity $\boldsymbol{u}$ is then given by $u_\alpha = \rho u_\alpha / \rho$.

In this thesis, we will exclusively consider the D2Q9 [**63**] LBGK model which is an LBGK model in two space dimensions (D2) with a square lattice and nine discrete velocities (Q9). In lattice units, the

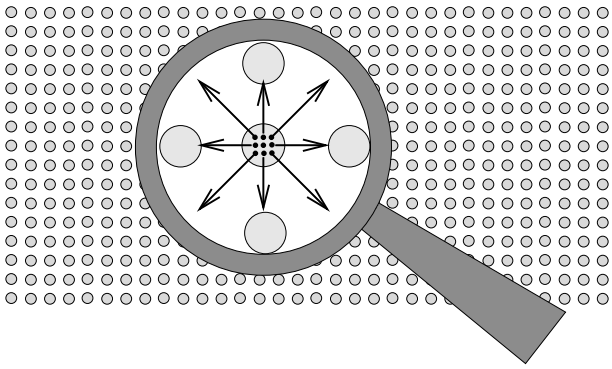


Figure 1.1: The D2Q9 LBGK model: Each grey circle marks a node of the square lattice. Under the magnifier, we can see the nine groups of particles (marked with black dots) and the corresponding particle velocities (pictured as arrows). This Figure is taken from Junk [**43**].

velocity vectors are given by

$$\boldsymbol{c}_0 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \qquad \boldsymbol{c}_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \qquad \boldsymbol{c}_2 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \qquad \boldsymbol{c}_3 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \qquad \boldsymbol{c}_4 = \begin{pmatrix} -1 \\ 1 \end{pmatrix}, \qquad \boldsymbol{c}_5 = \begin{pmatrix} -1 \\ 0 \end{pmatrix},$$

$$\boldsymbol{c}_6 = \begin{pmatrix} -1 \\ -1 \end{pmatrix}, \qquad \boldsymbol{c}_7 = \begin{pmatrix} 0 \\ -1 \end{pmatrix}, \qquad \boldsymbol{c}_8 = \begin{pmatrix} 1 \\ -1 \end{pmatrix}.$$

Note that non-zero particle velocities are chosen such that particles are moving from a given node to one of its next neighbours in lattice direction ($\boldsymbol{c}_1$, $\boldsymbol{c}_3$, $\boldsymbol{c}_5$, $\boldsymbol{c}_7$) or diagonal direction ($\boldsymbol{c}_2$, $\boldsymbol{c}_4$, $\boldsymbol{c}_6$, $\boldsymbol{c}_8$), as illustrated in Figure 1.1. Figure 1.2 visualises a typical particle density at a given lattice node. In the D2Q9 LBGK model, the lattice BGK equation for f has the form

$$f(t+\delta t, \boldsymbol{x}; \boldsymbol{v}) = (1-\omega) f(t, \boldsymbol{x} - \boldsymbol{v}\delta t; \boldsymbol{v}) + \omega f^{eq}(\rho(t, \boldsymbol{x} - \boldsymbol{v}\delta t), \boldsymbol{u}(t, \boldsymbol{x} - \boldsymbol{v}\delta t); \boldsymbol{v}), \tag{1.2}$$

for $\boldsymbol{v} = \boldsymbol{c}_0, \ldots, \boldsymbol{c}_8$, where δt denotes the time step size and $\omega \in [1, 2)$ plays the role of a relaxation parameter. Note that for the side length δx of the lattice, we have

$$\delta x = \|\boldsymbol{c}_1 \delta t\|_2 = \delta t. \tag{1.3}$$

The equilibrium particle density f^{eq} is given by

$$f^{eq}(\rho, \boldsymbol{u}; \boldsymbol{v}) = f^*(\boldsymbol{v}) \left(\rho + 3\rho u_\alpha v_\alpha + \frac{9}{2} \rho u_\alpha u_\beta \left(v_\alpha v_\beta - \frac{\delta_{\alpha\beta}}{3} \right) \right), \tag{1.4}$$

where

$$f^*(\boldsymbol{c}_j) = \begin{cases} 4/9 & \text{for } j = 0 \\ 1/9 & \text{for } j = 1, 3, 5, 7 \\ 1/36 & \text{for } j = 2, 4, 6, 8 \end{cases} \qquad \text{and} \qquad \delta_{\alpha_\beta} = \begin{cases} 1 & \text{for } \alpha = \beta \\ 0 & \text{otherwise.} \end{cases}$$

By straight calculation, we obtain

$$\langle f^{eq}, 1 \rangle_{\boldsymbol{v}} = \rho \qquad \text{and} \qquad \langle f^{eq}, v_\alpha \rangle_{\boldsymbol{v}} = \rho u_\alpha. \tag{1.5}$$

Equation (1.2) is typically solved in two steps, the collision step

$$\tilde{f}(t, \boldsymbol{x}; \boldsymbol{v}) = (1-\omega) f(t, \boldsymbol{x}; \boldsymbol{v}) + \omega f^{eq}(\rho(t, \boldsymbol{x}), \boldsymbol{u}(t, \boldsymbol{x}); \boldsymbol{v}) \tag{1.6}$$

and the propagation step

$$f(t+\delta t, \boldsymbol{x}; \boldsymbol{v}) = \tilde{f}(t, \boldsymbol{x} - \boldsymbol{v}\delta t; \boldsymbol{v}). \tag{1.7}$$

Combining (1.1) and (1.5) we find that conservation of both mass density ρ and momentum density ρu_α is guaranteed in the collision step. Since it is not possible to create or destroy mass or momentum by pure propagation we conclude that also the full LBGK equation (1.2) conserves mass density as well as momentum density. The complete LBGK method is summarised in Algorithm 1.1.

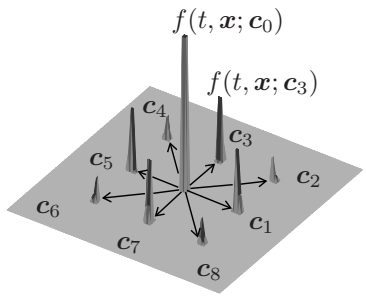


Figure 1.2: The relative magnitude of particle densities when the fluid is in equilibrium: The particle density corresponding to zero velocity is largest, the common magnitude of particle densities corresponding to velocities in lattice directions is larger than that of particle densities corresponding to velocities in diagonal direction. This Figure is taken from Junk [**43**].

```
initialise ρ(0, x) and u(0, x);
let f(0, x; v) = f^eq(ρ(0, x), u(0, x)) as defined in (1.4);
while t < t_max
   determine f^eq(ρ(t, x), u(t, x)) according to (1.4);
   do collision according to (1.6);
   do propagation according to (1.7);
   let t = t + δt;
   compute ρ(t, x) and ρu_α(t, x) according to (1.1);
   let u_α(t, x) = ρu_α(t, x)/ρ(t, x);
```

Algorithm 1.1: The lattice BGK method

1.1.2. Consistency analysis. We shall now follow the equivalent moment approach proposed by Junk [**42, 43**] to show consistency of the scheme (1.2) to the incompressible Navier-Stokes equations [**15, 49, 50, 80**]

$$\partial_\alpha u_\alpha = 0 \qquad \text{and} \qquad \partial_t(\varrho u_\alpha) + \partial_\beta(\varrho u_\alpha u_\beta) = \partial_\beta \tau_{\alpha\beta}. \tag{1.8}$$

In (1.8), ϱ denotes the constant density of the incompressible fluid while

$$\tau_{\alpha\beta} = -p\delta_{\alpha\beta} + \mu\left(\partial_\alpha u_\beta + \partial_\beta u_\alpha\right) \tag{1.9}$$

represents the stress tensor with dynamic pressure p and dynamic viscosity μ. Note that incompressible one-phase flow does not depend on density and in fact, the second equation in (1.8) can be reformulated such that ϱ does no longer appear. However, we will stick to the formulation above in order to maintain notational analogy to the two-phase case considered in Section 2.1.

Equivalent moment analysis makes sense only under the condition that the lattice distance δx is small compared to the characteristic length L of the flow, i. e.

$$\Delta x = \frac{\delta x}{L} \ll 1,$$

because otherwise, space resolution is too coarse for (1.2) to produce reasonable results. Reider and Sterling [**64**] showed that for a consistent approximation to incompressible flow, the characteristic speed U of the flow must fulfill $U = \mathcal{O}(\Delta x)$. For simplicity, we set the proportionality to one, i. e. we choose $U = \Delta x$. Introducing the characteristic time scale $T = L/U$ and using (1.3), we find that

$$\Delta t = \frac{\delta t}{T} = \frac{\delta x}{L} U = \Delta x^2.$$

The relation $\Delta t = \Delta x^2$ will be frequently used in the following. Introducing additional scaled quantities

$$\hat{x}_\alpha = \frac{x_\alpha}{L}, \qquad \hat{t} = \frac{t}{T}, \qquad \hat{f}\left(\hat{t}, \hat{\boldsymbol{x}}; \boldsymbol{v}\right) = f(t, \boldsymbol{x}; \boldsymbol{v}), \qquad \hat{\rho}\left(\hat{t}, \hat{\boldsymbol{x}}\right) = \rho(t, \boldsymbol{x}), \qquad \hat{u}_\alpha\left(\hat{t}, \hat{\boldsymbol{x}}\right) = \frac{1}{U} u_\alpha(t, \boldsymbol{x}),$$

and

$$\hat{f}^{eq}\left(\hat{\rho}, \hat{\boldsymbol{u}}; \boldsymbol{v}\right) = f^*(\boldsymbol{v}) \left(\hat{\rho} + 3\Delta x \hat{\rho} \hat{u}_\alpha v_\alpha + \frac{9\Delta x^2}{2} \hat{\rho} \hat{u}_\alpha \hat{u}_\beta \left(v_\alpha v_\beta - \frac{\delta_{\alpha\beta}}{3}\right)\right),$$

we can write the LBGK equation (1.2) in in terms of scaled quantities,

$$\hat{f}\left(\hat{t}+\Delta t,\hat{\boldsymbol{x}};\boldsymbol{v}\right)=(1-\omega)\,\hat{f}\left(\hat{t},\hat{\boldsymbol{x}}-\boldsymbol{v}\Delta x;\boldsymbol{v}\right)+\omega\hat{f}^{eq}\left(\hat{\rho}\left(\hat{t},\hat{\boldsymbol{x}}-\boldsymbol{v}\Delta x\right),\hat{\boldsymbol{u}}\left(\hat{t},\hat{\boldsymbol{x}}-\boldsymbol{v}\Delta x\right);\boldsymbol{v}\right).$$

In the following, we will exclusively work in the described scaling, so from now on we allow ourselves to skip the hat-superscripts. Therefore, the equation above takes the form

$$f(t+\Delta t,\boldsymbol{x};\boldsymbol{v})=(1-\omega)f(t,\boldsymbol{x}-\boldsymbol{v}\Delta x;\boldsymbol{v})+\omega f^{eq}(\rho(t,\boldsymbol{x}-\boldsymbol{v}\Delta x),\boldsymbol{u}(t,\boldsymbol{x}-\boldsymbol{v}\Delta x);\boldsymbol{v}) \tag{1.10}$$

with

$$f^{eq}(\rho,\boldsymbol{u};\boldsymbol{v})=f^{*}(\boldsymbol{v})\left(\rho+3\Delta x\rho u_\alpha v_\alpha+\frac{9\Delta x^2}{2}\rho u_\alpha u_\beta\left(v_\alpha v_\beta-\frac{\delta_{\alpha\beta}}{3}\right)\right). \tag{1.11}$$

Our aim is now to transform the LBGK equation (1.10) into a system of moment equations that contains mass and momentum equations as a subsystem. Similar reformulations of lattice Boltzmann equations can be found in d'Humières [**36**], d'Humières et al. [**37**], Klar [**45**], as well as Lallemand and Luo [**48**]. Here, we choose the scaled polynomials

$$Q_0(\boldsymbol{v})=1,\qquad Q_1(\boldsymbol{v})=\frac{v_1}{\Delta x},\qquad Q_2(\boldsymbol{v})=\frac{v_2}{\Delta x}, \tag{1.12a}$$

$$Q_3(\boldsymbol{v})=\frac{1}{\Delta x^2}\left(v_1^2-\frac{1}{3}\right),\qquad Q_4(\boldsymbol{v})=\frac{v_1v_2}{\Delta x^2},\qquad Q_5(\boldsymbol{v})=\frac{1}{\Delta x^2}\left(v_2^2-\frac{1}{3}\right), \tag{1.12b}$$

$$Q_6(\boldsymbol{v})=\frac{1}{\Delta x^3}\left(3\|\boldsymbol{v}\|_2^2-4\right)v_1,\qquad Q_7(\boldsymbol{v})=\frac{1}{\Delta x^3}\left(3\|\boldsymbol{v}\|_2^2-4\right)v_2, \tag{1.12c}$$

and

$$Q_8(\boldsymbol{v})=\frac{1}{\Delta x^4}\left(9\|\boldsymbol{v}\|_2^4-15\|\boldsymbol{v}\|_2^2+2\right) \tag{1.12d}$$

to produce the moments

$$M_k=\langle f,Q_k\rangle_{\boldsymbol{v}}\qquad\text{and}\qquad M_k^{eq}=\langle f^{eq},Q_k\rangle_{\boldsymbol{v}}, \tag{1.13}$$

for $k=0,\dots,8$. Note that due to (1.12a), we obtain from (1.13)

$$M_0=M_0^{eq}=\rho,\qquad M_1=M_1^{eq}=\rho u_1,\qquad\text{and}\qquad M_2=M_2^{eq}=\rho u_2.$$

Furthermore, we will often use the notation

$$\boldsymbol{\vartheta}=\begin{pmatrix}M_3 & M_4\\ M_4 & M_5\end{pmatrix}.$$

The chosen scaling of the polynomials (1.12) will be justified in Section 1.2.

We find that the mapping $f\mapsto\boldsymbol{Q}f=(M_0,\dots,M_8)^T$ respectively $f^{eq}\mapsto\boldsymbol{Q}f^{eq}=(M_0^{eq},\dots,M_8^{eq})^T$ is linear and invertible, so we can write

$$f=\sum_{k=0}^{8}\frac{Q_k}{\langle f^*,Q_k^2\rangle_{\boldsymbol{v}}}M_kf^*\qquad\text{respectively}\qquad f^{eq}=\sum_{k=0}^{8}\frac{Q_k}{\langle f^*,Q_k^2\rangle_{\boldsymbol{v}}}M_k^{eq}f^*$$

which yields

$$\begin{aligned}(1-\omega)&f(t,\boldsymbol{x}-\boldsymbol{v}\Delta x;\boldsymbol{v})+\omega f^{eq}(\rho(t,\boldsymbol{x}-\boldsymbol{v}\Delta x),\boldsymbol{u}(t,\boldsymbol{x}-\boldsymbol{v}\Delta x);\boldsymbol{v})\\ &=\sum_{k=0}^{8}\frac{Q_k(\boldsymbol{v})}{\langle f^*,Q_k^2\rangle_{\boldsymbol{v}}}\Big((1-\omega)M_k+\omega M_k^{eq}\Big)(t,\boldsymbol{x}-\boldsymbol{v}\Delta x)f^*(\boldsymbol{v}).\end{aligned}$$

Inserting this into (1.10) and applying $\boldsymbol{Q}$ to the resulting equation we obtain the discrete equivalent moment system of (1.10) which reads

$$M_k(t+\Delta t,\boldsymbol{x})=\sum_{j=0}^{8}\frac{1}{\langle f^*,Q_j^2\rangle_{\boldsymbol{v}}}\Big\langle f^*,Q_kQ_j\Big((1-\omega)M_j+\omega M_j^{eq}\Big)(t,\boldsymbol{x}-\boldsymbol{v}\Delta x)\Big\rangle_{\boldsymbol{v}}. \tag{1.14}$$

Since the summands on the right hand side of (1.14) are expressions of the form $\langle f^*, Pg(\boldsymbol{x}-\boldsymbol{v}\Delta x)\rangle_{\boldsymbol{v}}$, where P is a polynomial in $\boldsymbol{v}$ and g is a smooth function in $\boldsymbol{x}$, we now have a closer look at structures of this form. By definition, we have

$$\langle f^*, Pg(\boldsymbol{x}-\boldsymbol{v}\Delta x)\rangle_{\boldsymbol{v}} = \sum_{k=0}^{8} f^*(\boldsymbol{c}_k)P(\boldsymbol{c}_k)g(\boldsymbol{x}-\boldsymbol{c}_k\Delta x) \tag{1.15}$$

which can be interpreted as a finite difference expression. Usually, the coefficients $P(\boldsymbol{v})f^*(\boldsymbol{v})$ of such an expression are collected in a so-called stencil defined by

$$\begin{bmatrix} P(\boldsymbol{c}_8)f^*(\boldsymbol{c}_8) & P(\boldsymbol{c}_7)f^*(\boldsymbol{c}_7) & P(\boldsymbol{c}_6)f^*(\boldsymbol{c}_6) \\ P(\boldsymbol{c}_1)f^*(\boldsymbol{c}_1) & P(\boldsymbol{c}_0)f^*(\boldsymbol{c}_0) & P(\boldsymbol{c}_5)f^*(\boldsymbol{c}_5) \\ P(\boldsymbol{c}_2)f^*(\boldsymbol{c}_2) & P(\boldsymbol{c}_3)f^*(\boldsymbol{c}_3) & P(\boldsymbol{c}_4)f^*(\boldsymbol{c}_4) \end{bmatrix} g(\boldsymbol{x}) = \sum_{k=0}^{8} f^*(\boldsymbol{c}_k)P(\boldsymbol{c}_k)g(\boldsymbol{x}-\boldsymbol{c}_k\Delta x).$$

Applying Taylor expansion to check the consistency of all stencils appearing in (1.14) we find the discrete equivalent moment system (1.14) to be a finite difference approximation to a system of partial differential equations consisting of

$$\partial_t\rho + \partial_\alpha(\rho u_\alpha) = \frac{1}{6}\partial_{\alpha\alpha}\rho + \mathcal{O}\left(\Delta x^2\right), \tag{1.16a}$$

$$\partial_t(\rho u_\alpha) + \partial_\beta\Big((1-\omega)\vartheta_{\alpha\beta} + \omega\rho u_\alpha u_\beta\Big) + \frac{1}{3\Delta x^2}\partial_\alpha\rho = \frac{1}{6}\Big(\partial_{\beta\beta}(\rho u_\alpha) + 2\partial_{\alpha\beta}(\rho u_\beta)\Big) - \frac{1}{18}\partial_{\alpha\beta\beta}\rho + \mathcal{O}\left(\Delta x^2\right), \tag{1.16b}$$

$$\vartheta_{\alpha\beta} = \rho u_\alpha u_\beta - \frac{1}{3\omega}\left(\partial_\beta(\rho u_\alpha) + \partial_\alpha(\rho u_\beta) - \frac{1}{3}\partial_{\alpha\beta}\rho\right) + \mathcal{O}\left(\Delta x^2\right), \tag{1.16c}$$

and three other equations (see details in Junk [**42, 43**]). Since the remaining equations decouple from (1.16) and are not needed to show consistency of (1.14) to the incompressible Navier-Stokes equations (1.8), we will simply ignore them.

The number of equations in (1.16) can be further reduced by inserting (1.16c) into (1.16b) which yields

$$\begin{aligned}\partial_t(\rho u_\alpha) + \partial_\beta(\rho u_\alpha u_\beta) + \frac{1}{3\Delta x^2}\partial_\alpha\rho \\ = \frac{1}{3}\left(\frac{1}{\omega}-\frac{1}{2}\right)\partial_\beta\Big(\partial_\beta(\rho u_\alpha) + \partial_\alpha(\rho u_\beta)\Big) + \frac{1}{3}\partial_{\alpha\beta}\rho u_\beta - \frac{1}{18}\partial_{\alpha\beta\beta}\rho + \mathcal{O}\left(\Delta x^2\right).\end{aligned}$$

Requiring that all terms in this equation are of $\mathcal{O}(1)$ we find that we need $\partial_\alpha\rho$ to be of $\mathcal{O}\left(\Delta x^2\right)$ which justifies the assumption

$$\rho(t,\boldsymbol{x}) = \varrho + 3\varrho\Delta x^2 p(t,\boldsymbol{x}), \tag{1.17}$$

where ϱ is a constant and p is an order one function. We can now rewrite (1.16) in the form

$$\partial_\alpha u_\alpha = \mathcal{O}\left(\Delta x^2\right) \qquad \text{and} \qquad \partial_t(\varrho u_\alpha) + \partial_\beta(\varrho u_\alpha u_\beta) + \partial_\alpha p = \frac{\varrho}{3}\left(\frac{1}{\omega}-\frac{1}{2}\right)\partial_\beta(\partial_\beta u_\alpha + \partial_\alpha u_\beta) + \mathcal{O}\left(\Delta x^2\right).$$

Now we define the dynamic viscosity μ by

$$\mu = \frac{\varrho}{3}\left(\frac{1}{\omega}-\frac{1}{2}\right) \tag{1.18}$$

and apply (1.9), so we finally obtain

$$\partial_\alpha u_\alpha = \mathcal{O}\left(\Delta x^2\right) \qquad \text{and} \qquad \partial_t(\varrho u_\alpha) + \partial_\beta(\varrho u_\alpha u_\beta) = \partial_\beta\tau_{\alpha\beta} + \mathcal{O}\left(\Delta x^2\right). \tag{1.19}$$

Up to $\mathcal{O}\left(\Delta x^2\right)$, those are in fact the incompressible Navier-Stokes equations (1.8) with constant density ϱ and dynamic pressure p. This yields

Proposition 1.1. *The discrete equivalent moment system* (1.14) *of the lattice BGK equation* (1.10) *is an approximation of* $\mathcal{O}\left(\Delta x^2\right)$ *to the incompressible Navier-Stokes equations* (1.8).

1.1.3. Alternative formulation. He and Luo [**30**] proposed to exploit the pressure law (1.17) from the beginning by using the equilibrium distribution

$$f_{HL}^{eq}(\rho, \boldsymbol{u}; \boldsymbol{v}) = f^*(\boldsymbol{v}) \left(\rho + 3\Delta x \varrho u_\alpha v_\alpha + \frac{9\Delta x^2}{2} \varrho u_\alpha u_\beta \left(v_\alpha v_\beta - \frac{\delta_{\alpha\beta}}{3} \right) \right)$$

instead of (1.11). This implies that in place of (1.5) we have

$$\langle f_{HL}^{eq}, 1 \rangle_{\boldsymbol{v}} = \rho \qquad \text{and} \qquad \langle f_{HL}^{eq}, v_\alpha \rangle_{\boldsymbol{v}} = \varrho u_\alpha,$$

so to guarantee conservation of mass and momentum we must replace (1.1) with

$$\langle f, 1 \rangle_{\boldsymbol{v}} = \rho \qquad \text{and} \qquad \langle f, v_\alpha \rangle_{\boldsymbol{v}} = \varrho u_\alpha.$$

Performing then a moment analysis completely analogous to the one presented in Section 1.1.2 we obtain

$$\partial_\alpha u_\alpha = \mathcal{O}\left(\Delta x^2\right),$$

$$\partial_t(\varrho u_\alpha) + \partial_\beta\Big((1-\omega)\vartheta_{\alpha\beta} + \omega \varrho u_\alpha u_\beta\Big) + \partial_\alpha p = \frac{1}{6}\Big(\partial_{\beta\beta}(\varrho u_\alpha) + 2\partial_{\alpha\beta}(\varrho u_\beta)\Big) + \mathcal{O}\left(\Delta x^2\right),$$

and

$$\vartheta_{\alpha\beta} = \varrho u_\alpha u_\beta - \frac{1}{3\omega}\Big(\partial_\beta(\varrho u_\alpha) + \partial_\alpha(\varrho u_\beta)\Big) + \mathcal{O}\left(\Delta x^2\right)$$

instead of (1.16). Continuing the analysis, the above system of equations can be easily reduced to (1.19) if definitions (1.18) and (1.9) are applied. Thus, we can formulate

Proposition 1.2. *The discrete equivalent moment system* (1.14) *of the LBGK equation* (1.10) *with equilibrium distribution* f_{HL}^{eq} *instead of* f^{eq} *produces an approximation of* $\mathcal{O}\left(\Delta x^2\right)$ *to the incompressible Navier-Stokes equations* (1.8).

1.2. Experimental investigation of moments

The scaling of the polynomials (1.12) is chosen in order to produce moments M_k and M_k^{eq} of $\mathcal{O}(1)$. Since f^{eq} is given explicitly we directly calculate

$$M_0^{eq} = \rho, \qquad M_1^{eq} = \rho u_1, \qquad M_2^{eq} = \rho u_2, \qquad M_3^{eq} = \rho u_1^2, \qquad M_4^{eq} = \rho u_1 u_2, \qquad M_5^{eq} = \rho u_2^2,$$

$$M_6^{eq} = 0, \qquad M_7^{eq} = 0, \qquad \text{and} \qquad M_8^{eq} = 0.$$

The equilibrium moments M_0^{eq} to M_5^{eq} are obviously $\mathcal{O}(1)$, so we expect the moments M_0 to M_5 to be $\mathcal{O}(1)$ as well, see Junk et al. [**44**] for a detailed discussion of the ideas involved. However, we cannot prove that this expectation is fulfilled and, even worse, for $k = 6, 7, 8$ we cannot even guess the scaling of M_k from what we know about M_k^{eq}. Therefore, we want at least to experimentally investigate the moments appearing in some slightly non-trivial situation, namely in laminar flow around a circular cylinder [**82**].

We consider a two-dimensional channel with length $27.5L$ and height L, where L is the characteristic length of the flow, bounded by solid walls at top and bottom. Let this channel contain a solid circular cylinder with radius $L/4$ whose centre point is located at a distance of $L/2$ from both bottom and inflow of the channel. At each fluid-solid interface, we impose a no-slip condition by applying the bounce-back rule

$$f(t + \Delta t, \boldsymbol{x}; -\boldsymbol{c}_k) = f(t, \boldsymbol{x}; \boldsymbol{c}_k) \tag{1.20}$$

if $\boldsymbol{x} + \boldsymbol{c}_k \Delta x$ is is located outside the fluid domain. The bounce-back rule (1.20) sends each particle distribution streaming towards a boundary node back to the direction it came from. Bounce-back is known to be not very accurate at curved boundaries if certain care is not taken, however, here it is used because it is the most common boundary condition for the lattice BGK equation. Even though boundary conditions are not the major concern here, it should be noted that in the given situation, bounce-back can be considered a boundary condition either for the (smooth) cylinder itself or for a zig-zag approximation of it, being more accurate in the second case than in the first one. For a more detailed discussion of the treatment of no-slip boundary conditions in LBGK consult for example Ginzbourg and Adler [**19**], Ginzbourg and d'Humières [**20**], He et al. [**32**], or Mei et al. [**54, 55**]. At the inflow boundary, we apply a parabolic velocity profile whose maximum value equals the characteristic speed U of the flow.

This flow is now simulated on three different lattices with three different values of U and L, namely on a 220×40 lattice with $U = 0.1$ and $L = 2.5$, on a 440×80 lattice with $U = 0.05$ and $L = 5$, and on a 880×160 lattice with $U = 0.025$ and $L = 10$. We assume that the fluid has constant density $\varrho = 1$

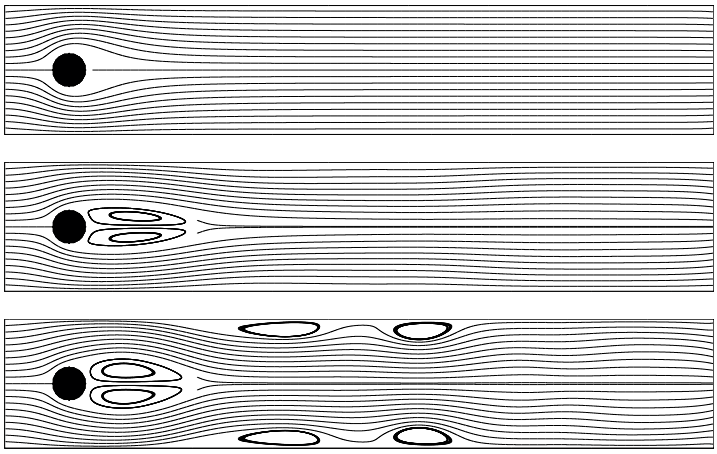


Figure 1.3: Stream lines of the velocity field of laminar flow around a circular cylinder for $Re = 3$ (top), $Re = 36$ (middle), and $Re = 108$ (bottom).

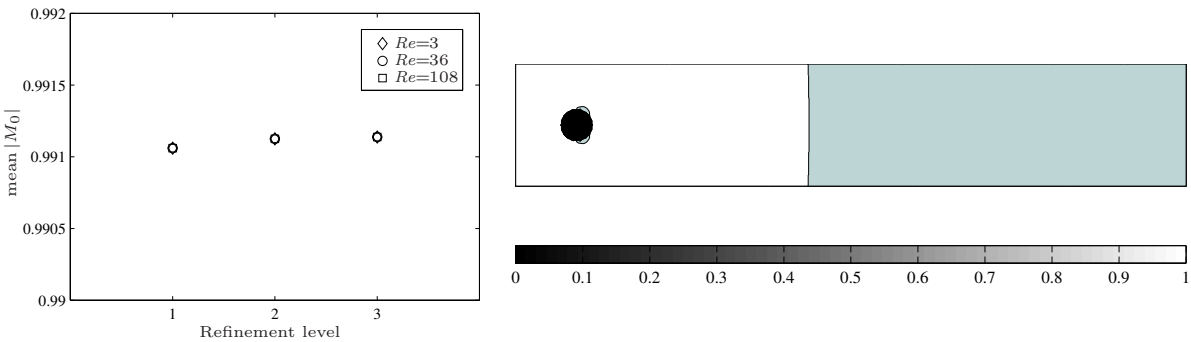


Figure 1.4: Left: mean absolute value of M_0 in each simulation, right: M_0 for $Re = 3$.

and carry out experiments for the dynamic viscosities $\mu = 1/12$ ($\omega = 4/3$), $\mu = 1/144$ ($\omega = 48/25$), and $\mu = 1/432$ ($\omega = 144/73$). Therefore, the dimensionless Reynolds number $Re = \varrho UL/\mu$ takes the values $Re = 3$, $Re = 36$, and $Re = 108$. We perform 10000 time steps on the 220×20 lattice, 20000 time steps on the 440×80 lattice, and 40000 time steps on the 880×160 lattice. Note that computation times are chosen such that a given fluid particle is moving the same dimensionless distance in each simulation (remember that $U = \Delta x$). In the remainder of this Section, we will refer to the 220×40 lattice as refinement level one, to the 440×80 lattice as refinement level two, and to the 880×160 lattice as refinement level three. The final velocity fields of the experiments on refinement level three are visualised in Figure 1.3.

Let us now turn our attention to the moments M_0 to M_8 of the particle distribution function f. For the above simulations, the mean absolute values of those moments are visualised in Figures 1.4 to 1.12. Furthermore, each Figure contains an exemplifying contour plot of the corresponding moment for $Re = 3$ on refinement level three.

We find that in each simulation, the moments M_0 to M_5 can be considered to be $\mathcal{O}(1)$ throughout the computational domain but M_6 and M_7 may take absolute values that are somewhat too large, even though their respective mean absolute values are clearly $\mathcal{O}(1)$. Furthermore, M_8 may take values that are far too large and even its mean absolute value is very often larger that $\mathcal{O}(1)$.

To obtain more detailed information about this problem, we consider histograms showing the decadic logarithm of the frequency of occurrence of the values of M_6, M_7, and M_8 in Figures 1.13, 1.14, respectively 1.15. The logarithm is plotted instead of the frequency of occurrence itself to show more details for values which do not appear very often. We learn from those histograms that nearly all values of the moments under investigation can be considered to be $\mathcal{O}(1)$. In fact, values larger than $\mathcal{O}(1)$ are so uncommon that they can be considered spurious anomalies. To get an idea of the positions where those anomalies appear, we consider plots that mark all points where $|M_6| > 10$, $|M_7| > 10$, and $|M_8| > 10$ in the simulation for $Re = 3$ on refinement level three (Figure 1.16). It is evident that all the points where

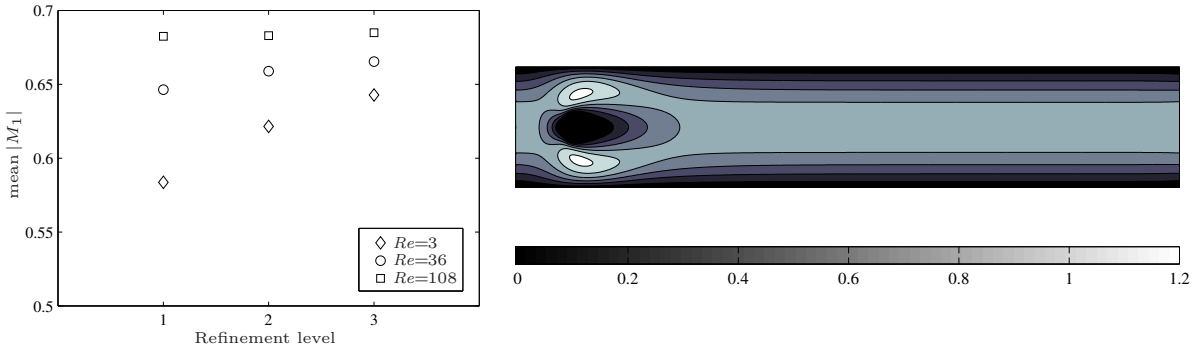


FIGURE 1.5: Left: mean absolute value of M_1 in each simulation, right: M_1 for $Re = 3$.

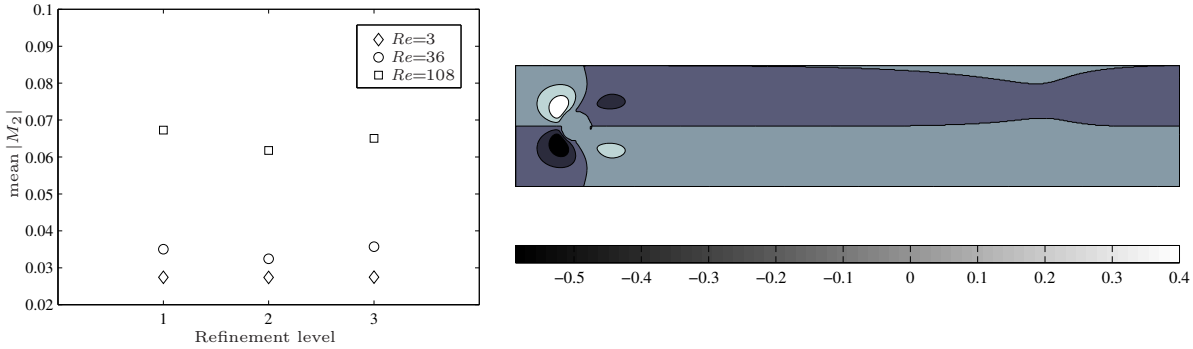


FIGURE 1.6: Left: mean absolute value of M_2 in each simulation, right: M_2 for $Re = 3$.

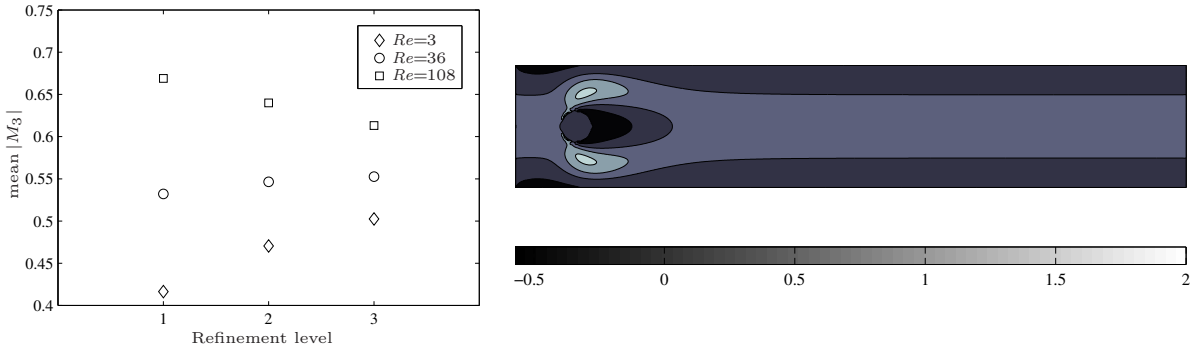


FIGURE 1.7: Left: mean absolute value of M_3 in each simulation, right: M_3 for $Re = 3$.

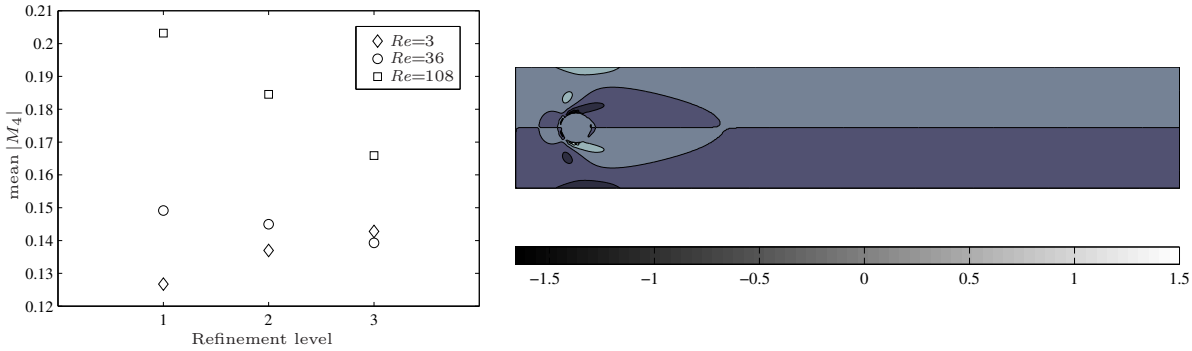


FIGURE 1.8: Left: mean absolute value of M_4 in each simulation, right: M_4 for $Re = 3$.

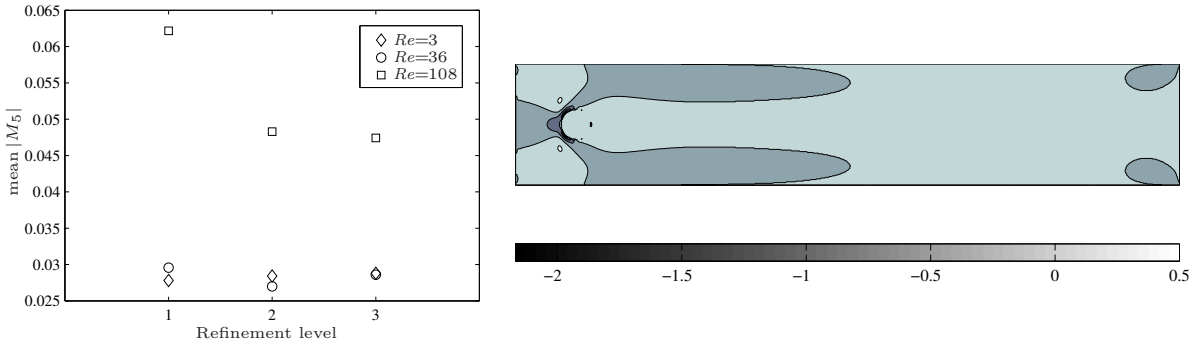


Figure 1.9: Left: mean absolute value of M_5 in each simulation, right: M_5 for $Re = 3$.

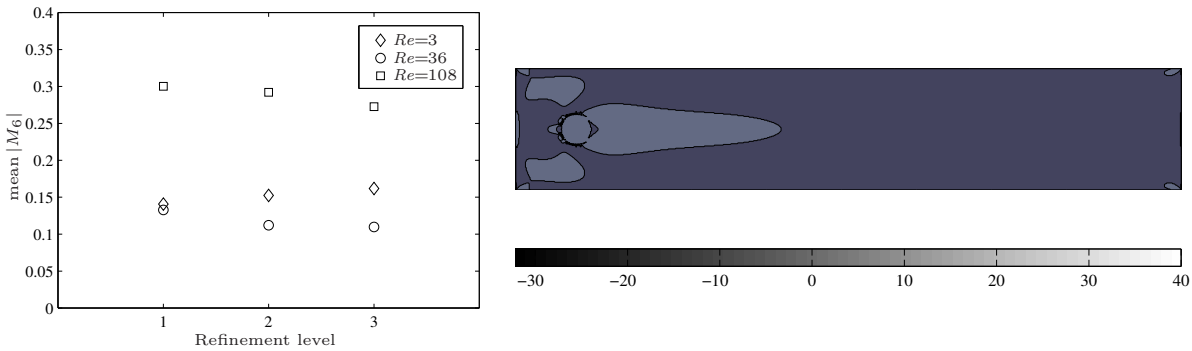


Figure 1.10: Left: mean absolute value of M_6 in each simulation, right: M_6 for $Re = 3$.

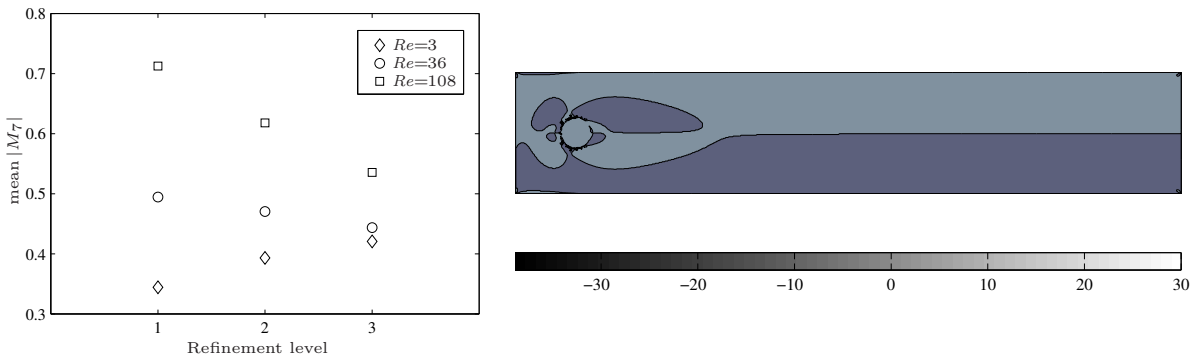


Figure 1.11: Left: mean absolute value of M_7 in each simulation, right: M_7 for $Re = 3$.

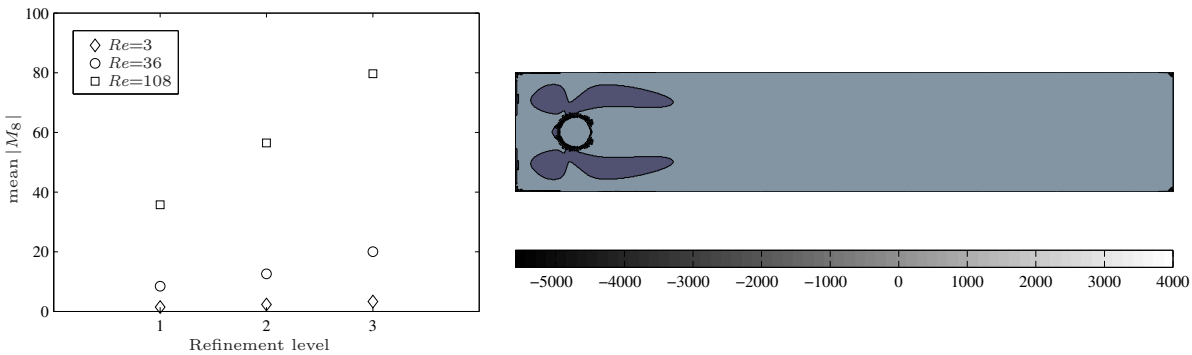


Figure 1.12: Left: mean absolute value of M_8 in each simulation, right: M_8 for $Re = 3$.

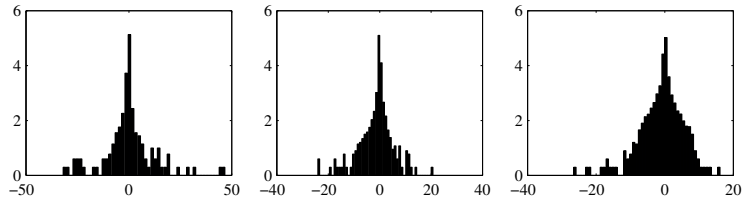


Figure 1.13: Histograms showing the values of M_6 (abscissa) versus the decadic logarithm of their frequency of occurrence (ordinate) for $Re = 3$ (left), $Re = 36$ (middle), and $Re = 108$ (right), as computed on refinement level three.

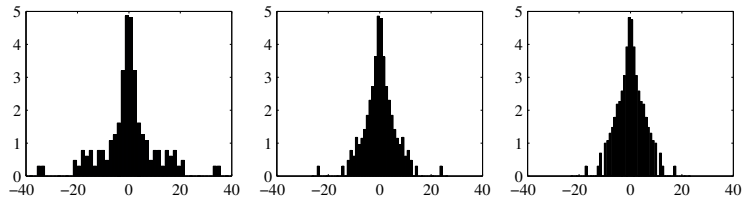


Figure 1.14: Histograms showing the values of M_7 (abscissa) versus the decadic logarithm of their frequency of occurrence (ordinate) for $Re = 3$ (left), $Re = 36$ (middle), and $Re = 108$ (right), as computed on refinement level three.

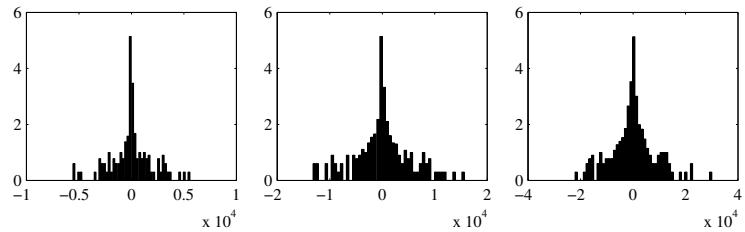


Figure 1.15: Histograms showing the values of M_8 (abscissa) versus the decadic logarithm of their frequency of occurrence (ordinate) for $Re = 3$ (left), $Re = 36$ (middle), and $Re = 108$ (right), as computed on refinement level three.

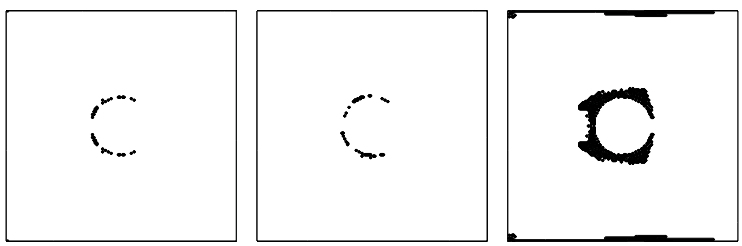


Figure 1.16: Position of the anomalies in M_6 (left), M_7 (middle), and M_8 (right) for $Re = 3$. Only the front part of the channel is shown because for each of the moments under consideration, appreciable anomalies appear only there.

any one of the problematic moments has an absolute value larger than 10 are located in the neighbourhood of a solid boundary, a great many of them in the surroundings of the obstacle. Also the points of contact between the solid boundaries and the inflow region turn out to be a source of spurious anomalies.

Remember now that the bounce back condition (1.20) is known to be somewhat inaccurate at curved boundaries. Remember also that M_6 and M_7 are computed using the polynomials (1.12c) that scale with $1/\Delta x^3$ while the calculation of M_8 is based on (1.12d) which even scales with $1/\Delta x^4$. Therefore, small errors in the particle distribution function f produce large errors in M_6, M_7, and M_8. Note that sensitivity to such a defect is increasing with Re because regularity of the flow is decreasing with Re. Note also that by rough analysis of the bulk flow one finds that in lowest order, M_6, M_7, and M_8 depend on derivatives of $\boldsymbol{u}$ of order three and more, so the absolute values of those moments can be expected to be large if the derivatives of $\boldsymbol{u}$ are large which is in fact the case in that part of the channel which contains the obstacle, compare Figure 1.3. Furthermore, such an analysis yields that none of the moments under

consideration is convected with the velocity field. However, the analysis performed here is not valid at the boundary, so we do not know whether its results are valid in the vicinity of the solid walls. See Cornubert et al. [**12**] for a study of similar effects in the vicinity of solid boundaries.

Concluding this Section we hold the bounce-back condition (1.20) and the amplification of small errors during moment calculation responsible for disturbing the order of magnitude of higher order moments in the vicinity of no-slip boundaries. Furthermore, we did not explicitly care about proper treatment of the points of contact between the no-slip boundaries and the inflow/outflow regions, so the appearance of errors in the surroundings of those points is not really surprising. Therefore, we state that the polynomials (1.12) are in fact scaled such that they produce moments of the particle density f which are of $\mathcal{O}(1)$ in the bulk of the flow.

CHAPTER 2

Modelling of two-phase flow

Advancing the lattice BGK equation to a suitable model for immiscible two-phase flow is not at all straightforward. Mathematical theory is complicated by the presence of an interface, where both density and stress tensor may have a jump singularity and surface tension effects must be taken into account. Furthermore, the treatment of those singularities in the context of the lattice BGK method is a problem of its own.

In this Chapter, we will consider those difficulties in detail. Section 2.1 contains the mathematical theory of two-phase flow, as far as it is needed for the following, and Section 2.2 introduces a corresponding LBGK model, namely the so-called immiscible lattice BGK approach. Section 2.3 contains a detailed investigation of the intrinsic interface tracking method of immiscible lattice BGK and finally, Section 2.4 describes the modelling of surface tension effects.

2.1. Mathematical theory of incompressible two-phase flow

This Section provides the theoretical background for the derivation of the immiscible lattice BGK method. Assuming the flow in each phase is governed by the incompressible Navier-Stokes equations we will deduce one system of equations governing the flow in the whole domain. This system will consist of weak forms of the incompressible Navier-Stokes equations and the transport equation, the latter describing the motion of the interface.

After stating the basic assumptions of the theory in Section 2.1.1, we will derive the full-space form of the governing equations in Section 2.1.2 and in Section 2.1.3, we will present a reformulation of the full-space incompressible Navier-Stokes equations. This reformulation will turn out to be essential for simulating surface tension effects with the immiscible lattice BGK model (compare Section 2.4.2).

2.1.1. Basic assumptions. We consider a bounded open domain $\Omega = (0, L_1) \times \cdots \times (0, L_d) \subset \mathbb{R}^d$ with $L_j > 0$, for $j = 1, \ldots, d$, and require all quantities appearing here to be periodic at the boundary $\partial\Omega$. We assume some red fluid occupies a bounded open domain $\Omega^r(t) \subset \Omega$ and some blue fluid covers its open complement $\Omega^b(t) = \Omega \setminus \overline{\Omega^r(t)}$. The phases are separated by the regular interface $\Gamma(t) = \partial\Omega^r(t)$ and we define the indicator function $\chi : [0, t_{max}] \times \Omega \to \{0, 1\}$ such that

$$\chi(t, \boldsymbol{x}) = \begin{cases} 1 & \text{for } \boldsymbol{x} \in \Omega^r(t) \\ 0 & \text{for } \boldsymbol{x} \in \Omega^b(t). \end{cases} \tag{2.1}$$

The whole geometry is exemplified in Figure 2.1. Furthermore, we assume the movement of Γ to be determined by the velocity field $\boldsymbol{u} : [0, t_{max}] \times \Omega \to \mathbb{R}^d$ according to

$$\Gamma(t) = \{\boldsymbol{x}(t) \,:\, \partial_t \boldsymbol{x}(t) = \boldsymbol{u}(t, \boldsymbol{x}(t)),\, \boldsymbol{x}(0) \in \Gamma(0)\}\,.$$

In addition, we suppose

$$\Omega^l(t) = \left\{\boldsymbol{x}(t) \,:\, \partial_t \boldsymbol{x}(t) = \boldsymbol{u}(t, \boldsymbol{x}(t)),\, \boldsymbol{x}(0) \in \Omega^l(0)\right\} \qquad \text{for } l = r, b,$$

as well as

$$\boldsymbol{u}\Big|_{[0,t_{max}]\times\Omega^r(t)} \in \mathcal{C}^2\left([0, t_{max}] \times \Omega^r(t), \mathbb{R}^d\right)$$

and

$$\boldsymbol{u}\Big|_{[0,t_{max}]\times\Omega^b(t)} \in \mathcal{C}^2\left([0, t_{max}] \times \Omega^b(t), \mathbb{R}^d\right).$$

Finally, to make sure the ordinary differential equation $\partial_t \boldsymbol{x} = \boldsymbol{u}(t, \boldsymbol{x})$ has a unique solution in $[0, t_{max}]$, we assume $\boldsymbol{u}$ is uniformly Lipschitz continuous in $\Omega = \Omega^r(t) \cup \Omega^b(t) \cup \Gamma(t)$. This implies that in each phase, $\chi(t, \boldsymbol{x})$ is described by the transport equation

$$\partial_t \chi + u_\alpha \partial_\alpha \chi = 0. \tag{2.2}$$

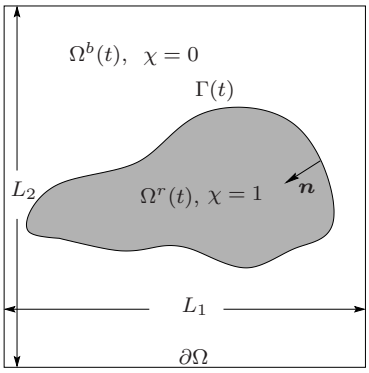


Figure 2.1: Example of a two-phase flow geometry in $\mathbb{R}^2$

2.1.2. Derivation of the full-space equations. Let now the flow in each phase be governed by the incompressible Navier-Stokes equations (1.8). Note that the density ϱ is supposed to be constant in each phase, i. e.

$$\varrho(t,\boldsymbol{x}) = \begin{cases} \varrho^r & \text{for } \boldsymbol{x} \in \Omega^r(t) \\ \varrho^b & \text{for } \boldsymbol{x} \in \Omega^b(t), \end{cases}$$

where ϱ^r and ϱ^b are non-negative real constants, and that due to the first equation in (1.8), the transport equation (2.2) can be written in the form

$$\partial_t \chi + \partial_\alpha \left(u_\alpha \chi\right) = 0. \tag{2.3}$$

The conditions at the interface $\Gamma(t)$ are [**49, 68**]

$$[u_\alpha] = 0 \qquad \text{and} \qquad [\tau_{\alpha\beta} n_\beta] = \sigma\kappa n_\alpha, \tag{2.4}$$

where σ is the coefficient of surface tension, κ is the curvature of Γ, $\boldsymbol{n}$ denotes the inner unit normal vector of Γ (see again Figure 2.1), and $[\cdot]$ denotes the jump over the interface, i. e.

$$[g](t,\boldsymbol{x}) = \lim_{\substack{\boldsymbol{y}\to\boldsymbol{x} \\ \boldsymbol{y}\in\Omega^b(t)}} g(t,\boldsymbol{y}) - \lim_{\substack{\boldsymbol{y}\to\boldsymbol{x} \\ \boldsymbol{y}\in\Omega^r(t)}} g(t,\boldsymbol{y}), \qquad \text{for } \boldsymbol{x} \in \Gamma(t).$$

For our purpose, it is convenient to transform the one-phase Navier-Stokes equations (1.8) together with the jump conditions (2.4) into one system of partial differential equations on the whole domain Ω, as it is frequently done in mathematical modelling of two-phase flow. For example, Brackbill et al. [**5**] give a rather physical approach to this transformation while Chang et al. [**8**] as well as Miller [**56**] arrive at similar results using level set functions. In the following, we will present a transformation based on distribution theory [**13, 18, 69**].

We start with defining $\mathcal{D}(\Omega) = \mathcal{C}_0^\infty(\Omega,\mathbb{R})$ and denoting the space of continuous linear forms on $\mathcal{D}(\Omega)$ by $\mathcal{D}'(\Omega)$. If $K \in \mathcal{D}'(\Omega)$ we mark its value on the test function $\varphi \in \mathcal{D}(\Omega)$ with $\langle K, \varphi\rangle_\Omega$. Furthermore, we say that a sequence $\{\varphi_k \in \mathcal{D}(\Omega)\}_{k\in\mathbb{N}}$ converges in $\mathcal{D}(\Omega)$ to $\varphi \in \mathcal{D}(\Omega)$ if (i) there is a fixed compact set $\Phi \subset \Omega$ such that $\operatorname{supp}\varphi_k \subset \Phi$ for each $k \in \mathbb{N}$ and (ii) for any multi-index J each sequence $\{\partial_J \varphi_k\}_{k\in\mathbb{N}}$ of $|J|$-th derivatives of the φ_k converges uniformly on Φ to $\partial_J\varphi$. Continuity of $K \in \mathcal{D}'(\Omega)$ denotes that for any sequence of test functions $\{\varphi_k\}_{k\in\mathbb{N}}$ with $\lim_{k\to\infty}\varphi_k = \varphi$ we have

$$\lim_{k\to\infty} \langle K, \varphi_k\rangle_\Omega = \langle K, \varphi\rangle_\Omega$$

and linearity of $K \in \mathcal{D}'(\Omega)$ means that for any $a, b \in \mathbb{R}$ and arbitrary $\varphi_1, \varphi_2 \in \mathcal{D}(\Omega)$ it is true that

$$\langle K, a\varphi_1 + b\varphi_2\rangle_\Omega = a\langle K, \varphi_1\rangle_\Omega + b\langle K, \varphi_2\rangle_\Omega.$$

Additionally, we say that a sequence $\{K_k\}_{k\in\mathbb{N}}$ of continuous linear forms converges in $\mathcal{D}'(\Omega)$ to the continuous linear form K if for every $\varphi \in \mathcal{D}(\Omega)$

$$\lim_{k\to\infty} \langle K - K_k, \varphi\rangle_\Omega = 0. \tag{2.5}$$

The space $\mathcal{D}'(\Omega)$ is often referred to as the dual space of $\mathcal{D}(\Omega)$ and a continuous linear form $K \in \mathcal{D}'(\Omega)$ is called a distribution on Ω. Distributions that can be expressed in the form

$$\langle K, \varphi \rangle_\Omega = \int_\Omega K(\boldsymbol{x})\varphi(\boldsymbol{x})\, d\boldsymbol{x}$$

are called regular while all others are denoted as singular.

Now, we start the derivation of the full-space equations. First, we multiply the single-phase Navier-Stokes equations (1.8) with an arbitrary test function $\varphi \in \mathcal{D}(\Omega)$ and integrate over $\Omega^r(t)$ as well as $\Omega^b(t)$. This results in

$$\int_{\Omega^l(t)} \partial_\alpha u_\alpha \varphi\, d\boldsymbol{x} = 0 \qquad \text{and} \qquad \int_{\Omega^l(t)} \partial_t(\varrho u_\alpha)\varphi\, d\boldsymbol{x} + \int_{\Omega^l(t)} \partial_\beta(\varrho u_\alpha u_\beta)\varphi\, d\boldsymbol{x} = \int_{\Omega^l(t)} (\partial_\beta \tau_{\alpha\beta})\varphi\, d\boldsymbol{x} \quad (2.6)$$

for $l = r, b$. Applying the transport theorem [**15**] provides

$$\int_{\Omega^l(t)} \partial_t(\varrho u_\alpha)\varphi\, d\boldsymbol{x} = \frac{d}{dt}\int_{\Omega^l(t)} \varrho u_\alpha \varphi\, d\boldsymbol{x} - \int_{\Omega^l(t)} \partial_\beta(\varrho u_\alpha u_\beta \varphi)\, d\boldsymbol{x}$$

and the divergence theorem yields [**16**]

$$-\int_{\Omega^l(t)} \partial_\beta(\varrho u_\alpha u_\beta \varphi)\, d\boldsymbol{x} = \int_{\partial\Omega^l(t)} \varrho u_\alpha u_\beta n^l_\beta \varphi\, d\mathscr{h}^l,$$

where $\boldsymbol{n}^l$ denotes the inner unit normal vector of $\partial\Omega^l(t)$ and $\mathscr{h}^l$ stands for the surface measure on $\partial\Omega^l(t)$. Therefore, we have

$$\int_{\Omega^l(t)} \partial_t(\varrho u_\alpha)\varphi\, d\boldsymbol{x} = \frac{d}{dt}\int_{\Omega^l(t)} \varrho u_\alpha \varphi\, d\boldsymbol{x} + \int_{\partial\Omega^l(t)} \varrho u_\alpha u_\beta n^l_\beta \varphi\, d\mathscr{h}^l.$$

Exploiting again the divergence theorem we obtain

$$\int_{\Omega^l(t)} \partial_\alpha u_\alpha \varphi\, d\boldsymbol{x} = -\int_{\partial\Omega^l(t)} u_\alpha n^l_\alpha \varphi\, d\mathscr{h}^l - \int_{\Omega^l(t)} u_\alpha \partial_\alpha \varphi\, d\boldsymbol{x},$$

$$\int_{\Omega^l(t)} \partial_\beta(\varrho u_\alpha u_\beta)\varphi\, d\boldsymbol{x} = -\int_{\partial\Omega^l(t)} \varrho u_\alpha u_\beta n^l_\beta \varphi\, d\mathscr{h}^l - \int_{\Omega^l(t)} \varrho u_\alpha u_\beta \partial_\beta \varphi\, d\boldsymbol{x},$$

as well as

$$\int_{\Omega^l(t)} (\partial_\beta \tau_{\alpha\beta})\varphi\, d\boldsymbol{x} = -\int_{\partial\Omega^l(t)} \tau_{\alpha\beta} n^l_\beta \varphi\, d\mathscr{h}^l - \int_{\Omega^l(t)} \tau_{\alpha\beta}\partial_\beta \varphi\, d\boldsymbol{x}.$$

Inserting now all those results into (2.6) and remembering φ is compactly supported on Ω we end up with

$$\int_{\Omega^l(t)} u_\alpha \partial_\alpha \varphi\, d\boldsymbol{x} + \int_{\Gamma(t)} u_\alpha n^l_\alpha \varphi\, d\mathscr{h} = 0 \tag{2.7a}$$

and

$$\frac{d}{dt}\int_{\Omega^l(t)} \varrho u_\alpha \varphi\, d\boldsymbol{x} - \int_{\Omega^l(t)} \varrho u_\alpha u_\beta \partial_\beta \varphi\, d\boldsymbol{x} = -\int_{\Omega^l(t)} \tau_{\alpha\beta}\partial_\beta \varphi\, d\boldsymbol{x} - \int_{\Gamma(t)} \tau_{\alpha\beta} n^l_\beta \varphi\, d\mathscr{h}, \tag{2.7b}$$

where $\mathscr{h}$ denotes the surface measure on $\Gamma(t)$. Since the functions $u_\alpha : [0, t_{max}] \times \Omega \to \mathbb{R}$, $\varrho u_\alpha : [0, t_{max}] \times \Omega \to \mathbb{R}$, $\varrho u_\alpha u_\beta : [0, t_{max}] \times \Omega \to \mathbb{R}$, and $\tau_{\alpha\beta} : [0, t_{max}] \times \Omega \to \mathbb{R}$ are smooth on Ω up to at most a finite jump at $\Gamma(t)$ we know that u_α, ϱu_α, $\varrho u_\alpha u_\beta$, and $\tau_{\alpha\beta}$ are regular distributions on Ω, so we have

$$\langle K, \varphi \rangle_\Omega = \int_{\Omega^r(t)} K\varphi\, d\boldsymbol{x} + \int_{\Omega^b(t)} K\varphi\, d\boldsymbol{x}$$

for $K \in \{u_\alpha, \varrho u_\alpha, \varrho u_\alpha u_\beta, \tau_{\alpha\beta}\}$ and some arbitrary $\varphi \in \mathcal{D}(\Omega)$. Furthermore, using the relation $\boldsymbol{n} = \boldsymbol{n}^r = -\boldsymbol{n}^b$ on $\Gamma(t)$ we can now rewrite (2.7) in the form

$$\langle u_\alpha, \partial_\alpha \varphi \rangle_\Omega - \int_{\Gamma(t)} [u_\alpha] n_\alpha \varphi\, d\mathscr{h} = 0$$

respectively

$$\frac{d}{dt}\langle \varrho u_\alpha, \varphi \rangle_\Omega - \langle \varrho u_\alpha u_\beta, \partial_\beta \varphi \rangle_\Omega = -\langle \tau_{\alpha\beta}, \partial_\beta \varphi \rangle_\Omega + \int_{\Gamma(t)} [\tau_{\alpha\beta} n_\beta]\varphi\, d\mathscr{h}.$$

At this stage, we exploit the jump conditions (2.4) which results in

$$\langle u_\alpha, \partial_\alpha \varphi\rangle_\Omega = 0 \qquad \text{and} \qquad \frac{d}{dt}\langle \varrho u_\alpha, \varphi\rangle_\Omega - \langle \varrho u_\alpha u_\beta, \partial_\beta \varphi\rangle_\Omega = -\langle \tau_{\alpha\beta}, \partial_\beta \varphi\rangle_\Omega + \int_{\Gamma(t)} \sigma\kappa n_\alpha \varphi \, d\mathscr{h}. \tag{2.8}$$

Let us now consider differentiation in $\mathcal{D}'(\Omega)$. If $K \in \mathcal{C}^1(\Omega, \mathbb{R})$ the divergence theorem yields $\langle \partial_\alpha K, \varphi\rangle_\Omega = -\langle K, \partial_\alpha \varphi\rangle_\Omega$ for each $\varphi \in \mathcal{D}(\Omega)$. In analogy to this rigorous result, the derivative in the sense of distributions is for any $K \in \mathcal{D}'(\Omega)$ defined by

$$\langle \partial_\alpha K, \varphi\rangle_\Omega = -\langle K, \partial_\alpha \varphi\rangle_\Omega, \qquad \text{for each } \varphi \in \mathcal{D}(\Omega). \tag{2.9}$$

Moreover, given some $K : [0, t_{max}] \to \mathcal{D}'(\Omega)$ we say that $K \in \mathcal{C}^1\big([0, t_{max}], \mathcal{D}'(\Omega)\big)$ if $\langle K(t), \varphi\rangle_\Omega \in \mathcal{C}^1([0, t_{max}], \mathbb{R})$ for each test function $\varphi \in \mathcal{D}'(\Omega)$ and $\partial_t K$ is then defined by

$$\langle \partial_t K(t), \varphi\rangle_\Omega = \frac{d}{dt}\langle K(t), \varphi\rangle_\Omega, \qquad \text{where} \qquad \frac{d}{dt}\langle K(t), \varphi\rangle_\Omega = \lim_{\Delta t \to 0} \left\langle \frac{K(t+\Delta t) - K(t)}{\Delta t}, \varphi \right\rangle_\Omega$$

for each $\varphi \in \mathcal{D}(\Omega)$. From (2.5), we obtain $\partial_t K \in \mathcal{D}'(\Omega)$ and thus, we conclude there is a distribution $\partial_t(\varrho u_\alpha) \in \mathcal{D}'(\Omega)$ given by

$$\langle \partial_t(\varrho u_\alpha), \varphi\rangle_\Omega = \frac{d}{dt}\langle \varrho u_\alpha, \varphi\rangle_\Omega \qquad \text{for each } \varphi \in \mathcal{D}(\Omega). \tag{2.10}$$

Finally, we want to express the surface integral in equation (2.8) by a continuous linear form in $\mathcal{D}'(\Omega)$. Thus, we define $\delta_{\Gamma(t)}$ for each $g \in \mathcal{C}^0(\Gamma, \mathbb{R})$ and any $\varphi \in \mathcal{D}(\Omega)$ by

$$\langle g\delta_{\Gamma(t)}, \varphi\rangle_\Omega = \int_{\Gamma(t)} g\varphi \, d\mathscr{h}. \tag{2.11}$$

This definition yields

$$\langle n_\alpha \delta_\Gamma, \varphi\rangle_\Omega = \int_\Gamma n_\beta \delta_{\alpha\beta} \varphi \, d\mathscr{h},$$

due to the divergence theorem, we obtain

$$\int_\Gamma n_\beta \delta_{\alpha\beta} \varphi \, d\mathscr{h} = -\int_{\Omega^r} \partial_\beta(\varphi \delta_{\alpha\beta}) \, d\boldsymbol{x},$$

and recalling definitions (2.1) and (2.9) we find that

$$-\int_{\Omega^r} \partial_\beta(\varphi\delta_{\alpha\beta}) \, d\boldsymbol{x} = -\int_\Omega \chi \partial_\alpha \varphi \, d\boldsymbol{x} = -\langle \chi, \partial_\alpha \varphi\rangle_\Omega = \langle \partial_\alpha \chi, \varphi\rangle_\Omega .$$

Consequently, we obtain

$$\langle \partial_\alpha \chi, \varphi\rangle_\Omega = \langle n_\alpha \delta_\Gamma, \varphi\rangle_\Omega .$$

Using (2.9), (2.10), and (2.11) we can reformulate the incompressible Navier-Stokes equations (2.8) as

$$\langle \partial_\alpha u_\alpha, \varphi\rangle_\Omega = 0 \qquad \text{and} \qquad \langle \partial_t(\varrho u_\alpha), \varphi\rangle_\Omega + \langle \partial_\beta(\varrho u_\alpha u_\beta), \varphi\rangle_\Omega = \langle \partial_\beta \tau_{\alpha\beta}, \varphi\rangle_\Omega + \langle \sigma\kappa n_\alpha \delta_\Gamma, \varphi\rangle_\Omega.$$

In addition, we find in complete analogy to the above derivation that the full-space form of the transport equation (2.3) is

$$\langle \partial_t \chi, \varphi\rangle_\Omega + \langle \partial_\alpha (u_\alpha \chi), \varphi\rangle_\Omega = 0.$$

Since all those equations hold for each $\varphi \in \mathcal{D}(\Omega)$ they express relations between distributions on Ω. Thus, we can formulate

Theorem 2.1. *On the assumptions summarised in Section 2.1.1, the incompressible Navier-Stokes equations* (1.8) *on Ω^l for $l = r, b$ and the jump conditions* (2.4) *on Γ can be merged to the full-space Navier-Stokes equations in $\mathcal{D}'(\Omega)$,*

$$\partial_\alpha u_\alpha = 0 \qquad \textit{and} \qquad \partial_t(\varrho u_\alpha) + \partial_\beta(\varrho u_\alpha u_\beta) = \partial_\beta \tau_{\alpha\beta} + \sigma\kappa n_\alpha \delta_\Gamma. \tag{2.12}$$

Furthermore, on the same assumptions, the transport equation (2.3) *can be written in $\mathcal{D}'(\Omega)$ as*

$$\partial_t \chi + \partial_\alpha(u_\alpha \chi) = 0 \tag{2.13}$$

and the indicator distribution $\chi \in \mathcal{D}'(\Omega)$ fulfils

$$\partial_\alpha \chi = n_\alpha \delta_\Gamma. \tag{2.14}$$

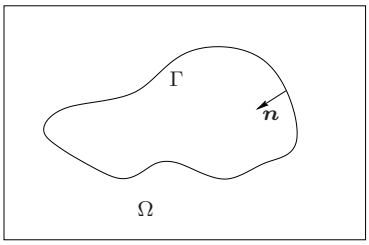


Figure 2.2: Example of an oriented one-dimensional $\mathcal{C}^2$-surface Γ with inner unit normal vector $\boldsymbol{n}$ in a domain $\Omega \subset \mathbb{R}^2$.

2.1.3. Divergence form of the surface tension term. In Section 2.4.2, we will find it essential to make use of the relation

$$\sigma\kappa n_\alpha \delta_\Gamma = \sigma\partial_\beta\left((\delta_{\alpha\beta} - n_\alpha n_\beta)\delta_\Gamma\right). \tag{2.15}$$

By means of a sketchy and formal proof, Lafaurie et al. [**47**] already demonstrated the availability of (2.15) in three-dimensional space. In the following, we will present a rigorous derivation in d-dimensional space, for $2 \leq d \in \mathbb{N}$, which is based only on elementary tools from differential geometry [**7, 74**].

For this purpose, let us consider an oriented $(d-1)$-dimensional $\mathcal{C}^2$-surface $\Gamma \subset \Omega \subset \mathbb{R}^d$, as exemplified in Figure 2.2. First, we recall that due to (2.9) and (2.11), we have

$$\left\langle \partial_\beta\Big((\delta_{\alpha\beta} - n_\alpha n_\beta)\delta_\Gamma\Big), \varphi\right\rangle_\Omega = -\int_\Gamma (\delta_{\alpha\beta} - n_\alpha n_\beta)\partial_\beta\varphi \, d\mathscr{H}$$

for each $\varphi \in \mathcal{D}(\Omega)$. Then, we remark that since Γ is a $\mathcal{C}^2$-surface there exists for each point $\boldsymbol{y} \in \Gamma$ an open neighbourhood $\mathcal{U}^k \subset \Omega$ such that $\boldsymbol{y} \in \mathcal{U}^k$ and $\mathcal{U}^k \cap \Gamma$ is the image of a bijective $\mathcal{C}^2$-mapping $\boldsymbol{h}^k : \mathbb{R}^{d-1} \supset \Theta^k \to \Omega \subset \mathbb{R}^d$. Next, we define $\left(\varphi \circ \boldsymbol{h}^k\right) : \Theta^k \to \mathbb{R}$, where '$\circ$' means $\left(\varphi \circ \boldsymbol{h}^k\right)(\boldsymbol{z}^k) = \varphi\left(\boldsymbol{h}^k(\boldsymbol{z}^k)\right)$ for each $\boldsymbol{z}^k \in \Theta^k$. If we now choose a partition of unity $\left\{\psi^k\right\}$ such that the support of $\left(\varphi\psi^k\right) \circ \boldsymbol{h}^k$ is compact in Θ^k we can write

$$-\int_\Gamma (\delta_{\alpha\beta} - n_\alpha n_\beta)\partial_\beta\varphi \, d\mathscr{H} = -\sum_k \left(\int_\Gamma (\delta_{\alpha\beta} - n_\alpha n_\beta)\partial_\beta\left(\varphi\psi^k\right) \, d\mathscr{H}\right).$$

Our next aim is to transform the integral on the right hand side of this equation to an integral over Θ^k. We know the columns of the matrix

$$\begin{pmatrix} \partial_1 h_1^k & \cdots & \partial_{d-1} h_1^k \\ \vdots & & \vdots \\ \partial_1 h_d^k & \cdots & \partial_{d-1} h_d^k \end{pmatrix}$$

to be tangential vectors of Γ which implies that for the inner unit normal vector $\boldsymbol{n}$ of Γ we have

$$n_\beta \partial_\gamma h_\beta^k = 0, \tag{2.16}$$

where $\beta = 1, \ldots, d$ and $\gamma = 1, \ldots, d-1$. From this, we obtain $\partial_\varepsilon\left(n_\beta\partial_\gamma h_\beta^k\right) = 0$ for $\varepsilon = 1 \ldots d-1$ and therefore,

$$\partial_\varepsilon n_\beta \partial_\gamma h_\beta^k + n_\beta \partial_{\gamma\varepsilon} h_\beta^k = 0. \tag{2.17}$$

Assuming now $\varphi \in \mathcal{D}(\Omega)$ we can decompose $\partial_\beta\varphi$ at any point $\boldsymbol{y} \in \Gamma$ according to

$$\partial_\beta\varphi = a_\gamma \partial_\gamma h_\beta^k + b n_\beta$$

with $a_\gamma, b \in \mathbb{R}$. Using (2.16) and $n_\alpha n_\alpha = 1$ we conclude

$$(\delta_{\alpha\beta} - n_\alpha n_\beta)\partial_\beta\varphi = a_\gamma \partial_\gamma h_\alpha^k \tag{2.18}$$

for $\alpha = 1, \ldots, d$. Using again (2.16) we find that

$$\partial_\varepsilon\left(\varphi \circ \boldsymbol{h}^k\right) = \partial_\varepsilon h_\beta^k \partial_\beta\varphi = \partial_\varepsilon h_\beta^k a_\gamma \partial_\gamma h_\beta^k.$$

Thus, we can express a_γ in the form

$$a_\gamma = \left(\partial_\varepsilon h_\beta^k \partial_\gamma h_\beta^k\right)^{-1} \partial_\varepsilon\left(\varphi \circ \boldsymbol{h}^k\right),$$

where the superscript '-1' denotes the corresponding entry of the inverse matrix, and inserting this result into (2.18) we obtain

$$(\delta_{\alpha\beta} - n_\alpha n_\beta)\partial_\beta \varphi = \partial_\gamma h^k_\alpha \left(\partial_\varepsilon h^k_\beta \partial_\gamma h^k_\beta\right)^{-1} \partial_\varepsilon \left(\varphi \circ \boldsymbol{h}^k\right).$$

Together with the substitution theorem, this finally yields

$$-\int_\Gamma (\delta_{\alpha\beta} - n_\alpha n_\beta)\partial_\beta \left(\varphi\psi^k\right)\, d\mathcal{H} = -\int_{\Theta^k} \partial_\gamma h^k_\alpha \left(\partial_\varepsilon h^k_\beta \partial_\gamma h^k_\beta\right)^{-1} \partial_\varepsilon \left(\left(\varphi\psi^k\right) \circ \boldsymbol{h}^k\right) \mathcal{J}\, d\boldsymbol{z}^k,$$

where

$$\mathcal{J} = \left\{ \det \left(\begin{pmatrix} \partial_1 h^k_1 & \cdots & \partial_1 h^k_d \\ \vdots & & \vdots \\ \partial_{d-1} h^k_1 & \cdots & \partial_{d-1} h^k_d \end{pmatrix} \begin{pmatrix} \partial_1 h^k_1 & \cdots & \partial_{d-1} h^k_1 \\ \vdots & & \vdots \\ \partial_1 h^k_d & \cdots & \partial_{d-1} h^k_d \end{pmatrix} \right) \right\}^{1/2} \tag{2.19}$$

denotes the Jacobian of the transformation $\mathcal{U}^k \cap \Gamma \ni \boldsymbol{y} \mapsto \boldsymbol{z}^k \in \Theta^k$ [**16**]. Since we chose $\left\{\psi^k\right\}$ such that the support of $\left(\varphi\psi^k\right) \circ \boldsymbol{h}^k$ is compact in Θ^k we can now use the divergence theorem to achieve

$$-\int_{\Theta^k} \mathcal{J}\partial_\gamma h^k_\alpha \left(\partial_\varepsilon h^k_\beta \partial_\gamma h^k_\beta\right)^{-1} \partial_\varepsilon \left(\left(\varphi\psi^k\right) \circ \boldsymbol{h}^k\right) d\boldsymbol{z}^k = \int_{\Theta^k} \partial_\varepsilon \left(\mathcal{J}\partial_\gamma h^k_\alpha \left(\partial_\varepsilon h^k_\beta \partial_\gamma h^k_\beta\right)^{-1}\right) \left(\varphi\psi^k\right) \circ \boldsymbol{h}^k\, d\boldsymbol{z}^k. \tag{2.20}$$

Next, we want to express the right hand side of equation (2.20) in terms of the curvature κ of Γ. However, before we can do this, we must find a way to calculate κ.

Curvature is defined by the negative of the differential of the trace of the Gaußian mapping, compare for example do Carmo [**7**] or Spivak [**74**]. We define the Gaußian mapping $\boldsymbol{\mathcal{N}}$ by $\boldsymbol{\mathcal{N}} = \boldsymbol{n} \circ \left(\boldsymbol{h}^k\right)^{-1}$, so if we further use D to represent the differential then curvature is given by

$$\kappa = -\operatorname{trace} D\boldsymbol{\mathcal{N}}. \tag{2.21}$$

Since the trace of an operator is invariant under basis transformation we can now represent $D\boldsymbol{\mathcal{N}}$ in the basis $\left\{\partial_\gamma \boldsymbol{h}^k\right\}$ of $\mathcal{T}_{\boldsymbol{y}}\Gamma$, where $\mathcal{T}_{\boldsymbol{y}}\Gamma$ denotes the tangent space at $\boldsymbol{y} \in \Gamma$, and calculate κ as the trace of the resulting matrix $\boldsymbol{A}^k$.

The above definition of $\boldsymbol{\mathcal{N}}$ yields $\boldsymbol{\mathcal{N}} \circ \boldsymbol{h}^k = \boldsymbol{n}$ which implies

$$D\boldsymbol{\mathcal{N}} D\boldsymbol{h}^k = D\boldsymbol{n} \qquad \text{and therefore,} \qquad D\boldsymbol{\mathcal{N}} = D\boldsymbol{n} D\left(\boldsymbol{h}^k\right)^{-1}. \tag{2.22}$$

Furthermore, from $\boldsymbol{n}\boldsymbol{n} = 1$ we conclude $D\left(\boldsymbol{n}\boldsymbol{n}\right) = \boldsymbol{0}$ and therefore, $\boldsymbol{n}D\boldsymbol{n} = \boldsymbol{0}$. This means $\boldsymbol{n}$ is perpendicular to each column of $D\boldsymbol{n} \in \mathbb{R}^{d\times(d-1)}$ and therefore, each column of $D\boldsymbol{n}$ is tangential to Γ. Thus we know that locally, we have $D\boldsymbol{n} : \mathcal{T}_{\boldsymbol{z}^k}\Theta^k \to \mathcal{T}_{\boldsymbol{y}}\Gamma$, where $\mathcal{T}_{\boldsymbol{z}^k}\Theta^k$ represents the tangent space at $\boldsymbol{z}^k \in \Theta^k$, and since we also know that $D\boldsymbol{h}^k : \mathcal{T}_{\boldsymbol{z}^k}\Theta^k \to \mathcal{T}_{\boldsymbol{y}}\Gamma$ we find that $D\boldsymbol{\mathcal{N}} : \mathcal{T}_{\boldsymbol{y}}\Gamma \to \mathcal{T}_{\boldsymbol{y}}\Gamma$. Now, we deduce the representation of $D\boldsymbol{\mathcal{N}}$ in the basis $\left\{\partial_\varepsilon \boldsymbol{h}^k\right\}$ of Γ, i. e. we determine $\boldsymbol{A}^k \in \mathbb{R}^{(d-1)\times(d-1)}$ such that

$$D\boldsymbol{\mathcal{N}} = D\boldsymbol{h}^k \boldsymbol{A}^k D\left(\boldsymbol{h}^k\right)^{-1}.$$

Multiplying from the right with $D\boldsymbol{h}^k$ and using (2.22) we find

$$D\boldsymbol{n} = D\boldsymbol{h}^k \boldsymbol{A}^k.$$

In component notation, this equation has the form

$$\partial_\gamma n_\alpha = \partial_\varepsilon h^k_\alpha A^k_{\varepsilon\gamma}.$$

Multiplying this from the left with $\partial_\zeta h^k_\alpha$, where $\zeta = 1, \ldots, d-1$, and then isolating $A^k_{\varepsilon\gamma}$ we gain

$$A^k_{\varepsilon\gamma} = \left(\partial_\varepsilon h^k_\beta \partial_\zeta h^k_\beta\right)^{-1} \partial_\zeta h^k_\alpha \partial_\gamma n_\alpha.$$

Using now (2.17) we finally end up with

$$A^k_{\varepsilon\gamma} = -\left(\partial_\varepsilon h^k_\beta \partial_\zeta h^k_\beta\right)^{-1} n_\alpha \partial_{\zeta\gamma} h^k_\alpha. \tag{2.23}$$

In the basis $\left\{\partial_\varepsilon \boldsymbol{h}^k\right\}$, definition 2.21 reduces to

$$\kappa = -A^k_{\varepsilon\varepsilon}, \tag{2.24}$$

so we have

$$\kappa = \left(\partial_\varepsilon h^k_\beta \partial_\zeta h^k_\beta\right)^{-1} n_\alpha \partial_{\zeta\varepsilon} h^k_\alpha.$$

At this stage, we are able to derive a representation of the right hand side of equation (2.20) in terms of κ. For this purpose, we assume without loss of generality that locally, the $\mathcal{C}^2$-surface Γ can be interpreted as the graph of a function $\tilde{h}^k : \Theta^k \to \mathbb{R}$, i. e. we can write

$$h^k_\beta(\boldsymbol{z}^k) = \begin{cases} z^k_\beta & \text{for } \beta = 1, \dots, d-1 \\ \tilde{h}^k(\boldsymbol{z}^k) & \text{for } \beta = d \end{cases}$$

for each $\boldsymbol{z}^k \in \Theta^k$. In this situation, we have

$$\partial_\varepsilon h^k_\beta = \begin{cases} \delta_{\varepsilon\beta} & \text{for } \beta = 1, \dots, d-1 \\ \partial_\varepsilon \tilde{h}^k & \text{for } \beta = d \end{cases} \tag{2.25}$$

as well as

$$\mathcal{J} = \left(1 + \partial_\gamma \tilde{h}^k \partial_\gamma \tilde{h}^k\right)^{1/2},$$

so we obtain from (2.16) and $n_\alpha n_\alpha = 1$

$$n_\beta = \begin{cases} \partial_\beta \tilde{h}^k / \mathcal{J} & \text{for } \beta = 1, \dots, d-1 \\ 1/\mathcal{J} & \text{for } \beta = d. \end{cases} \tag{2.26}$$

From (2.25), we gain

$$\partial_\varepsilon h^k_\beta \partial_\gamma h^k_\beta = \delta_{\varepsilon\gamma} + \partial_\varepsilon \tilde{h}^k \partial_\gamma \tilde{h}^k$$

which yields

$$\left(\partial_\varepsilon h^k_\beta \partial_\gamma h^k_\beta\right)^{-1} = \delta_{\varepsilon\gamma} - \frac{1}{\mathcal{J}^2} \partial_\varepsilon \tilde{h}^k \partial_\gamma \tilde{h}^k \tag{2.27}$$

and therefore, using again equation (2.25) we have

$$\partial_\gamma h^k_\alpha \left(\partial_\varepsilon h^k_\beta \partial_\gamma h^k_\beta\right)^{-1} = \begin{cases} \delta_{\varepsilon\alpha} - \partial_\varepsilon \tilde{h}^k \partial_\alpha \tilde{h}^k / \mathcal{J}^2 & \text{for } \alpha = 1, \dots, d-1 \\ \partial_\varepsilon \tilde{h}^k / \mathcal{J}^2 & \text{for } \alpha = d. \end{cases} \tag{2.28}$$

Additionally, we conclude from (2.25)

$$\partial_{\varepsilon\gamma} h^k_\beta = \delta_{\beta d} \partial_{\varepsilon\gamma} \tilde{h}^k$$

and since we know from (2.26) that $n_d = -1/\mathcal{J}$ we have

$$n_\beta \partial_{\varepsilon\gamma} h^k_\beta = n_\beta \delta_{\beta d} \partial_{\varepsilon\gamma} \tilde{h}^k = -\frac{\partial_{\varepsilon\gamma} \tilde{h}^k}{\mathcal{J}}.$$

Inserting now (2.27) and the above result into (2.23) we obtain

$$A_{\varepsilon\gamma} = \left(\delta_{\varepsilon\zeta} - \frac{1}{\mathcal{J}^2} \partial_\varepsilon \tilde{h}^k \partial_\zeta \tilde{h}^k\right) \frac{\partial_{\zeta\gamma} \tilde{h}^k}{\mathcal{J}}$$

and using (2.24) we end up with

$$\kappa = -\left(\frac{\partial_{\varepsilon\varepsilon} \tilde{h}^k}{\mathcal{J}} - \frac{1}{\mathcal{J}^3} \partial_{\varepsilon\zeta} \tilde{h}^k \partial_\varepsilon \tilde{h}^k \partial_\zeta \tilde{h}^k\right).$$

Due to (2.28), we know

$$\mathcal{J} \partial_\gamma h^k_\alpha \left(\partial_\varepsilon h^k_\beta \partial_\gamma h^k_\beta\right)^{-1} = \begin{cases} \mathcal{J} \delta_{\varepsilon\alpha} - \partial_\varepsilon \tilde{h}^k \partial_\alpha \tilde{h}^k / \mathcal{J} & \text{for } \alpha = 1, \dots, d-1 \\ \partial_\varepsilon \tilde{h}^k / \mathcal{J} & \text{for } \alpha = d. \end{cases} \tag{2.29}$$

Now, we want to calculate $\partial_\varepsilon \left(\mathcal{J} \partial_\gamma h^k_\alpha \left(\partial_\varepsilon h^k_\beta \partial_\gamma h^k_\beta \right)^{-1} \right)$ for later use. Noting that

$$\partial_\varepsilon \mathcal{J} = \frac{1}{\mathcal{J}} \partial_{\zeta\varepsilon} \tilde{h}^k \partial_\zeta \tilde{h}^k \qquad \text{and} \qquad \partial_\varepsilon \frac{1}{\mathcal{J}} = -\frac{1}{\mathcal{J}^3} \partial_{\varepsilon\zeta} \tilde{h}^k \partial_\zeta \tilde{h}^k$$

we compute

$$\partial_\varepsilon \frac{\partial_\varepsilon \tilde{h}^k}{\mathcal{J}} = \left(\frac{\partial_{\varepsilon\varepsilon} \tilde{h}^k}{\mathcal{J}} - \frac{1}{\mathcal{J}^3} \partial_{\zeta\varepsilon} \tilde{h}^k \partial_\varepsilon \tilde{h}^k \partial_\zeta \tilde{h}^k \right) = -\kappa$$

and

$$\partial_\varepsilon \left(\mathcal{J} \delta_{\varepsilon\alpha} - \frac{1}{\mathcal{J}} \partial_\varepsilon \tilde{h}^k \partial_\alpha \tilde{h}^k \right) = \frac{\partial_{\alpha\zeta} \tilde{h}^k}{\mathcal{J}} \partial_\zeta \tilde{h}^k - \partial_\varepsilon \left(\frac{\partial_\varepsilon \tilde{h}^k}{\mathcal{J}} \right) \partial_\alpha \tilde{h}^k - \frac{\partial_{\alpha\varepsilon} \tilde{h}^k}{\mathcal{J}} \partial_\varepsilon \tilde{h}^k = \kappa \partial_\alpha \tilde{h}^k.$$

Together with (2.26) and (2.29), those results lead to

$$\partial_\varepsilon \left(\mathcal{J} \partial_\gamma h^k_\alpha \left(\partial_\varepsilon h^k_\beta \partial_\gamma h^k_\beta \right)^{-1} \right) = \kappa n_\alpha \mathcal{J}$$

and comparing this with the right hand side of equation (2.20) we find that

$$\int_{\Theta^k} \partial_\varepsilon \left(\mathcal{J} \partial_\gamma h^k_\alpha \left(\partial_\varepsilon h^k_\beta \partial_\gamma h^k_\beta \right)^{-1} \right) (\varphi \psi^k) \circ \boldsymbol{h}^k \, d\boldsymbol{z}^k = \int_{\Theta^k} \kappa n_\alpha \left((\varphi \psi^k) \circ \boldsymbol{h}^k \right) \mathcal{J} \, d\boldsymbol{z}^k. \tag{2.30}$$

Transforming now $\Theta^k \ni \boldsymbol{z} \to \boldsymbol{y} \in \mathcal{U}^k$ and summing over k we get

$$\sum_k \left(\int_{\Theta^k} \kappa n_\alpha \left((\varphi \psi^k) \circ \boldsymbol{h}^k \right) \mathcal{J} \, d\boldsymbol{z}^k \right) = \int_\Gamma \kappa n_\alpha \varphi \, d\mathcal{H}.$$

Finally, we apply again (2.11) to obtain

$$\int_\Gamma \kappa n_\alpha \varphi \, d\mathcal{H} = \langle \kappa n_\alpha \delta_\Gamma, \varphi \rangle_\Omega.$$

Summarising all the results derived so far in this Section we have a rigorous proof of (2.15) and therefore, we showed

Theorem 2.2. *Suppose the assumptions collected in Section 2.1.1 hold and Γ is a $(d-1)$-dimensional $\mathcal{C}^2$-surface in Ω. Then, in $\mathcal{D}'(\Omega)$, the two-phase incompressible Navier-Stokes equations* (2.12) *can be written in the form*

$$\partial_\alpha u_\alpha = 0 \qquad \text{and} \qquad \partial_t(\varrho u_\alpha) + \partial_\beta \left(\varrho u_\alpha u_\beta - \sigma(\delta_{\alpha\beta} - n_\alpha n_\beta) \delta_{\Gamma(t)} \right) = \partial_\beta \tau_{\alpha\beta}. \tag{2.31}$$

Let us conclude this Section with a short excursion. Suppose for a smooth function $g : \Gamma \to \mathbb{R}$ we define the surface derivative $\boldsymbol{\partial}_{\mathcal{S}} g$ by

$$(\partial_{\mathcal{S}} g)_\alpha = \partial_\varepsilon h^k_\alpha \left(\partial_\gamma h^k_\beta \partial_\varepsilon h^k_\beta \right)^{-1} \partial_\gamma \left(g \circ \boldsymbol{h}^k \right)$$

and the Laplace-Beltrami operator $\partial^2_{\mathcal{S}}$ by

$$\partial^2_{\mathcal{S}} g = \frac{1}{\mathcal{J}} \partial_\varepsilon \left(\mathcal{J} \partial_\gamma \left(g \circ \boldsymbol{h}^k \right) \left(\partial_\varepsilon h^k_\beta \partial_\gamma h^k_\beta \right)^{-1} \right).$$

Then, we find that after a transformation of coordinates, the integrals in equation (2.30) can be rewritten in the form

$$\int_{\Theta^k} \partial_\varepsilon \left(\mathcal{J} \partial_\gamma h^k_\alpha \left(\partial_\varepsilon h^k_\beta \partial_\gamma h^k_\beta \right)^{-1} \right) \left((\varphi \psi^k) \circ \boldsymbol{h}^k \right) d\boldsymbol{z}^k = \int_\Gamma \partial^2_{\mathcal{S}} id_{\Gamma;\alpha} \left(\varphi \psi^k \right) d\mathcal{H},$$

where $\boldsymbol{id}_\Gamma$ denotes identity on Γ, respectively

$$\int_{\Theta^k} \kappa n_\alpha \left((\varphi \psi^k) \circ \boldsymbol{h}^k \right) \mathcal{J} \, d\boldsymbol{z}^k = \int_\Gamma \kappa n_\alpha \left(\varphi \psi^k \right) d\mathcal{H}.$$

Therefore, summing over k we obtain

$$\int_\Gamma \partial^2_{\mathcal{S}} id_{\Gamma;\alpha} \varphi \, d\mathcal{H} = \int_\Gamma \kappa n_\alpha \varphi \, d\mathcal{H}$$

and thus, we see that it is also possible to define the curvature κ by $\kappa \boldsymbol{n} = \partial^2_{\mathcal{S}} \boldsymbol{id}_\Gamma$, as it is done in Bänsch [**2**], instead of using (2.21).

2.2. Immiscible lattice BGK

In this Section, we present an LBGK model of two-phase flow which is consistent to the full-space incompressible Navier-Stokes equations (2.31). To the best of the author's knowledge, there are presently three competitive approaches to simulating two-phase flow with LBGK, namely the immiscible LBGK (ILBGK) method described below, the interacting potential method [**70, 71, 72, 73**], and the free energy method [**34, 58, 59, 76, 77**]. Each of those approaches has its respective drawbacks, see Hou et al. [**35**] and Luo [**51, 52**]. We settle on ILBGK because it is the only technique that allows to separate the treatment of surface tension effects from interface tracking. Furthermore, it is a common drawback of the interacting potential method and the free energy approach that in general flow, it is not possible to explicitly fix the coefficient of surface tension a priori.

Simulating two-phase flow we must solve the transport equation (2.13) together with the two-phase Navier-Stokes equations (2.31). Note that both density ϱ and stress tensor $\tau_{\alpha\beta}$ are likely to jump across the interface Γ. The possibility of jumps on Γ is a very crucial point because the appearance of discontinuities does not coincide with the consistency analysis presented in Section 1.1.2. However, here we will bypass this problem by requiring that neither ϱ nor $\tau_{\alpha\beta}$ jumps across the interface.

The basic ideas of immiscible lattice BGK date back to the immiscible lattice gas model published in 1988 by Rothman and Keller [**66**]. The immiscible lattice Boltzmann method in two space dimensions was introduced in 1991 by Gunstensen et al. [**28**], extended to three space dimensions by Gunstensen and Rothman [**27**] in 1992, and further improved by Grunau et al. [**26**] in 1993. In the latter paper, immiscible lattice BGK was introduced by adopting the ideas of Chen et al. [**9**] and Qian et al. [**63**] to the immiscible lattice Boltzmann model. Note that ILBGK is a very simple form of immiscible lattice Boltzmann.

We shall describe the classical formulation of immiscible lattice BGK in Section 2.2.1 and present a more reasonable variation in Section 2.2.2.

2.2.1. Original method. We consider again the situation described in Section 2.1.1. As in one-phase LBGK, we assume that Ω is covered by a regular lattice with equidistant nodes and again, we denote the set of lattice nodes by $\mathcal{X}$. At a given time t, we consider particle densities $f^r(t,\boldsymbol{x};\boldsymbol{v})$ and $f^b(t,\boldsymbol{x};\boldsymbol{v})$ of red respectively blue fluid particles located at position $\boldsymbol{x}\in\mathcal{X}$ and moving with velocity $\boldsymbol{v}\in\mathcal{V}_9=\{\boldsymbol{c}_j\,:\,j=0,\dots,8\}$.

Recalling the polynomials (1.12) we can express red and blue mass density in the form

$$\rho^l=\left\langle f^l,Q_0\right\rangle_{\boldsymbol{v}}\qquad\text{for } l=r,b. \tag{2.32}$$

Full (or uncoloured) particle density f is defined by

$$f(t,\boldsymbol{x};\boldsymbol{v})=f^r(t,\boldsymbol{x};\boldsymbol{v})+f^b(t,\boldsymbol{x};\boldsymbol{v}). \tag{2.33}$$

Therefore, full mass density ρ and uncoloured momentum density $\rho\boldsymbol{u}$ are given by

$$\rho=\langle f,Q_0\rangle_{\boldsymbol{v}}\qquad\text{respectively}\qquad\rho u_1=\langle f,Q_1\rangle_{\boldsymbol{v}}\qquad\text{and}\qquad\rho u_2=\langle f,Q_2\rangle_{\boldsymbol{v}}\,. \tag{2.34}$$

In analogy to one-phase LBGK, we determine the flow velocity $\boldsymbol{u}$ by $u_\alpha=\rho u_\alpha/\rho$.

Coloured particle density is governed by the lattice BGK equation

$$f^l(t+\Delta t,\boldsymbol{x};\boldsymbol{v})=(1-\omega)f^l(t,\boldsymbol{x}-\boldsymbol{v}\Delta x;\boldsymbol{v})+\omega f_S^{l,eq}\left(\rho^l(t,\boldsymbol{x}-\boldsymbol{v}\Delta x),\boldsymbol{u}(t,\boldsymbol{x}-\boldsymbol{v}\Delta x);\boldsymbol{v}\right) \tag{2.35}$$

with equilibrium particle density

$$f_S^{l,eq}\left(\rho^l,\boldsymbol{u};\boldsymbol{v}\right)=f^*(\boldsymbol{v})\left(\rho^l+3\Delta x\rho^l u_\alpha v_\alpha+\frac{9\Delta x^2}{2}\rho^l u_\alpha u_\beta\left(v_\alpha v_\beta-\frac{\delta_{\alpha\beta}}{3}\right)\right)+\frac{1}{2}S(\boldsymbol{v}), \tag{2.36}$$

where the relaxation parameter ω is assumed to be constant in Ω. The term $S(\boldsymbol{v})$ in (2.36) models the effects of surface tension and will be discussed in detail in Section 2.4.2. Note here that S fulfils $\langle S,Q_0\rangle_{\boldsymbol{v}}=0$ and therefore,

$$\rho^l=\left\langle f_S^{l,eq},Q_0\right\rangle_{\boldsymbol{v}}.$$

Together with (2.32), this yields mass conservation for both phases in immiscible lattice BGK.

In analogy to one-phase LBGK, the evolution equation (2.35) is typically solved by conducting first the collision step

$$\tilde{f}^l(t,\boldsymbol{x};\boldsymbol{v})=(1-\omega)f^l(t,\boldsymbol{x};\boldsymbol{v})+\omega f_S^{l,eq}(\rho^l(t,\boldsymbol{x}),\boldsymbol{u}(t,\boldsymbol{x});\boldsymbol{v}) \tag{2.37}$$

and then the propagation step

$$f^l(t+\Delta t, \boldsymbol{x}; \boldsymbol{v}) = \tilde{f}^l(t, \boldsymbol{x} - \boldsymbol{v}\Delta x; \boldsymbol{v}), \tag{2.38}$$

compare (1.6) and (1.7). Note that due to

$$\rho^l = \left\langle f^l, Q_0 \right\rangle_{\boldsymbol{v}} = \left\langle f_S^{l,eq}, Q_0 \right\rangle_{\boldsymbol{v}},$$

ρ^l is not affected by the collision step.

To prevent both phases from mixing with each other, we apply a third step between collision and propagation, the so-called recolouring step. The basic idea of recolouring is to keep the interface sharp by reallocating $\tilde{f}^l(t, \boldsymbol{x}; \boldsymbol{v})$ in the vicinity of $\Gamma(t)$ such that as few coloured mass as possible crosses the interface in the propagation step. To achieve this goal, we first have to compute the local colour gradient which is for $\boldsymbol{x} \in \mathcal{X}$ typically defined by

$$\hat{\boldsymbol{F}}(t, \boldsymbol{x}) = \frac{1}{\Delta x} \sum_j \boldsymbol{c}_j \Big(\rho^r(t, \boldsymbol{x} + \boldsymbol{c}_j \Delta x) - \rho^b(t, \boldsymbol{x} + \boldsymbol{c}_j \Delta x) \Big). \tag{2.39}$$

Then, we seek particle densities $\mathcal{R}(t, \boldsymbol{x}; \boldsymbol{v})$ and $\mathcal{B}(t, \boldsymbol{x}; \boldsymbol{v})$ such that

$$\hat{F}_1 \left\langle \mathcal{R} - \mathcal{B}, Q_1 \right\rangle_{\boldsymbol{v}} + \hat{F}_2 \left\langle \mathcal{R} - \mathcal{B}, Q_2 \right\rangle_{\boldsymbol{v}} \to \max \tag{2.40a}$$

under the constraints

$$\left\langle \mathcal{R}, Q_0 \right\rangle_{\boldsymbol{v}} = \rho^r, \qquad \left\langle \mathcal{B}, Q_0 \right\rangle_{\boldsymbol{v}} = \rho^b, \qquad \text{and} \qquad \mathcal{R} + \mathcal{B} = f. \tag{2.40b}$$

Note that (2.40a) maximises the projection of the difference between the coloured momentum vectors $(\left\langle \mathcal{R}, Q_1 \right\rangle_{\boldsymbol{v}}, \left\langle \mathcal{R}, Q_2 \right\rangle_{\boldsymbol{v}})^T$ and $(\left\langle \mathcal{B}, Q_1 \right\rangle_{\boldsymbol{v}}, \left\langle \mathcal{B}, Q_2 \right\rangle_{\boldsymbol{v}})^T$ into the direction of the local colour gradient $\hat{\boldsymbol{F}}$ and that (2.40b) ensures conservation of coloured and uncoloured mass as well as uncoloured momentum. Note also that this procedure has an effect only if the colour gradient is non-zero.

Let us now rewrite the constrained maximisation problem (2.40) in a form which is more convenient for implementation. For this purpose, we insert the last constraint in (2.40b) into (2.40a) to obtain

$$2 \left(\hat{F}_1 \left\langle \mathcal{R}, Q_1 \right\rangle_{\boldsymbol{v}} + \hat{F}_2 \left\langle \mathcal{R}, Q_2 \right\rangle_{\boldsymbol{v}} \right) - \left(\hat{F}_1 \left\langle f, Q_1 \right\rangle_{\boldsymbol{v}} + \hat{F}_2 \left\langle f, Q_2 \right\rangle_{\boldsymbol{v}} \right) \to \max.$$

Since f is conserved during the recolouring step, $\hat{F}_1 \left\langle f, Q_1 \right\rangle_{\boldsymbol{v}} + \hat{F}_2 \left\langle f, Q_2 \right\rangle_{\boldsymbol{v}}$ is also conserved. For this reason, we may require

$$\hat{F}_1 \left\langle \mathcal{R}, Q_1 \right\rangle_{\boldsymbol{v}} + \hat{F}_2 \left\langle \mathcal{R}, Q_2 \right\rangle_{\boldsymbol{v}} \to \max$$

instead of (2.40a), as it is done for example in Grunau et al. [**26**] or Nie et al. [**57**].

Furthermore, it is possible to drop the second constraint in (2.40b), as we will show in the following. We consider the zeroth order discrete moment of equation (2.33),

$$\left\langle f^r, Q_0 \right\rangle_{\boldsymbol{v}} + \left\langle f^b, Q_0 \right\rangle_{\boldsymbol{v}} = \left\langle f, Q_0 \right\rangle_{\boldsymbol{v}},$$

and due to (2.32) and (2.34), this yields $\rho^r + \rho^b = \rho$. From the first and the third constraint in (2.40b), we now obtain

$$\left\langle \mathcal{R}, Q_0 \right\rangle_{\boldsymbol{v}} + \rho^b = \left\langle \mathcal{R} + \mathcal{B}, Q_0 \right\rangle_{\boldsymbol{v}},$$

so we just deduced $\rho^b = \left\langle \mathcal{B}, Q_0 \right\rangle_{\boldsymbol{v}}$ which therefore needs not be required as a constraint. Thus, (2.40) reduces to

$$\hat{F}_1 \left\langle \mathcal{R}, Q_1 \right\rangle_{\boldsymbol{v}} + \hat{F}_2 \left\langle \mathcal{R}, Q_2 \right\rangle_{\boldsymbol{v}} \to \max \qquad \text{such that} \qquad \left\langle \mathcal{R}, Q_0 \right\rangle_{\boldsymbol{v}} = \rho^r \qquad \text{and} \qquad \mathcal{R} + \mathcal{B} = f. \tag{2.41}$$

The resulting treatment of the maximisation problem is summarised in Algorithm 2.1, the recolouring step is described in Algorithm 2.2, and the complete ILBGK method is presented in Algorithm 2.3.

```
compute F̂_α(t,x) c_{j;α} for each j;
list the results in descending order;
let dens = 0;
while the list is nonempty and dens < ρ^r(t,x)
    assuming that F̂_α(t,x) c_{j̃;α} is the first element of the list
        let inc = min{ρ^r(t,x) − dens, f(t,x,c_j̃)};
        let R(t,x;c_j̃) = inc;
        let B(t,x;c_j̃) = f(t,x,c_j̃) − inc;
        let dens = dens + inc;
        remove F̂_α(t,x) c_{j̃;α} from the list;
```

Algorithm 2.1: Typical implementation of the constrained maximisation problem (2.41)

```
compute F̂(t,x) according to (2.39);
use Algorithm 2.1 to find R(t,x;v) and B(t,x;v) such that (2.41) is fulfilled;
let f̃^r(t,x,v) = R(t,x;v);
let f̃^b(t,x,v) = B(t,x;v);
```

Algorithm 2.2: The original recolouring step

```
initialise ρ^l(0,x) and u(0,x);
let f^l(0,x;v) = f_S^{l,eq}(ρ^r(0,x), u(0,x)) as defined by (2.36);
while t < t_max
    determine f_S^{l,eq} according to (2.36);
    do collision according to (2.37);
    do recolouring according to Algorithm 2.2;
    do propagation according to (2.38);
    let t = t + Δt;
    compute ρ^l(t,x) according to (2.32);
    compute f(t,x;v) according to (2.33);
    compute ρ(t,x) and ρu_α(t,x) according to (2.34);
    let u_α(t,x) = ρu_α(t,x)/ρ(t,x);
```

Algorithm 2.3: The immiscible lattice BGK method

2.2.2. Reformulation. In the following, we will present a very convenient reformulation of ILBGK which was introduced by Ginzburg and Steiner [**22, 23**]. Instead of solving LBGK equations for red and blue particle densities and computing uncoloured particle density according to (2.33), we now solve the LBGK equation for uncoloured particle density,

$$f(t+\Delta t,\boldsymbol{x};\boldsymbol{v}) = (1-\omega)f(t,\boldsymbol{x}-\boldsymbol{v}\Delta x;\boldsymbol{v}) + \omega f_S^{eq}\left(\rho(t,\boldsymbol{x}-\boldsymbol{v}\Delta x),\boldsymbol{u}(t,\boldsymbol{x}-\boldsymbol{v}\Delta x);\boldsymbol{v}\right), \tag{2.42}$$

with equilibrium particle density

$$f_S^{eq}\left(\rho,\boldsymbol{u};\boldsymbol{v}\right) = f^*(\boldsymbol{v})\left(\rho + 3\Delta x\rho u_\alpha v_\alpha + \frac{9\Delta x^2}{2}\rho u_\alpha u_\beta\left(v_\alpha v_\beta - \frac{\delta_{\alpha\beta}}{3}\right)\right) + S(\boldsymbol{v}) \tag{2.43}$$

along with equation (2.35) for red particle density. As usual, we divide (2.42) into the collision step

$$\tilde{f}(t,\boldsymbol{x};\boldsymbol{v}) = (1-\omega)f(t,\boldsymbol{x};\boldsymbol{v}) + \omega f_S^{eq}\left(\rho(t,\boldsymbol{x}),\boldsymbol{u}(t,\boldsymbol{x});\boldsymbol{v}\right) \tag{2.44}$$

and the propagation step

$$f(t+\Delta t,\boldsymbol{x};\boldsymbol{v}) = \tilde{f}(t,\boldsymbol{x}-\boldsymbol{v}\Delta x;\boldsymbol{v}). \tag{2.45}$$

Blue particle density is finally given by $f^b = f - f^r$.

We will now show that also the recolouring step can be formulated without explicitly considering the blue phase. For this purpose, we must replace the colour gradient $\hat{\boldsymbol{F}}$ with a quantity that can be computed without knowledge of ρ^b. In order to find a proper replacement, we must find out what $\boldsymbol{F}$ is consistent to.

For this purpose, we shall proceed as follows. We know that lattice BGK computes no functions in the strict sense but only discrete values at the lattice points $\boldsymbol{x} \in \mathcal{X}$. However, we suppose there are smooth Δx-dependent functions interpolating those values. We further assume those interpolating functions to converge in a suitable sense towards limit functions for $\Delta x \to 0$. Finally, we say the scheme is consistent to a given system of continuous equations if those equations are fulfilled by the limit functions for $\Delta x \to 0$. Remark that this concept applies not only to lattice BGK but to any finite difference-based numerical scheme.

In particular, we assume for $l = r, b$ there are functions $\rho^l_{\Delta x} \in \mathcal{C}^1([0, t_{max}] \times \Omega, \mathbb{R})$ such that $\rho^l_{\Delta x}(t, \boldsymbol{x}) = \rho^l(t, \boldsymbol{x})$ for each $t \in [0, t_{max}]$ and $\boldsymbol{x} \in \mathcal{X}$ and we define

$$\rho^+_{\Delta x} = \rho^r_{\Delta x} + \rho^b_{\Delta x} \qquad \text{as well as} \qquad \rho^-_{\Delta x} = \rho^r_{\Delta x} - \rho^b_{\Delta x}.$$

Therefore, the local colour gradient $\hat{\boldsymbol{F}}$ takes the form

$$\hat{\boldsymbol{F}}_{\Delta x}(t, \boldsymbol{x}) = \frac{1}{\Delta x} \sum_j \boldsymbol{c}_j \rho^-_{\Delta x}(t, \boldsymbol{x} + \boldsymbol{c}_j \Delta x).$$

Additionally, we assume

$$\lim_{\Delta x \to 0} \left\langle \rho^l_{\Delta x}, \varphi \right\rangle_\Omega = \left\langle \rho^l_0, \varphi \right\rangle_\Omega \tag{2.46a}$$

for any test function $\varphi \in \mathcal{D}'(\Omega)$ and inspired by (1.17), we suppose

$$\rho^r_0(t, \boldsymbol{x}) = \begin{cases} \varrho^r & \text{for } \boldsymbol{x} \in \Omega^r(t) \\ 0 & \text{for } \boldsymbol{x} \in \Omega^b(t) \end{cases} \qquad \text{and} \qquad \rho^b_0(t, \boldsymbol{x}) = \begin{cases} 0 & \text{for } \boldsymbol{x} \in \Omega^r(t) \\ \varrho^b & \text{for } \boldsymbol{x} \in \Omega^b(t), \end{cases} \tag{2.46b}$$

where ϱ^r and ϱ^b denote the constant density of red respectively blue incompressible fluid. This implies

$$\rho^+_0(t, \boldsymbol{x}) = \begin{cases} \varrho^r & \text{for } \boldsymbol{x} \in \Omega^r(t) \\ \varrho^b & \text{for } \boldsymbol{x} \in \Omega^b(t) \end{cases} \qquad \text{as well as} \qquad \rho^-_0(t, \boldsymbol{x}) = \begin{cases} \varrho^r & \text{for } \boldsymbol{x} \in \Omega^r(t) \\ -\varrho^b & \text{for } \boldsymbol{x} \in \Omega^b(t). \end{cases} \tag{2.46c}$$

Let us now start analysing the consistency of the local colour gradient $\hat{\boldsymbol{F}}_{\Delta x}$. First, we state

$$\left\langle \hat{F}_{\Delta x;\alpha}(t, \boldsymbol{x}), \varphi(\boldsymbol{x}) \right\rangle_\Omega = \left\langle \frac{1}{\Delta x} \sum_j c_{j;\alpha} \rho^-_{\Delta x}(t, \boldsymbol{x} + \boldsymbol{c}_j \Delta x), \varphi(\boldsymbol{x}) \right\rangle_\Omega, \tag{2.47}$$

where $\varphi \in \mathcal{D}(\Omega)$ is an arbitrary test function. Since $\rho^-_{\Delta x}$ is a regular distribution we have

$$\left\langle \frac{1}{\Delta x} \sum_j c_{j;\alpha} \rho^-_{\Delta x}(t, \boldsymbol{x} + \boldsymbol{c}_j \Delta x), \varphi(\boldsymbol{x}) \right\rangle_\Omega = \frac{1}{\Delta x} \sum_j c_{j;\alpha} \int_\Omega \rho^-_{\Delta x}(t, \boldsymbol{x} + \boldsymbol{c}_j \Delta x) \varphi(\boldsymbol{x}) \, d\boldsymbol{x},$$

and because φ is of compact support in Ω we know that

$$\frac{1}{\Delta x} \sum_j c_{j;\alpha} \int_\Omega \rho^-_{\Delta x}(t, \boldsymbol{x} + \boldsymbol{c}_j \Delta x) \varphi(\boldsymbol{x}) \, d\boldsymbol{x} = \frac{1}{\Delta x} \sum_j c_{j;\alpha} \int_{\operatorname{supp} \varphi} \rho^-_{\Delta x}(t, \boldsymbol{x} + \boldsymbol{c}_j \Delta x) \varphi(\boldsymbol{x}) \, d\boldsymbol{x}.$$

The coordinate transformation $\boldsymbol{x} + \boldsymbol{c}_j \Delta x \mapsto \boldsymbol{y}$ yields

$$\frac{1}{\Delta x} \sum_j c_{j;\alpha} \int_{\operatorname{supp} \varphi} \rho^-_{\Delta x}(t, \boldsymbol{x} + \boldsymbol{c}_j \Delta x) \varphi(\boldsymbol{x}) \, d\boldsymbol{x} = \frac{1}{\Delta x} \sum_j c_{j;\alpha} \int_{\operatorname{supp} \varphi + \boldsymbol{c}_j \Delta x} \rho^-_{\Delta x}(t, \boldsymbol{y}) \varphi(\boldsymbol{y} - \boldsymbol{c}_j \Delta x) \, d\boldsymbol{y},$$

where $\operatorname{supp} \varphi + \boldsymbol{c}_j \Delta x = \{\boldsymbol{y} : \boldsymbol{y} - \boldsymbol{c}_j \Delta x \in \operatorname{supp} \varphi\}$. Since Ω is open and $\operatorname{supp} \varphi$ is compact in Ω, we find that also $\operatorname{supp} \varphi + \boldsymbol{c}_j \Delta x$ is compact in Ω if Δx is chosen small enough. Thus,

$$\frac{1}{\Delta x} \sum_j c_{j;\alpha} \int_{\operatorname{supp} \varphi + \boldsymbol{c}_j \Delta x} \rho^-_{\Delta x}(t, \boldsymbol{y}) \varphi(\boldsymbol{y} - \boldsymbol{c}_j \Delta x) \, d\boldsymbol{y} = \frac{1}{\Delta x} \sum_j c_{j;\alpha} \int_\Omega \rho^-_{\Delta x}(t, \boldsymbol{y}) \varphi(\boldsymbol{y} - \boldsymbol{c}_j \Delta x) \, d\boldsymbol{y}.$$

At this stage, we obtain by Taylor expansion

$$\varphi(\boldsymbol{y} - \boldsymbol{c}_j \Delta x) = \varphi(\boldsymbol{y}) - \Delta x c_{j;\beta} \partial_\beta \varphi(\boldsymbol{y}) + \frac{\Delta x^2}{2} c_{j;\beta} c_{j;\gamma} \partial_{\beta\gamma} \varphi(\boldsymbol{y}) + \mathcal{O}\left(\Delta x^3\right)$$

and therefore,

$$\frac{1}{\Delta x}\sum_j c_{j;\alpha}\int_\Omega \rho^-_{\Delta x}(t,\boldsymbol{y})\varphi(\boldsymbol{y}-\boldsymbol{c}_j\Delta x)\,d\boldsymbol{y} = \frac{1}{\Delta x}\sum_j c_{j;\alpha}\int_\Omega \rho^-_{\Delta x}(t,\boldsymbol{y})\varphi(\boldsymbol{y})\,d\boldsymbol{y}$$
$$-\sum_j c_{j;\alpha}c_{j;\beta}\int_\Omega \rho^-_{\Delta x}(t,\boldsymbol{y})\partial_\beta\varphi(\boldsymbol{y})\,d\boldsymbol{y} + \frac{\Delta x}{2}\sum_j c_{j;\alpha}c_{j;\beta}c_{j;\gamma}\int_\Omega \rho^-_{\Delta x}(t,\boldsymbol{y})\partial_{\beta\gamma}\varphi(\boldsymbol{y})\,d\boldsymbol{y} + \mathcal{O}\left(\Delta x^2\right),$$

where the first and third term on the right hand side equal zero because $\sum_j c_{j;\alpha} = 0$ respectively $\sum_j c_{j;\alpha}c_{j;\beta}c_{j;\gamma} = 0$. By virtue of (2.46), the second term on the right hand side fulfils

$$\lim_{\Delta x\to 0}\left(-\sum_j c_{j;\alpha}c_{j;\beta}\int_\Omega \rho^-_{\Delta x}(t,\boldsymbol{y})\partial_\beta\varphi(\boldsymbol{y})\,d\boldsymbol{y}\right) = -\sum_j c_{j;\alpha}c_{j;\beta}\int_\Omega \rho^-_0(t,\boldsymbol{y})\partial_\beta\varphi(\boldsymbol{y})\,d\boldsymbol{y}.$$

Observing

$$\sum_j c_{j;\alpha}c_{j;\beta} = 6\delta_{\alpha\beta} \qquad \text{and} \qquad \rho^-_0(t,\boldsymbol{y}) = \left(\varrho^r+\varrho^b\right)\chi(t,\boldsymbol{y}) - \varrho^b,$$

where χ is given by (2.1), we gain

$$-\sum_j c_{j;\alpha}c_{j;\beta}\int_\Omega \rho^-_0(t,\boldsymbol{y})\partial_\beta\varphi(\boldsymbol{y})\,d\boldsymbol{y} = -6\left(\varrho^r+\varrho^b\right)\int_\Omega \chi(t,\boldsymbol{y})\partial_\alpha\varphi(\boldsymbol{y})\,d\boldsymbol{y} + 6\varrho^b\int_\Omega \partial_\alpha\varphi(\boldsymbol{y})\,d\boldsymbol{y}. \tag{2.48}$$

Due to the divergence theorem [**16**],

$$6\varrho^b\int_\Omega \partial_\alpha\varphi(\boldsymbol{y})\,d\boldsymbol{y} = 0,$$

so the right hand side of (2.48) can be rewritten in the form

$$-6\left(\varrho^r+\varrho^b\right)\int_\Omega \chi(t,\boldsymbol{y})\partial_\alpha\varphi(\boldsymbol{y})\,d\boldsymbol{y} = 6\left(\varrho^r+\varrho^b\right)\langle\partial_\alpha\chi,\varphi\rangle_\Omega,$$

and applying now (2.14) we obtain

$$6\left(\varrho^r+\varrho^b\right)\langle\partial_\alpha\chi,\varphi\rangle_\Omega = 6\left(\varrho^r+\varrho^b\right)\langle n_\alpha\delta_\Gamma,\varphi\rangle_\Omega$$

for an arbitrary test function $\varphi\in\mathcal{D}(\Omega)$. Thus, we can formulate

Proposition 2.1. *In $\mathcal{D}'(\Omega)$,*

$$\lim_{\Delta x\to 0}\hat{F}_{\Delta x;\alpha} = 6\left(\varrho^r+\varrho^b\right)n_\alpha\delta_\Gamma.$$

The colour gradient $\hat{\boldsymbol{F}}_{\Delta x}$ is needed only for the maximisation procedure (2.41) and it converges for $\Delta x\to 0$ to a vector which is parallel to $\boldsymbol{n}$. However, the term which is maximised in (2.41) can be interpreted as the Euclidean scalar product of the vectors $\hat{\boldsymbol{F}}_{\Delta x}$ and $\left(\langle\mathcal{R},Q_1\rangle_{\boldsymbol{v}},\langle\mathcal{R},Q_2\rangle_{\boldsymbol{v}}\right)^T$, where the latter represents coloured momentum of the red phase. Due to the fact that the Euclidean scalar product of two vectors in $\mathbb{R}^d$ is maximal if those vectors point into the same direction, minimal if they point into opposite directions, and zero if they are perpendicular to each other, the maximisation procedure (2.41) moves for $\Delta x\to 0$ as much red mass as possible into the direction parallel to $\hat{\boldsymbol{F}}_{\Delta x}$, i. e. the direction parallel to $\boldsymbol{n}$. However, this behaviour does not change if we replace $\hat{\boldsymbol{F}}_{\Delta x}$ with any other vector whose limit for $\Delta x\to 0$ points into the same direction. Inspired by (2.14) we choose

$$\boldsymbol{F}_{\Delta x}(t,\boldsymbol{x}) = \frac{1}{6\Delta x}\sum_j \boldsymbol{c}_j\chi_{\Delta x}(t,\boldsymbol{x}+\boldsymbol{c}_j\Delta x), \tag{2.49}$$

where $\chi_{\Delta x}\in\mathcal{C}^1\left([0,t_{max}]\times\Omega,\mathbb{R}\right)$ and $\lim_{\Delta x\to 0}\chi_{\Delta x} = \chi$ with χ defined by (2.1), and then, we repeat the arguments above to show

Proposition 2.2. *In $\mathcal{D}'(\Omega)$,*

$$\lim_{\Delta x\to 0}F_{\Delta x;\alpha} = n_\alpha\delta_\Gamma.$$

```
compute F_α(t,x) c_{j;α} for each j;
list the results in descending order;
let dens = 0;
while the list is nonempty and dens < ρ^r(t,x)
    assuming F_α(t,x) c_{j̃;α} is the first element of the list
        let inc = min {ρ^r(t,x) − dens, f(t,x,c_j̃)};
        let R(t,x;c_j̃) = inc;
        let dens = dens + inc;
        remove F_α(t,x) c_{j̃;α} from the list;
```

Algorithm 2.4: Implementation of the constrained maximisation problem (2.51)

```
compute F(t,x) according to (2.49);
use Algorithm 2.4 to find R(t,x;v) such that (2.51) is fulfilled;
let f̃^r(t,x;v) = R(t,x;v);
```

Algorithm 2.5: The reformulated recolouring step

Let us remark that from definition (2.49), we obtain

$$F_{\Delta x;1}(t,\boldsymbol{x}) = \frac{1}{6\Delta x}\begin{bmatrix}-1 & 0 & 1\\ -1 & 0 & 1\\ -1 & 0 & 1\end{bmatrix}\chi_{\Delta x}(t,\boldsymbol{x}) = \partial_1\chi_{\Delta x} + \mathcal{O}\left(\Delta x^2\right) \qquad \text{in } \mathcal{D}'(\Omega) \tag{2.50a}$$

as well as

$$F_{\Delta x;2}(t,\boldsymbol{x}) = \frac{1}{6\Delta x}\begin{bmatrix}1 & 1 & 1\\ 0 & 0 & 0\\ -1 & -1 & -1\end{bmatrix}\chi_{\Delta x}(t,\boldsymbol{x}) = \partial_2\chi_{\Delta x} + \mathcal{O}\left(\Delta x^2\right) \qquad \text{in } \mathcal{D}'(\Omega). \tag{2.50b}$$

It is also possible to prove Proposition 2.2 using some other stencils for computing the local colour gradient as long as they are consistent to the components of the gradient. For example,

$$\frac{1}{12\Delta x}\begin{bmatrix}-1 & 0 & 1\\ -4 & 0 & 4\\ -1 & 0 & 1\end{bmatrix} \leftrightarrow \partial_1 + \mathcal{O}\left(\Delta x^2\right) \qquad \text{and} \qquad \frac{1}{12\Delta x}\begin{bmatrix}1 & 4 & 1\\ 0 & 0 & 0\\ -1 & -4 & -1\end{bmatrix} \leftrightarrow \partial_2 + \mathcal{O}\left(\Delta x^2\right)$$

are a reasonable choice. However, from now on, we will exclusively work with the colour gradient $\boldsymbol{F}_{\Delta x}$ defined by (2.49). Furthermore, to avoid complicated notation, we will from now on skip the Δx-subscripts as long as we do not perform a consistency analysis for $\Delta x \to 0$. Thus, the constrained maximisation problem (2.41) takes the form

$$F_1\left\langle \mathcal{R}, Q_1\right\rangle_{\boldsymbol{v}} + F_2\left\langle \mathcal{R}, Q_2\right\rangle_{\boldsymbol{v}} \to \max \qquad \text{such that} \qquad \left\langle \mathcal{R}, Q_0\right\rangle_{\boldsymbol{v}} = \rho^r \qquad \text{and} \qquad \mathcal{R} \le f. \tag{2.51}$$

For this reason, the maximisation Algorithm 2.1 reduces to Algorithm 2.4 and recolouring itself takes the form presented in Algorithm 2.5.

Next, we observe that red particle density is completely reorganised by the recolouring Algorithm 2.5 and therefore, the preceding collision step for the red phase is obsolete. Furthermore, combining (2.32) and (2.38) we find that

$$\rho^r(t+\Delta t,\boldsymbol{x}) = \left\langle \tilde{f}^r(t,\boldsymbol{x}-\boldsymbol{v}\Delta x;\boldsymbol{v}), Q_0(\boldsymbol{v})\right\rangle_{\boldsymbol{v}}, \tag{2.52}$$

i. e. red density can be updated directly from the output of the recolouring step. Altogether, we obtain the ILBGK Algorithm 2.6.

Finally, we realize red particle density is needed only for computing ρ^r. Since $\langle S, Q_0\rangle_{\boldsymbol{v}}=0$ the surface tension term $S(\boldsymbol{v})$ does not influence (2.52). Therefore, we remove it from the definition of red equilibrium

```
initialise ρ(0,x), ρ^r(0,x), and u(0,x);
let f(0,x;v) = f_S^eq(ρ(0,x), u(0,x)) as defined by (2.43);
while t < t_max
   determine f_S^eq(ρ(t,x), u(t,x)) according to (2.43);
   do uncoloured collision according to (2.44);
   do recolouring according to Algorithm 2.5;
   compute ρ^r(t + Δt, x) according to (2.52);
   do uncoloured propagation according to (2.45);
   let t = t + Δt;
   compute ρ(t,x) and ρu_α(t,x) according to (2.34);
   let u_α(t,x) = ρu_α(t,x)/ρ(t,x);
```

Algorithm 2.6: Reformulation of the immiscible lattice BGK method

particle density and use

$$f^{r,eq}\left(\rho^r, \boldsymbol{u}; \boldsymbol{v}\right) = f^*(\boldsymbol{v})\left(\rho^r + 3\Delta x \rho^r u_\alpha v_\alpha + \frac{9\Delta x^2}{2}\rho^r u_\alpha u_\beta\left(v_\alpha v_\beta - \frac{\delta_{\alpha\beta}}{3}\right)\right) \tag{2.53}$$

instead of $f_S^{r,eq}$ for the theoretical investigations to be performed in subsequent Sections.

2.3. Interface tracking in immiscible lattice BGK

In this Section, we want to investigate the intrinsic interface tracking scheme of immiscible lattice BGK. Again, we will assume that both ϱ and $\tau_{\alpha\beta}$ are continuous in Ω.

First, we will perform an equivalent moment analysis to learn which part of immiscible lattice BGK is responsible for interface tracking (Section 2.3.1) and after this, we shall isolate the interface tracking portion and consider it separately (Section 2.3.2). Finally, we will numerically investigate the isolated ILBGK interface tracking scheme (Section 2.3.3).

2.3.1. Equivalent moment analysis. Let us now investigate how interface tracking is managed in ILBGK and why it is necessary to recolour particle densities. For this purpose, we first consider the evolution equation (2.35) for the red phase with equilibrium particle density $f^{r,eq}$ as defined by (2.53) without any recolouring and apply the equivalent moment analysis introduced in Section 1.1.2.

Recalling the polynomials (1.12) we define

$$M_k^r = \langle f^r, Q_k\rangle_{\boldsymbol{v}} \qquad \text{and} \qquad M_k^{r,eq} = \langle f^{r,eq}, Q_k\rangle_{\boldsymbol{v}} \qquad \text{for } k = 0,\dots,8$$

which yields

$$M_0^{r,eq} = \rho^r, \qquad M_1^{r,eq} = \rho^r u_1, \qquad M_2^{r,eq} = \rho^r u_2, \qquad M_3^{r,eq} = \rho^r u_1^2, \qquad M_4^{r,eq} = \rho^r u_1 u_2,$$

$$M_5^{r,eq} = \rho^r u_2^2, \qquad M_6^{r,eq} = 0, \qquad M_7^{r,eq} = 0, \qquad M_8^{r,eq} = 0.$$

Proceeding now in analogy to what we did in Section 1.1.2 we find the equivalent moment system of equation (2.35) for $l = r$ to take the form

$$M_k^r(t+\Delta t, \boldsymbol{x}) = \sum_{j=0}^{8} \frac{1}{\langle f^*, Q_j^2\rangle_{\boldsymbol{v}}}\left\langle f^*, Q_k Q_j\left((1-\omega)M_j^r + \omega M_j^{r,eq}\right)(t, \boldsymbol{x} - \boldsymbol{v}\Delta x)\right\rangle_{\boldsymbol{v}},$$

compare (1.14), and therefore, due to coloured mass conservation we obtain for $k = 0$

$$\begin{aligned}\rho^r(t+\Delta t,\boldsymbol{x}) = \langle f^*, Q_0\rho^r(t,\boldsymbol{x}-\boldsymbol{v}\Delta x)\rangle_{\boldsymbol{v}} &+ \Big\langle f^*, 3\Delta x^2 Q_1\Big((1-\omega)M_1^r + \omega\left(\rho^r u_1\right)\Big)(t,\boldsymbol{x}-\boldsymbol{v}\Delta x)\Big\rangle_{\boldsymbol{v}} \\ &+ \Big\langle f^*, 3\Delta x^2 Q_2\Big((1-\omega)M_2^r + \omega\left(\rho^r u_2\right)\Big)(t,\boldsymbol{x}-\boldsymbol{v}\Delta x)\Big\rangle_{\boldsymbol{v}} \\ &+ \Big\langle f^*, \frac{9\Delta x^4}{2} Q_3\Big((1-\omega)M_3^r + \omega\rho^r u_1^2\Big)(t,\boldsymbol{x}-\boldsymbol{v}\Delta x)\Big\rangle_{\boldsymbol{v}} \\ &+ \Big\langle f^*, 9\Delta x^4 Q_4\Big((1-\omega)M_4^r + \omega\rho^r u_1 u_2\Big)(t,\boldsymbol{x}-\boldsymbol{v}\Delta x)\Big\rangle_{\boldsymbol{v}} \\ + \Big\langle f^*, \frac{9\Delta x^4}{2} Q_5\Big((1-\omega)M_5^r + \omega\rho^r u_2^2\Big)(t,\boldsymbol{x}-\boldsymbol{v}\Delta x)\Big\rangle_{\boldsymbol{v}} &+ \Big\langle f^*, \frac{3\Delta x^6}{2} Q_6(1-\omega)M_6^r(t,\boldsymbol{x}-\boldsymbol{v}\Delta x)\Big\rangle_{\boldsymbol{v}} \\ + \Big\langle f^*, \frac{3\Delta x^6}{2} Q_7(1-\omega)M_7^r(t,\boldsymbol{x}-\boldsymbol{v}\Delta x)\Big\rangle_{\boldsymbol{v}} &+ \Big\langle f^*, \frac{\Delta x^8}{16} Q_8(1-\omega)M_8^r(t,\boldsymbol{x}-\boldsymbol{v}\Delta x)\Big\rangle_{\boldsymbol{v}}. \end{aligned} \tag{2.54}$$

In those regions of Ω where $f^b = 0$, we have $f^r = f$ (compare (2.33)) which yields $M_1^r = \rho^r u_1$ as well as $M_2^r = \rho^r u_2$ and therefore, $u_1 = M_1^r/\rho^r$ as well as $u_2 = M_2^r/\rho^r$. However, in those regions where both f^r and f^b are positive we find that $f^r \neq f$ and therefore, $M_1^r \neq \rho^r u_1$ as well as $M_1^r \neq \rho^r u_1$. This implies $u_1 \neq M_1^r/\rho^r$ and $u_2 \neq M_2^r/\rho^r$. Thus, in regions of mixed phases, equation (2.54) forces the red phase to move with a velocity which is different from the actual flow velocity $\boldsymbol{u}$. Fortunately, ILBGK does not suffer from this effect because before propagation takes place, coloured particle densities are redistributed by the recolouring step.

Let us now avoid this problem by choosing $\omega = 1$ which turns (2.35) into

$$f^r(t+\Delta t,\boldsymbol{x};\boldsymbol{v}) = f^{r,eq}\left(\rho^r(t,\boldsymbol{x}-\boldsymbol{v}\Delta x),\boldsymbol{u}(t,\boldsymbol{x}-\boldsymbol{v}\Delta x);\boldsymbol{v}\right) \tag{2.55}$$

and (2.54) into

$$\begin{aligned}\rho^r(t+\Delta t,\boldsymbol{x}) = \langle f^*, Q_0\rho^r(t,\boldsymbol{x}-\boldsymbol{v}\Delta x)\rangle_{\boldsymbol{v}} &+ \left\langle f^*, 3\Delta x^2 Q_1\left(\rho^r u_1\right)(t,\boldsymbol{x}-\boldsymbol{v}\Delta x)\right\rangle_{\boldsymbol{v}} \\ + \left\langle f^*, 3\Delta x^2 Q_2\left(\rho^r u_2\right)(t,\boldsymbol{x}-\boldsymbol{v}\Delta x)\right\rangle_{\boldsymbol{v}} &+ \left\langle f^*, \frac{9\Delta x^4}{2} Q_3\left(\rho^r u_1^2\right)(t,\boldsymbol{x}-\boldsymbol{v}\Delta x)\right\rangle_{\boldsymbol{v}} \\ + \left\langle f^*, 9\Delta x^4 Q_4\left(\rho^r u_1 u_2\right)(t,\boldsymbol{x}-\boldsymbol{v}\Delta x)\right\rangle_{\boldsymbol{v}} &+ \left\langle f^*, \frac{9\Delta x^4}{2} Q_5\left(\rho^r u_2^2\right)(t,\boldsymbol{x}-\boldsymbol{v}\Delta x)\right\rangle_{\boldsymbol{v}}. \end{aligned} \tag{2.56}$$

The right hand side of this equation consists of expressions of the form (1.15) which can be interpreted as finite difference stencils. In particular, the polynomials

$$Q_0, \qquad 3\Delta x^2 Q_1, \qquad 3\Delta x^2 Q_2, \qquad \frac{9\Delta x^4}{2} Q_3, \qquad 9\Delta x^4 Q_4, \qquad \text{and} \qquad \frac{9\Delta x^4}{2} Q_5$$

produce the stencils

$$Q_0 \leftrightarrow \frac{1}{36}\begin{bmatrix}1 & 4 & 1\\ 4 & 16 & 4\\ 1 & 4 & 1\end{bmatrix} = 1 + \frac{1}{36}\begin{bmatrix}1 & 4 & 1\\ 4 & -20 & 4\\ 1 & 4 & 1\end{bmatrix} \leftrightarrow 1 + \frac{\Delta x^2}{6}\partial_{\alpha\alpha} + \mathcal{O}\left(\Delta x^4\right),$$

$$3\Delta x^2 Q_1 \leftrightarrow -\frac{\Delta x}{12}\begin{bmatrix}-1 & 0 & 1\\ -4 & 0 & 4\\ -1 & 0 & 1\end{bmatrix} \leftrightarrow \Delta x^2\partial_1 + \mathcal{O}\left(\Delta x^4\right),$$

$$3\Delta x^2 Q_2 \leftrightarrow -\frac{\Delta x}{12}\begin{bmatrix}1 & 4 & 1\\ 0 & 0 & 0\\ -1 & -4 & -1\end{bmatrix} \leftrightarrow \Delta x^2\partial_2 + \mathcal{O}\left(\Delta x^4\right),$$

$$\frac{9\Delta x^4}{2} Q_3 \leftrightarrow \frac{\Delta x^2}{12}\begin{bmatrix}1 & -2 & 1\\ 4 & -8 & 4\\ 1 & -2 & 1\end{bmatrix} \leftrightarrow \frac{\Delta x^4}{2}\partial_{11} + \mathcal{O}\left(\Delta x^6\right),$$

$$9\Delta x^4 Q_4 \leftrightarrow \frac{\Delta x^2}{4}\begin{bmatrix}-1 & 0 & 1\\ 0 & 0 & 0\\ 1 & 0 & -1\end{bmatrix} \leftrightarrow \Delta x^4 \partial_{12} + \mathcal{O}\left(\Delta x^6\right),$$

respectively

$$\frac{9\Delta x^4}{2} Q_5 \leftrightarrow \frac{\Delta x^2}{12}\begin{bmatrix}1 & 4 & 1\\ -2 & -8 & -2\\ 1 & 4 & 1\end{bmatrix} \leftrightarrow \frac{\Delta x^4}{2}\partial_{22} + \mathcal{O}\left(\Delta x^6\right).$$

Therefore, (2.56) is equivalent to

$$\begin{aligned}
&\rho^r(t+\Delta t,\boldsymbol{x}) = \rho^r(t,\boldsymbol{x})\\
&\quad + \frac{1}{36}\begin{bmatrix}1 & 4 & 1\\ 4 & -20 & 4\\ 1 & 4 & 1\end{bmatrix}\rho^r(t,\boldsymbol{x}) - \frac{\Delta x}{12}\begin{bmatrix}-1 & 0 & 1\\ -4 & 0 & 4\\ -1 & 0 & 1\end{bmatrix}\left(\rho^r u_1\right)(t,\boldsymbol{x}) - \frac{\Delta x}{12}\begin{bmatrix}1 & 4 & 1\\ 0 & 0 & 0\\ -1 & -4 & -1\end{bmatrix}\left(\rho^r u_2\right)(t,\boldsymbol{x})\\
&\qquad + \frac{\Delta x^2}{12}\begin{bmatrix}1 & -2 & 1\\ 4 & -8 & 4\\ 1 & -2 & 1\end{bmatrix}\left(\rho^r u_1^2\right)(t,\boldsymbol{x}) + \frac{\Delta x^2}{4}\begin{bmatrix}-1 & 0 & 1\\ 0 & 0 & 0\\ 1 & 0 & -1\end{bmatrix}\left(\rho^r u_1 u_2\right)(t,\boldsymbol{x})\\
&\qquad\qquad + \frac{\Delta x^2}{12}\begin{bmatrix}1 & 4 & 1\\ -2 & -8 & -2\\ 1 & 4 & 1\end{bmatrix}\left(\rho^r u_1^2\right)(t,\boldsymbol{x})
\end{aligned}$$

and thus,

$$\begin{aligned}
&\langle \rho^r(t+\Delta t,\boldsymbol{x}),\varphi(\boldsymbol{x})\rangle_\Omega = \langle \rho^r(t,\boldsymbol{x}),\varphi(\boldsymbol{x})\rangle_\Omega\\
&\quad + \frac{1}{36}\left\langle \begin{bmatrix}1 & 4 & 1\\ 4 & -20 & 4\\ 1 & 4 & 1\end{bmatrix}\rho^r(t,\boldsymbol{x}),\varphi(\boldsymbol{x})\right\rangle_\Omega - \frac{\Delta x}{12}\left\langle \begin{bmatrix}-1 & 0 & 1\\ -4 & 0 & 4\\ -1 & 0 & 1\end{bmatrix}\left(\rho^r u_1\right)(t,\boldsymbol{x}),\varphi(\boldsymbol{x})\right\rangle_\Omega\\
&\quad - \frac{\Delta x}{12}\left\langle \begin{bmatrix}1 & 4 & 1\\ 0 & 0 & 0\\ -1 & -4 & -1\end{bmatrix}\left(\rho^r u_2\right)(t,\boldsymbol{x}),\varphi(\boldsymbol{x})\right\rangle_\Omega + \frac{\Delta x^2}{12}\left\langle \begin{bmatrix}1 & -2 & 1\\ 4 & -8 & 4\\ 1 & -2 & 1\end{bmatrix}\left(\rho^r u_1^2\right)(t,\boldsymbol{x}),\varphi(\boldsymbol{x})\right\rangle_\Omega\\
&\quad + \frac{\Delta x^2}{4}\left\langle \begin{bmatrix}-1 & 0 & 1\\ 0 & 0 & 0\\ 1 & 0 & -1\end{bmatrix}\left(\rho^r u_1 u_2\right)(t,\boldsymbol{x}),\varphi(\boldsymbol{x})\right\rangle_\Omega + \frac{\Delta x^2}{12}\left\langle \begin{bmatrix}1 & 4 & 1\\ -2 & -8 & -2\\ 1 & 4 & 1\end{bmatrix}\left(\rho^r u_1^2\right)(t,\boldsymbol{x}),\varphi(\boldsymbol{x})\right\rangle_\Omega
\end{aligned}$$

for any test function $\varphi \in \mathcal{D}(\Omega)$. Adjoining the finite difference stencils we obtain

$$\begin{aligned}
&\langle \rho^r(t+\Delta t,\boldsymbol{x}),\varphi(\boldsymbol{x})\rangle_\Omega = \langle \rho^r(t,\boldsymbol{x}),\varphi(\boldsymbol{x})\rangle_\Omega\\
&\quad + \frac{1}{36}\left\langle \rho^r(t,\boldsymbol{x}),\begin{bmatrix}1 & 4 & 1\\ 4 & -20 & 4\\ 1 & 4 & 1\end{bmatrix}\varphi(\boldsymbol{x})\right\rangle_\Omega - \frac{\Delta x}{12}\left\langle \left(\rho^r u_1\right)(t,\boldsymbol{x}),\begin{bmatrix}1 & 0 & -1\\ 4 & 0 & -4\\ 1 & 0 & -1\end{bmatrix}\varphi(\boldsymbol{x})\right\rangle_\Omega\\
&\quad - \frac{\Delta x}{12}\left\langle \left(\rho^r u_2\right)(t,\boldsymbol{x}),\begin{bmatrix}-1 & -4 & -1\\ 0 & 0 & 0\\ 1 & 4 & 1\end{bmatrix}\varphi(\boldsymbol{x})\right\rangle_\Omega + \frac{\Delta x^2}{12}\left\langle \left(\rho^r u_1^2\right)(t,\boldsymbol{x}),\begin{bmatrix}1 & -2 & 1\\ 4 & -8 & 4\\ 1 & -2 & 1\end{bmatrix}\varphi(\boldsymbol{x})\right\rangle_\Omega\\
&\quad + \frac{\Delta x^2}{4}\left\langle \left(\rho^r u_1 u_2\right)(t,\boldsymbol{x}),\begin{bmatrix}-1 & 0 & 1\\ 0 & 0 & 0\\ 1 & 0 & -1\end{bmatrix}\varphi(\boldsymbol{x})\right\rangle_\Omega + \frac{\Delta x^2}{12}\left\langle \left(\rho^r u_1^2\right)(t,\boldsymbol{x}),\begin{bmatrix}1 & 4 & 1\\ -2 & -8 & -2\\ 1 & 4 & 1\end{bmatrix}\varphi(\boldsymbol{x})\right\rangle_\Omega
\end{aligned}$$

and Taylor expansion yields

$$\begin{aligned}\langle\rho^r(t+\Delta t,\boldsymbol{x}),\varphi(\boldsymbol{x})\rangle_\Omega &= \langle\rho^r(t,\boldsymbol{x}),\varphi(\boldsymbol{x})\rangle_\Omega + \frac{\Delta x^2}{6}\langle\rho^r(t,\boldsymbol{x}),\partial_{\alpha\alpha}\varphi(\boldsymbol{x})\rangle_\Omega + \Delta x^2\langle(\rho^r u_1)(t,\boldsymbol{x}),\partial_1\varphi(\boldsymbol{x})\rangle_\Omega \\ &+ \Delta x^2\langle(\rho^r u_2)(t,\boldsymbol{x}),\partial_2\varphi(\boldsymbol{x})\rangle_\Omega + \frac{\Delta x^4}{2}\left\langle\left(\rho^r u_1^2\right)(t,\boldsymbol{x}),\partial_{11}\varphi(\boldsymbol{x})\right\rangle_\Omega + \Delta x^4\langle(\rho^r u_1u_2)(t,\boldsymbol{x}),\partial_{12}\varphi(\boldsymbol{x})\rangle_\Omega \\ &+ \frac{\Delta x^4}{2}\left\langle\left(\rho^r u_1^2\right)(t,\boldsymbol{x}),\partial_{22}\varphi(\boldsymbol{x})\right\rangle_\Omega + \mathcal{O}\left(\Delta x^4\right)\end{aligned}$$

which is equivalent to

$$\begin{aligned}\langle\rho^r(t+\Delta t,\boldsymbol{x}),\varphi(\boldsymbol{x})\rangle_\Omega &= \langle\rho^r(t,\boldsymbol{x}),\varphi(\boldsymbol{x})\rangle_\Omega + \frac{\Delta x^2}{6}\langle\rho^r(t,\boldsymbol{x}),\partial_{\alpha\alpha}\varphi(\boldsymbol{x})\rangle_\Omega \\ &+ \Delta x^2\langle(\rho^r u_1)(t,\boldsymbol{x}),\partial_1\varphi(\boldsymbol{x})\rangle_\Omega + \Delta x^2\langle(\rho^r u_2)(t,\boldsymbol{x}),\partial_2\varphi(\boldsymbol{x})\rangle_\Omega + \mathcal{O}\left(\Delta x^4\right).\end{aligned}$$

Using now $\Delta t = \Delta x^2$ we obtain

$$\begin{aligned}\left\langle\frac{\rho^r(t+\Delta t,\boldsymbol{x}-\rho^r(t,\boldsymbol{x})}{\Delta t},\varphi(\boldsymbol{x})\right\rangle_\Omega &- \langle(\rho^r u_1)(t,\boldsymbol{x}),\partial_1\varphi(\boldsymbol{x})\rangle_\Omega - \langle(\rho^r u_2)(t,\boldsymbol{x}),\partial_2\varphi(\boldsymbol{x})\rangle_\Omega \\ &= \frac{1}{6}\langle\rho^r(t,\boldsymbol{x}),\partial_{\alpha\alpha}\varphi(\boldsymbol{x})\rangle_\Omega + \mathcal{O}\left(\Delta x^2\right)\end{aligned}$$

and Taylor expanding the first term on the left hand side provides

$$\begin{aligned}\langle\partial_t\rho^r(t,\boldsymbol{x}),\varphi(\boldsymbol{x})\rangle_\Omega - \langle(\rho^r u_1)(t,\boldsymbol{x}),\partial_1\varphi(\boldsymbol{x})\rangle_\Omega - \langle(\rho^r u_2)(t,\boldsymbol{x}),\partial_2\varphi(\boldsymbol{x})\rangle_\Omega &= \frac{1}{6}\langle\rho^r(t,\boldsymbol{x}),\partial_{\alpha\alpha}\varphi(\boldsymbol{x})\rangle_\Omega \\ &+ \mathcal{O}\left(\Delta x^2\right)\end{aligned}$$

for each test function $\varphi \in \mathcal{D}(\Omega)$. By definition, this is equivalent to

$$\partial_t\rho^r + \partial_\alpha\left(\rho^r u_\alpha\right) = \frac{1}{6}\partial_{\alpha\alpha}\rho^r + \mathcal{O}\left(\Delta x^2\right) \tag{2.57}$$

in $\mathcal{D}'(\Omega)$. Note that we cannot apply an analogue to (1.17) here because in the vicinity of the interface, the gradient of ρ^r is of $\mathcal{O}(1/\Delta x)$. Therefore, the viscous term on the right hand side of (2.57) must be explicitly dealt with.

Neglecting the $\mathcal{O}\left(\Delta x^2\right)$ term on the right hand side we find that equation (2.57) combines the effects of the transport equation

$$\partial_t\rho^r + \partial_\alpha\left(\rho^r u_\alpha\right) = 0 \tag{2.58}$$

and the diffusion equation

$$\partial_t\rho^r = \frac{1}{6}\partial_{\alpha\alpha}\rho^r. \tag{2.59}$$

(This behaviour can be illustrated by considering the case of a constant velocity field $\boldsymbol{u}^{(const)}$, where the split scheme

$$\hat{\rho}^r(t+\Delta t,\boldsymbol{x}) = \tilde{\rho}^r(t,\boldsymbol{x}) + \frac{\Delta t}{6}\partial_{\alpha\alpha}\tilde{\rho}^r(t,\boldsymbol{x}), \qquad \tilde{\rho}^r(t,\boldsymbol{x}) = \hat{\rho}^r(t,\boldsymbol{x}) - \Delta t u_\alpha^{(const)}\partial_\alpha\hat{\rho}^r(t,\boldsymbol{x})$$

with initial condition $\hat{\rho}^r(0,\boldsymbol{x}) = \rho^r(0,\boldsymbol{x})$ for each $x \in \mathcal{X}$ determines exactly the same approximation to the solution ρ^r of the convection-diffusion equation

$$\partial_t\rho^r + u_\alpha^{(const)}\partial_\alpha\rho^r = \frac{1}{6}\partial_{\alpha\alpha}\rho^r$$

as the unsplit scheme

$$\hat{\rho}^r(t+\Delta t,\boldsymbol{x}) = \hat{\rho}^r(t,\boldsymbol{x}) + \Delta t\left(\frac{1}{6}\partial_{\alpha\alpha}\hat{\rho}^r(t,\boldsymbol{x}) - u_\alpha^{(const)}\partial_\alpha\hat{\rho}^r(t,\boldsymbol{x})\right)$$

with the same initial condition.) Interpreting now ρ^r as a non-normalised indicator function for the red phase we find (2.58) to be in some sense a non-normalised variant of the transport equation (2.13) and, therefore, to be responsible for interface tracking. Equation (2.59), however, causes the phases to diffuse into each other and thus, it is responsible for the smearing of the interface. Altogether, we find that (2.57) effectuates both tracking and smearing of the interface. The latter is another reason for the necessity of the recolouring step.

```
compute F_α(t,x) c_{j;α} for each j;
list the results in descending order;
let dens = 0;
while the list is nonempty and dens < χ(t,x)
    assuming F_α(t,x) c_{j̃;α} is the first element of the list
        let inc = min{χ(t,x) − dens, ℓ(u(t,x), c_j̃);
        let ℛ(t,x; c_j̃) = inc;
        let dens = dens + inc;
        remove F_α(t,x) c_{j̃;α} from the list;
```

Algorithm 2.7: Typical implementation of the maximisation problem (2.65)

2.3.2. Isolation. To study the intrinsic ILBGK interface tracking scheme, it is most convenient to separate interface tracking from the remainder of ILBGK. For this purpose, we will now put up a stand-alone ILBGK interface tracking method.

Recalling the derivation of (2.57) from (2.55) we find that the nonlinear term

$$f^*(\boldsymbol{v})\frac{9\Delta x^2}{2}\rho^r u_\alpha u_\beta\left(v_\alpha v_\beta - \frac{\delta_{\alpha\beta}}{3}\right)$$

in the equilibrium distribution (2.53) contributes only to the error term in (2.57), so we decide to remove it. Furthermore, it seems most suitable to solve a transport equation for the normalised indicator function χ instead of ρ^r (Section 2.3.1). Therefore, we consider a pseudo-particle density R such that $\langle R, Q_0\rangle_{\boldsymbol{v}} = \chi$ and a corresponding equilibrium pseudo-particle density

$$R^{eq}(\chi, \boldsymbol{u}; \boldsymbol{v}) = f^*(\boldsymbol{v})\left(\chi + 3\Delta x \chi u_\alpha v_\alpha\right) \tag{2.60}$$

which obviously fulfils $\langle R^{eq}, Q_0\rangle_{\boldsymbol{v}} = \chi$. Then, we solve the LBGK equation

$$R\left(t+\Delta t, \boldsymbol{x}; \boldsymbol{v}\right) = R^{eq}\big(\chi(t, \boldsymbol{x} - \boldsymbol{v}\Delta x), \boldsymbol{u}(t, \boldsymbol{x} - \boldsymbol{v}\Delta x); \boldsymbol{v}\big) \tag{2.61}$$

whose zeroth order moment equation

$$\begin{aligned}\chi(t+\Delta t, \boldsymbol{x}) = \chi(t,\boldsymbol{x}) &+ \frac{1}{36}\begin{bmatrix}1 & 4 & 1\\ 4 & -20 & 4\\ 1 & 4 & 1\end{bmatrix}\chi(t,\boldsymbol{x})\\ &- \frac{\Delta x}{12}\begin{bmatrix}-1 & 0 & 1\\ -4 & 0 & 4\\ -1 & 0 & 1\end{bmatrix}(\chi u_1)(t,\boldsymbol{x}) - \frac{\Delta x}{12}\begin{bmatrix}1 & 4 & 1\\ 0 & 0 & 0\\ -1 & -4 & -1\end{bmatrix}(\chi u_2)(t,\boldsymbol{x})\end{aligned} \tag{2.62}$$

is in $\mathcal{D}'(\Omega)$ consistent to

$$\partial_t \chi + \partial_\alpha(\chi u_\alpha) = \frac{1}{6}\partial_{\alpha\alpha}\chi + \mathcal{O}\left(\Delta x^2\right). \tag{2.63}$$

Scheme (2.61) can be used independently of the actual ILBGK method. However, if we want to apply recolouring to this stand-alone scheme we have to construct a proper substitute for the uncoloured particle density f. Motivated by (2.60) and (2.61), we define

$$\ell\left(\boldsymbol{u}; \boldsymbol{v}\right) = f^*(\boldsymbol{v})\left(1 + 3\Delta x u_\alpha v_\alpha\right). \tag{2.64}$$

and replace (2.51) with

$$F_1\langle \mathcal{R}, Q_1\rangle_{\boldsymbol{v}} + F_2\langle \mathcal{R}, Q_2\rangle_{\boldsymbol{v}} \to \max \qquad \text{such that} \qquad \langle \mathcal{R}, Q_0\rangle_{\boldsymbol{v}} = \chi \qquad \text{and} \qquad \mathcal{R} \le \ell. \tag{2.65}$$

This results in the maximisation Algorithm 2.7 and the recolouring Algorithm 2.8. Note that ℓ plays in fact the role of uncoloured particle density in the sense of representing the total amount of mass that is sent into the direction of $\boldsymbol{v}$. After recolouring, we update χ according to

$$\chi(t+\Delta x, \boldsymbol{x}) = \langle \mathcal{R}(t, \boldsymbol{x} - \boldsymbol{v}\Delta x; \boldsymbol{v}), Q_0(\boldsymbol{v})\rangle_{\boldsymbol{v}}, \tag{2.66}$$

compare (2.52). The complete immiscible lattice BGK procedure for solving the transport equation (2.13) is summarised in Algorithm 2.9.

compute $\boldsymbol{F}(t,\boldsymbol{x})$ according to (2.49);
compute $\not{f}(\boldsymbol{u}(t,\boldsymbol{x});\boldsymbol{v})$ according to (2.64);
use Algorithm 2.7 to find $\mathcal{R}(t,\boldsymbol{x};\boldsymbol{v})$ such that (2.65) is fulfilled;

Algorithm 2.8: recolouring step for stand-alone ILBGK interface tracking

initialise $\rho^r(0,\boldsymbol{x})$ and $\boldsymbol{u}(0,\boldsymbol{x})$;
while $t \leq t_{max}$
  do recolouring according to Algorithm 2.8;
  compute $\chi(t+\Delta t,\boldsymbol{x})$ according to (2.66);
  let $t = t + \Delta t$;
  if necessary, update $\boldsymbol{u}(t,\boldsymbol{x})$ with any method;

Algorithm 2.9: ILBGK interface tracking

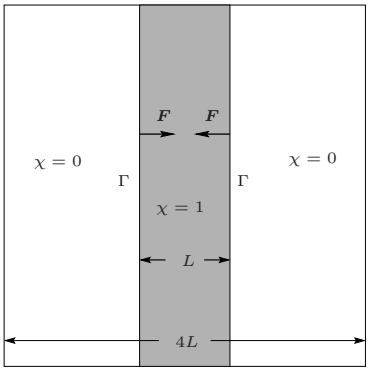


Figure 2.3: The initial situation of the model problem with straight interface Γ, $\chi = 1$ in a slice of width L and $\chi = 0$ everywhere else on $\Omega = [0, 4L] \times [0, 4L]$.

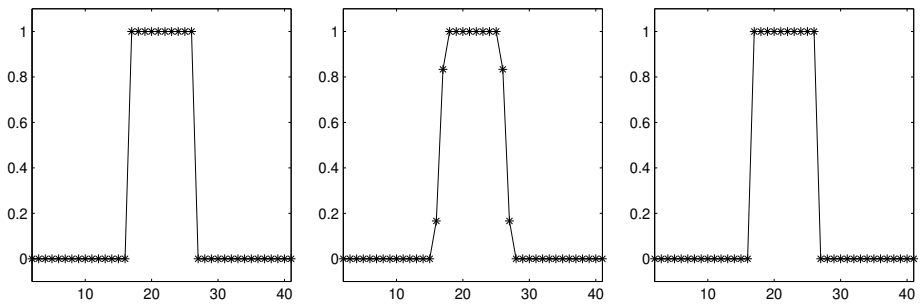


Figure 2.4: Initial state and the ILBGK solution of the model problem for $\boldsymbol{u}_\alpha(t,\boldsymbol{x}) = 0$. Each graphic shows a cut through the twentieth line of the 40×40 lattice. The pictures are taken at $t = 0$ (left), $t = \Delta t$ (middle), and $t = 2\Delta t$ (right).

2.3.3. Numerical investigation. Let us now investigate the interface tracking Algorithm 2.9 by means of a simple model problem. We consider a square domain $\Omega = [0, 4L] \times [0, 4L]$, where $\chi(0,\boldsymbol{x}) = 1$ on a slab of width L and $\chi(0\boldsymbol{x}) = 0$ everywhere else on Ω, as illustrated in Figure 2.3. For our convenience, we assume $\rho = 1$ in $[0, t_{max}] \times \Omega$. We further assume $\partial\Omega$ is periodic in each direction, so transport of the slab to the left or right is essentially a one-dimensional process.

Let us now perform some numerical experiments. For the beginning, we consider the static case, i. e. $u_\alpha(t,\boldsymbol{x}) = 0$ for each $t \in [0, t_{max}]$ and every $\boldsymbol{x} \in \Omega$. In this case, $\chi(t,\boldsymbol{x}) = \chi(0,\boldsymbol{x})$ for each $t \in [0, t_{max}]$. Nevertheless, ILBGK interface tracking exhibits diffusion during the first time step. However, diffusive effects are completely compensated at the end of the second time step, see Figure 2.4 for an illustration. This procedure repeats again and again until t_{max} is reached. To make clear where this behaviour comes from, we will now exemplify the performance of Algorithm 2.9 in the given situation.

For this purpose, we choose $\tilde{\boldsymbol{x}} \in \mathcal{X}$ such that $\chi(0, \tilde{\boldsymbol{x}}) = 1$ and $\chi(0, \tilde{\boldsymbol{x}} + \boldsymbol{c}_1 \Delta x) = 0$, for example $\tilde{\boldsymbol{x}} = \Delta x(26, 20)^T$ on a 40×40 lattice. Initialising the pseudo-particle density R with its equilibrium state we obtain from (2.60) and (2.64)

$$R(0, \tilde{\boldsymbol{x}}; \boldsymbol{v}) = \ell(\boldsymbol{u}(t, \tilde{\boldsymbol{x}}); \boldsymbol{v}) = f^*(\boldsymbol{v}), \quad R(0, \tilde{\boldsymbol{x}} + \boldsymbol{c}_1 \Delta x; \boldsymbol{v}) = 0, \quad \text{and} \quad \ell(\boldsymbol{u}(t, \tilde{\boldsymbol{x}} + \boldsymbol{c}_j \Delta x); \boldsymbol{v}) = f^*(\boldsymbol{v}).$$

According to Algorithm 2.9, we now have to apply the recolouring step. Following Algorithm 2.7, we find that $\mathcal{R}(0, \boldsymbol{x}; \boldsymbol{v}) = R(0, \boldsymbol{x}; \boldsymbol{v})$ for each $\boldsymbol{x} \in \Omega$. Propagating $\mathcal{R}$ according to (2.61) we obtain

$$R(\Delta t, \tilde{\boldsymbol{x}}; \boldsymbol{c}_j) = \begin{cases} 4/9, & \text{for } j = 0 \\ 1/9, & \text{for } j = 1, 3, 7 \\ 1/36, & \text{for } j = 2, 8 \\ 0, & \text{for } j = 4, 5, 6 \end{cases} \quad \text{and} \quad R(\Delta t, \tilde{\boldsymbol{x}} + \boldsymbol{c}_1 \Delta x; \boldsymbol{c}_j) = \begin{cases} 1/9, & \text{for } j = 1 \\ 1/36, & \text{for } j = 2, 8 \\ 0, & \text{for } j = 0, 3, 4, 5, 6, 7. \end{cases}$$

At this stage, the first time step is finished and we have

$$\chi(\Delta t, \tilde{\boldsymbol{x}}) = 5/6 \qquad \text{and} \qquad \chi(\Delta t, \tilde{\boldsymbol{x}} + \boldsymbol{c}_1 \Delta x) = 1/6.$$

Thus, during the first time step, one sixth of the red mass at $\tilde{\boldsymbol{x}}$ diffuses into the blue phase while one sixth of the blue mass at $\tilde{\boldsymbol{x}} + \boldsymbol{c}_1 \Delta x$ diffuses into the red phase. This coincides exactly with the diffusion term in equation (2.63).

The second time step starts again with the recolouring Algorithm 2.8. This time, we obtain

$$\mathcal{R}(\Delta t, \tilde{\boldsymbol{x}}; \boldsymbol{c}_j) = \begin{cases} 4/9, & \text{for } j = 0 \\ 1/9, & \text{for } j = 3, 5, 7 \\ 1/36, & \text{for } j = 4, 6 \\ 0, & \text{for } j = 1, 2, 8 \end{cases} \qquad \text{and} \qquad \mathcal{R}(\Delta t, \tilde{\boldsymbol{x}} + \boldsymbol{c}_1 \Delta x; \boldsymbol{c}_j) = \begin{cases} 1/9, & \text{for } j = 5 \\ 1/36, & \text{for } j = 4, 6 \\ 0, & \text{for } j = 0, 1, 2, 3, 7, 8 \end{cases}$$

which leads to

$$R(2\Delta t, \tilde{\boldsymbol{x}}; \boldsymbol{v}) = R(0, \tilde{\boldsymbol{x}}; \boldsymbol{v}) \qquad \text{and} \qquad R(2\Delta t, \tilde{\boldsymbol{x}} + \boldsymbol{c}_1 \Delta x; \boldsymbol{v}) = R(0, \tilde{\boldsymbol{x}} + \boldsymbol{c}_1 \Delta x; \boldsymbol{v})$$

after propagation. An analogous investigation of the second interface in the model problem, i. e. the jump from $\chi = 0$ to $\chi = 1$, reveals completely symmetric behaviour and away from the jumps in χ, Algorithm 2.9 shows no effect at all. Thus, we conclude $\chi(2\Delta t, \boldsymbol{x}) = \chi(0, \boldsymbol{x})$ throughout Ω.

Since the situation at the end of the second time step is identical to the initial condition we now deduce that the scene will repeat again and again if we continue the experiment. Therefore, we find that ILBGK interface tracking yields the correct solution if we run the code for an even number of time steps while the result is perturbed if we run the code for an odd number of time steps. Note, however, that red mass is always perfectly conserved in the sense that percental mass loss is below machine accuracy.

Let us now assume χ is moved to the right by the constant velocity field $\boldsymbol{u} = (0.25, 0)^T$, again on a 40×40 lattice. Figure 2.5 illustrates the initial setting and the first eight time steps of the ILBGK interface tracking Algorithm 2.9 in this case.

The correct solution corresponds to the red slice being moved one lattice point to the right in four time steps. We observe that this is nearly perfectly met, only the fronts are smeared out a little bit after the fourth time step. However, after eight time steps, we have exactly the same situation as after four time steps, so we conclude that the same scene will repeat again and again until the simulation stops. Thus, we know the smearing of the interface not to grow in time. At the end of the eighth time step, the loss of red mass equals $1.4211 \cdot 10^{-14}$ percent.

2.4. Modelling surface tension in immiscible lattice BGK

In the following, we shall examine how to include surface tension effects into immiscible lattice BGK, i. e. how to find an ILBGK model which is consistent to the two-phase incompressible Navier-Stokes equations (2.31). As before, we shall allow neither density ϱ nor stress tensor $\tau_{\alpha\beta}$ to be discontinuous at the interface.

We will start with some theoretical investigations in Section 2.4.1, construct a consistent LBGK model of surface tension in Section 2.4.2, and finally, we will perform some numerical experiments in Section 2.4.3.

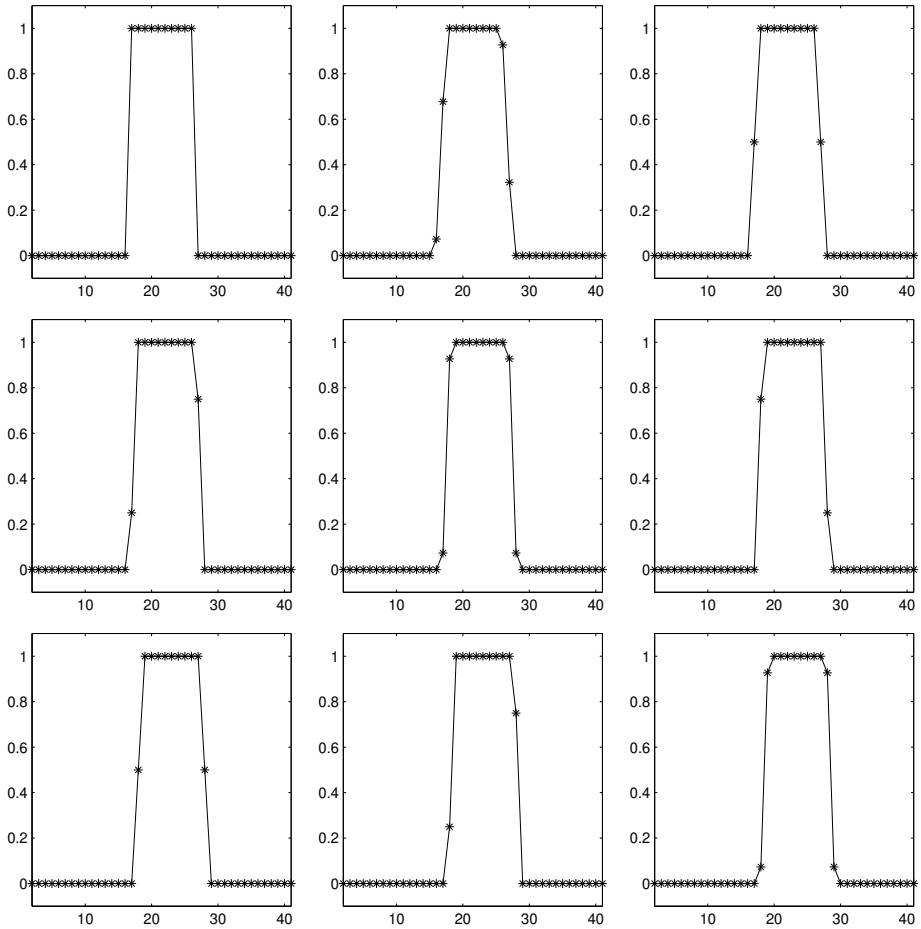


Figure 2.5: The initial setting and the first eight time steps (line by line, top left to bottom right) of the ILBGK interface tracking Algorithm 2.9 for the model problem with $\boldsymbol{u} = (0.25, 0)^T$ on a 40×40 lattice. All graphs show a cut through χ along the twentieth line of the lattice.

2.4.1. Theoretical background. Here, we want to give reasons for the assertion that it is possible to express the surface tension term $\partial_\beta((\delta_{\alpha\beta} - n_\alpha n_\beta)\delta_\Gamma)$ in terms of the local colour gradient $\boldsymbol{F}_{\Delta x}$. Note that we put the subscript Δx because we carry out a consistency analysis for $\Delta x \to 0$. In particular, we will constitute the claim that for an arbitrary test function $\varphi \in \mathcal{D}(\Omega)$

$$\langle (\delta_{\alpha\beta} - n_\alpha n_\beta)\delta_\Gamma, \varphi \rangle_\Omega = \lim_{\Delta x \to 0} \left\langle \delta_{\alpha\beta} \|\boldsymbol{F}_{\Delta x}\|_2 - \frac{F_{\Delta x;\alpha} F_{\Delta x;\beta}}{\|\boldsymbol{F}_{\Delta x}\|_2}, \varphi \right\rangle_\Omega , \tag{2.67}$$

where the right hand side is well-defined because for $\|\boldsymbol{F}_{\Delta x}\|_2 \to 0$ also $\boldsymbol{F}_{\Delta x;\alpha} \to 0$ and $\boldsymbol{F}_{\Delta x;\beta}/\|\boldsymbol{F}_{\Delta x}\|_2$ remains bounded, by proving

$$\langle (\delta_{\alpha\beta} - n_\alpha n_\beta)\delta_\Gamma, \varphi \rangle_\Omega = \lim_{\Delta x \to 0} \left\langle \delta_{\alpha\beta} \|\nabla \chi_{\Delta x}\|_2 - \frac{\partial_\alpha \chi_{\Delta x} \partial_\beta \chi_{\Delta x}}{\|\nabla \chi_{\Delta x}\|_2}, \varphi \right\rangle_\Omega , \tag{2.68}$$

where $\nabla = (\partial_1, \ldots, \partial_d)^T$. The idea of replacing

$$\left\langle \delta_{\alpha\beta} \|\boldsymbol{F}_{\Delta x}\|_2 - \frac{F_{\Delta x;\alpha} F_{\Delta x;\beta}}{\|\boldsymbol{F}_{\Delta x}\|_2}, \varphi \right\rangle_\Omega \qquad \text{with} \qquad \left\langle \delta_{\alpha\beta} \|\nabla \chi_{\Delta x}\|_2 - \frac{\partial_\alpha \chi_{\Delta x} \partial_\beta \chi_{\Delta x}}{\|\nabla \chi_{\Delta x}\|_2}, \varphi \right\rangle_\Omega$$

is supported by (2.50). To be well prepared for the investigations to follow, we will first consider some facts from differential geometry [**7, 74**]. Note that geometrical considerations are presented in the general context of d-dimensional space, not in the special $D2Q9$ setting.

In Section 2.1.3, we already assumed Γ to be an oriented $(d-1)$-dimensional $\mathcal{C}^2$-surface in $\Omega \subset \mathbb{R}^d$. For this reason, there is a tubular neighbourhood of Γ [**7**], i. e. for each $\boldsymbol{y} \in \Gamma$ we can choose an interval $I_{\boldsymbol{y}}$ of length $\varepsilon(\boldsymbol{y})$ along the normal direction of Γ in $\boldsymbol{y}$ such that for any $\Gamma \ni \boldsymbol{y}^1 \neq \boldsymbol{y}^2 \in \Gamma$ we have

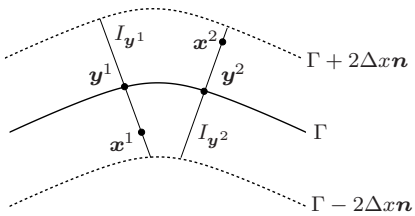


Figure 2.6: The tubular neighbourhood $\Gamma_{2\Delta x}$ around an interface Γ

$I_{\boldsymbol{y}^1} \cap I_{\boldsymbol{y}^2} = \emptyset$. Thus, if we choose

$$\Delta x \leq \frac{1}{2} \min_{\boldsymbol{y}\in\Gamma}\{\varepsilon(\boldsymbol{y})\}$$

there is an open neighbourhood $\Gamma_{2\Delta x}(t)$ of $\Gamma(t)$ such that any $\boldsymbol{x} \in \Gamma_{2\Delta x}(t)$ can be uniquely represented by

$$\boldsymbol{x} = \boldsymbol{y} + \Delta x \ell \boldsymbol{n}(\boldsymbol{y}),$$

where $\boldsymbol{y} \in \Gamma$ is fixed, $\boldsymbol{n}(\boldsymbol{y})$ denotes the inner unit normal vector of Γ at $\boldsymbol{y}$, and $\ell \in (-2, 2)$, see Figure 2.6 for a sketch. Note that for $\ell \in (-1, 1)$, the above equation describes points in the tubular neighbourhood $\Gamma_{\Delta x} \subset \Gamma_{2\Delta x}$ of Γ. Furthermore, we already found in Section 2.1.3 that since Γ is a $\mathcal{C}^2$-surface there exists for each point $\boldsymbol{y} \in \Gamma$ an open neighbourhood $\mathcal{U}^k \subset \Omega$ such that $\mathcal{U}^k \cap \Gamma$ is the image of a bijective $\mathcal{C}^2$-mapping $\boldsymbol{h}^k : \mathbb{R}^{d-1} \supset \Theta^k \to \Omega \subset \mathbb{R}^d$. Thus, each $\boldsymbol{x} \in \Gamma_{2\Delta x}$ can be written in the form

$$\boldsymbol{x} = \boldsymbol{h}^k\left(\boldsymbol{z}^k\right) + \Delta x \ell \boldsymbol{n} \circ \boldsymbol{h}^k\left(\boldsymbol{z}^k\right) \tag{2.69}$$

with $\boldsymbol{z}^k \in \Theta^k$.

Let us now recall the function $\chi_{\Delta x} \in \mathcal{C}^1\left([0, t_{max}] \times \Omega, \mathbb{R}\right)$ fulfilling $\lim_{\Delta x\to 0} \chi_{\Delta x} = \chi$. In Section 2.2.2, we did not need any further assumptions on the structure of $\chi_{\Delta x}$ in order to analyse the consistency of the local colour gradient $\boldsymbol{F}_{\Delta x}$. Here, however, we shall not investigate the colour gradient $\boldsymbol{F}_{\Delta x}$, which is a linear function of χ, but instead, we will examine nonlinear terms containing $\|\boldsymbol{F}_{\Delta x}\|_2$. Therefore, we need the additional assumptions

$$\chi_{\Delta x}(t, \boldsymbol{x}) = \begin{cases} 1 & \text{for } \boldsymbol{x} \in \Omega^r \setminus \Gamma_{\Delta x} \\ \eta_{\Delta x}\left(t, \boldsymbol{z}^k, \ell\right) & \text{for } \boldsymbol{x} \in \Gamma_{\Delta x} \cap \mathcal{U}^k \\ 0 & \text{for } \boldsymbol{x} \in \Omega^b \setminus \Gamma_{\Delta x} \end{cases} \tag{2.70a}$$

as well as

$$\eta_{\Delta x} \in \mathcal{C}^1\left([0, t_{max}] \times \Theta^k \times (-1, 1), \mathbb{R}\right), \qquad \partial_\ell \eta_{\Delta x} \geq 0, \qquad \lim_{\Delta x\to 0} \eta_{\Delta x}\left(t, \boldsymbol{z}^k, \ell\right) = \eta_0\left(t, \boldsymbol{z}^k, \ell\right) \tag{2.70b}$$

$$\eta_0 \in \mathcal{C}^1\left([0, t_{max}] \times \Theta^k \times (-1, 1), \mathbb{R}\right), \qquad \text{and} \qquad \partial_\ell \eta_0 \geq 0. \tag{2.70c}$$

An example of such a function will be given in the Appendix. Note that due to (2.70a) and $\chi_{\Delta x} \in \mathcal{C}^1\left([0, t_{max}] \times \Omega, \mathbb{R}\right)$, we have

$$\eta_{\Delta x}\left(t, \boldsymbol{z}^k, 1\right) = 1, \qquad \eta_{\Delta x}\left(t, \boldsymbol{z}^k, -1\right) = 0, \qquad \partial_\ell \eta_{\Delta x}\left(t, \boldsymbol{z}^k, \pm 1\right) = 0, \tag{2.71}$$

and analogous properties of η_0. Note also that since $\Gamma_{\Delta x}$ reduces to Γ in the limit $\Delta x \to 0$, condition (2.70a) immediately implies

$$\lim_{\Delta x\to 0} \chi_{\Delta x} = \begin{cases} 1 & \text{for } \boldsymbol{x} \in \Omega^r \\ 0 & \text{for } \boldsymbol{x} \in \Omega^b. \end{cases}$$

Let us now investigate the right hand side of (2.68). For this purpose, we first consider

$$\langle \|\nabla \chi_{\Delta x}\|_2, \varphi\rangle_\Omega = \int_\Omega \|\nabla \chi_{\Delta x}(t, \boldsymbol{x}))\|_2 \, \varphi(\boldsymbol{x}) \, d\boldsymbol{x}$$

for an arbitrary test function $\varphi \in \mathcal{D}(\Omega)$. Outside of $\Gamma_{\Delta x}$, the function $\chi_{\Delta x}$ is, by definition, constant in space, so we gain

$$\int_\Omega \|\nabla \chi_{\Delta x}(t, \boldsymbol{x})\|_2 \, \varphi(\boldsymbol{x}) \, d\boldsymbol{x} = \int_{\Gamma_{\Delta x}} \|\nabla \chi_{\Delta x}(t, \boldsymbol{x})\|_2 \, \varphi(\boldsymbol{x}) \, d\boldsymbol{x}.$$

At this stage, It is convenient to represent $\boldsymbol{x} \in \Gamma_{\Delta x}$ in the form (2.69). For this reason, we choose a partition of unity $\{\psi^k\}$ such that the support of $(\varphi\psi^k) \circ \boldsymbol{h}^k$ is compact in Θ^k (compare Section 2.1.3). This yields

$$\int_{\Gamma_{\Delta x}} \|\nabla\chi_{\Delta x}(t,\boldsymbol{x})\|_2\, \varphi(\boldsymbol{x})\, d\boldsymbol{x} = \sum_k \int_{\Gamma_{\Delta x}} \|\nabla\chi_{\Delta x}(t,\boldsymbol{x})\|_2 \left(\varphi\psi^k\right)(\boldsymbol{x})\, d\boldsymbol{x}.$$

Then, we apply the coordinate transformation $\Gamma_{\Delta x} \ni \boldsymbol{x} \mapsto \left(\boldsymbol{z}^k, \ell\right) \in \Theta^k \times (-1,1)$ to obtain

$$\begin{aligned}\int_{\Gamma_{\Delta x}} \|\nabla\chi_{\Delta x}(t,\boldsymbol{x})\|_2\, \varphi(\boldsymbol{x})\, d\boldsymbol{x} = \int_{\Theta^k}\int_{-1}^{1} &\left\|\nabla\chi_{\Delta x}\left(t, \boldsymbol{h}^k\left(\boldsymbol{z}^k\right) + \Delta x\ell\boldsymbol{n}\circ\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)\right)\right\|_2 \\ &\cdot \left(\varphi\psi^k\right)\left(\boldsymbol{h}^k\left(\boldsymbol{z}^k\right) + \Delta x\ell\boldsymbol{n}\circ\boldsymbol{h}^k\left(\boldsymbol{z}\right)\right)\widehat{\mathcal{J}}\left(\boldsymbol{z}^k,\ell\right)\, d\ell\, d\boldsymbol{z}^k\end{aligned}$$

with the Jacobian $\widehat{\mathcal{J}}\left(\boldsymbol{z}^k,\ell\right)$ given by

$$\widehat{\mathcal{J}}\left(\boldsymbol{z}^k,\ell\right) = \left|\det\begin{pmatrix}\partial_1 x_1 & \cdots & \partial_{d-1}x_1 & \partial_\ell x_1\\ \vdots & & \vdots & \vdots\\ \partial_1 x_d & \cdots & \partial_{d-1}x_d & \partial_\ell x_d\end{pmatrix}\right|,$$

where ∂_β means $\partial/\partial z^k_\beta$ for $\beta = 1,\dots,d-1$ and ∂_ℓ stands for $\partial/\partial\ell$. Thus,

$$\widehat{\mathcal{J}}(\boldsymbol{z},\ell) = \left|\det\begin{pmatrix}\partial_1 h^k_1 + \Delta x\ell\partial_1 n^k_1 & \cdots & \partial_{d-1}h^k_1 + \Delta x\ell\partial_{d-1}n^k_1 & \Delta x n^k_1\\ \vdots & & \vdots & \vdots\\ \partial_1 h^k_d + \Delta x\ell\partial_1 n^k_d & \cdots & \partial_{d-1}h^k_d + \Delta x\ell\partial_{d-1}n^k_d & \Delta x n^k_d\end{pmatrix}\right| \tag{2.72}$$

with $\boldsymbol{n}^k = \boldsymbol{n}\circ\boldsymbol{h}^k$. Applying now (2.70a) we obtain

$$\begin{aligned}&\int_{\Theta^k}\int_{-1}^{1}\left\|\nabla\chi_{\Delta x}\left(t,\boldsymbol{h}^k\left(\boldsymbol{z}^k\right) + \Delta x\ell\boldsymbol{n}\circ\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)\right)\right\|_2 \left(\varphi\psi^k\right)\left(\boldsymbol{h}^k\left(\boldsymbol{z}^k\right) + \Delta x\ell\boldsymbol{n}\circ\boldsymbol{h}^k\left(\boldsymbol{z}\right)\right)\widehat{\mathcal{J}}\left(\boldsymbol{z}^k,\ell\right)\, d\ell\, d\boldsymbol{z}^k\\ &\quad= \int_{\Theta^k}\int_{-1}^{1}\left\|\partial_\beta\eta_{\Delta x}\left(t,\boldsymbol{z}^k,\ell\right)\nabla z^k_\beta + \partial_\ell\eta_{\Delta x}\left(t,\boldsymbol{z}^k,\ell\right)\nabla\ell\right\|_2 \left(\varphi\psi^k\right)\left(\boldsymbol{h}^k\left(\boldsymbol{z}^k\right) + \Delta x\ell\boldsymbol{n}\circ\boldsymbol{h}^k\left(\boldsymbol{z}\right)\right)\\ &\qquad\cdot\widehat{\mathcal{J}}\left(\boldsymbol{z}^k,\ell\right)\, d\ell\, d\boldsymbol{z}^k.\end{aligned} \tag{2.73}$$

Now, we have to investigate $\nabla\ell$ in some more detail. We know the Jacobian of the transformation $\Gamma_{\Delta x} \ni \boldsymbol{x} \mapsto \left(\boldsymbol{z}^k,\ell\right) \in \Theta^k\times(-1,1)$ is given by (2.72) and that of the inverse transformation $\Theta^k\times(-1,1) \ni \left(\boldsymbol{z}^k,\ell\right) \mapsto \boldsymbol{x} \in \Gamma_{\Delta x}$ has the form

$$\widehat{\mathcal{J}}^{-1}(\boldsymbol{x}) = \left|\det\begin{pmatrix}\partial_1 z^k_1 & \cdots & \partial_d z^k_1\\ \vdots & & \vdots\\ \partial_1 z^k_{d-1} & \cdots & \partial_d z^k_{d-1}\\ \partial_1\ell & \cdots & \partial_d\ell\end{pmatrix}\right|,$$

where ∂_α represents $\partial/\partial x_\alpha$ for $\alpha = 1,\dots,d$, so the implicit function theorem provides

$$\begin{pmatrix}\partial_1 z^k_1 & \cdots & \partial_d z^k_1\\ \vdots & & \vdots\\ \partial_1 z^k_{d-1} & \cdots & \partial_d z^k_{d-1}\\ \partial_1\ell & \cdots & \partial_d\ell\end{pmatrix}\begin{pmatrix}\partial_1 h^k_1 + \Delta x\ell\partial_1 n^k_1 & \cdots & \partial_{d-1}h^k_1 + \Delta x\ell\partial_{d-1}n^k_1 & \Delta x n^k_1\\ \vdots & & \vdots & \vdots\\ \partial_1 h^k_d + \Delta x\ell\partial_1 n^k_d & \cdots & \partial_{d-1}h^k_d + \Delta x\ell\partial_{d-1}n^k_d & \Delta x n^k_d\end{pmatrix} = \begin{pmatrix}1 & & \\ & \ddots & \\ & & 1\end{pmatrix}.$$

In particular, this means

$$\begin{pmatrix}\partial_1 h^k_1 + \Delta x\ell\partial_1 n^k_1 & \cdots & \partial_1 h^k_d + \Delta x\ell\partial_1 n^k_d\\ \vdots & & \vdots\\ \partial_{d-1}h^k_1 + \Delta x\ell\partial_{d-1}n^k_1 & \cdots & \partial_{d-1}h^k_d + \Delta x\ell\partial_{d-1}n^k_d\\ \Delta x n^k_1 & \cdots & \Delta x n^k_d\end{pmatrix}\begin{pmatrix}\partial_1\ell\\ \vdots\\ \partial_d\ell\end{pmatrix} = \begin{pmatrix}0\\ \vdots\\ 0\\ 1\end{pmatrix}. \tag{2.74}$$

In Section 2.1.3, we already found $n_\alpha^k \partial_\beta h_\alpha^k = 0$ as well as $n_\alpha^k \partial_\beta n_\alpha^k = 0$. Since also $n_\alpha^k n_\alpha^k = 1$ by definition we realize that

$$\nabla \ell = \frac{1}{\Delta x} \boldsymbol{n}^k \tag{2.75}$$

is a solution of (2.74). This solution is unique because the matrix in (2.74) is invertible.

Using (2.75) we find the right hand side of equation (2.73) to fulfil

$$\begin{aligned}
&\int_{\Theta^k} \int_{-1}^{1} \left\| \partial_\beta \eta_{\Delta x} \left(t, \boldsymbol{z}^k, \ell\right) \nabla z_\beta^k + \partial_\ell \eta_{\Delta x} \left(t, \boldsymbol{z}^k, \ell\right) \nabla \ell \right\|_2 \left(\varphi \psi^k\right) \left(\boldsymbol{h}^k \left(\boldsymbol{z}^k\right) + \Delta x \ell \boldsymbol{n}^k \left(\boldsymbol{z}^k\right)\right) \widehat{\mathcal{J}} \left(\boldsymbol{z}^k, \ell\right) \, d\ell \, d\boldsymbol{z}^k \\
&= \int_{\Theta^k} \int_{-1}^{1} \left\| \Delta x \partial_\beta \eta_{\Delta x} \left(t, \boldsymbol{z}^k, \ell\right) \nabla z_\beta^k + \partial_\ell \eta_{\Delta x} \left(t, \boldsymbol{z}^k, \ell\right) \boldsymbol{n}^k \left(\boldsymbol{z}^k\right) \right\|_2 \left(\varphi \psi^k\right) \left(\boldsymbol{h}^k \left(\boldsymbol{z}^k\right) + \Delta x \ell \boldsymbol{n}^k \left(\boldsymbol{z}^k\right)\right) \\
&\qquad \cdot \widetilde{\mathcal{J}} \left(\boldsymbol{z}^k, \ell\right) \, d\ell \, d\boldsymbol{z}^k,
\end{aligned}$$

where

$$\widetilde{\mathcal{J}}(\boldsymbol{z}, \ell) = \frac{1}{\Delta x} \widehat{\mathcal{J}}(\boldsymbol{z}, \ell) = \left| \det \begin{pmatrix} \partial_1 h_1^k + \Delta x \ell \partial_1 n_1^k & \cdots & \partial_{d-1} h_1^k + \Delta x \ell \partial_{d-1} n_1^k & n_1^k \\ \vdots & & \vdots & \vdots \\ \partial_1 h_d^k + \Delta x \ell \partial_1 n_d^k & \cdots & \partial_{d-1} h_d^k + \Delta x \ell \partial_{d-1} n_d^k & n_d^k \end{pmatrix} \right|,$$

and therefore, applying the third condition in (2.70b) we determine

$$\begin{aligned}
&\lim_{\Delta x \to 0} \int_{\Theta^k} \int_{-1}^{1} \left\| \Delta x \partial_\beta \eta_{\Delta x} \left(t, \boldsymbol{z}^k, \ell\right) \nabla z_\beta^k + \partial_\ell \eta_{\Delta x} \left(t, \boldsymbol{z}^k, \ell\right) \boldsymbol{n}^k \left(\boldsymbol{z}^k\right) \right\|_2 \left(\varphi \psi^k\right) \left(\boldsymbol{h}^k \left(\boldsymbol{z}^k\right) + \Delta x \ell \boldsymbol{n}^k \left(\boldsymbol{z}^k\right)\right) \\
&\qquad \cdot \widetilde{\mathcal{J}} \left(\boldsymbol{z}^k, \ell\right) \, d\ell \, d\boldsymbol{z}^k = \int_{\Theta^k} \int_{-1}^{1} \left\| \partial_\ell \eta_0 \left(t, \boldsymbol{z}^k, \ell\right) \boldsymbol{n}^k \left(\boldsymbol{z}^k\right) \right\|_2 \left(\varphi \psi^k\right) \left(\boldsymbol{h}^k \left(\boldsymbol{z}^k\right)\right) \mathcal{J} \left(\boldsymbol{z}^k\right) \, d\ell \, d\boldsymbol{z}^k
\end{aligned}$$

with

$$\begin{aligned}
\mathcal{J}(\boldsymbol{z}) &= \left| \det \begin{pmatrix} \partial_1 h_1^k & \cdots & \partial_{d-1} h_1^k & n_1^k \\ \vdots & & \vdots & \vdots \\ \partial_1 h_d^k & \cdots & \partial_{d-1} h_d^k & n_d^k \end{pmatrix} \right| \\
&= \left(\det \begin{pmatrix} \partial_1 h_1^k & \cdots & \partial_1 h_d^k \\ \vdots & & \vdots \\ \partial_{d-1} h_1^k & \cdots & \partial_{d-1} h_d^k \\ n_1^k & \cdots & n_d^k \end{pmatrix} \det \begin{pmatrix} \partial_1 h_1^k & \cdots & \partial_{d-1} h_1^k & n_1^k \\ \vdots & & \vdots & \vdots \\ \partial_1 h_d^k & \cdots & \partial_{d-1} h_d^k & n_d^k \end{pmatrix} \right)^{1/2}.
\end{aligned}$$

Using again $n_\alpha^k \partial_\beta h_\alpha^k = 0$ and $n_\alpha^k n_\alpha^k = 1$ we find by straight calculation that

$$\begin{aligned}
&\left(\det \begin{pmatrix} \partial_1 h_1^k & \cdots & \partial_1 h_d^k \\ \vdots & & \vdots \\ \partial_{d-1} h_1^k & \cdots & \partial_{d-1} h_d^k \\ n_1^k & \cdots & n_d^k \end{pmatrix} \det \begin{pmatrix} \partial_1 h_1^k & \cdots & \partial_{d-1} h_1^k & n_1^k \\ \vdots & & \vdots & \vdots \\ \partial_1 h_d^k & \cdots & \partial_{d-1} h_d^k & n_d^k \end{pmatrix} \right)^{1/2} \\
&\qquad = \left\{ \det \left(\begin{pmatrix} \partial_1 h_1^k & \cdots & \partial_1 h_d^k \\ \vdots & & \vdots \\ \partial_{d-1} h_1^k & \cdots & \partial_{d-1} h_d^k \end{pmatrix} \begin{pmatrix} \partial_1 h_1^k & \cdots & \partial_{d-1} h_1^k \\ \vdots & & \vdots \\ \partial_1 h_d^k & \cdots & \partial_{d-1} h_d^k \end{pmatrix} \right) \right\}^{1/2}
\end{aligned}$$

and thus, $\mathcal{J}(\boldsymbol{z})$ equals the Jacobian given by (2.19), i. e. the Jacobian of the transformation $\mathcal{U}^k \cap \Gamma \ni \boldsymbol{y} \mapsto \boldsymbol{z}^k \in \Theta^k$. Furthermore, due to $\left\| \boldsymbol{n}^k \right\|_2 = 1$ we have

$$\begin{aligned}
&\int_{\Theta^k} \int_{-1}^{1} \left\| \partial_\ell \eta_0 \left(t, \boldsymbol{z}^k, \ell\right) \boldsymbol{n}^k \left(\boldsymbol{z}^k\right) \right\|_2 \left(\varphi \psi^k\right) \left(\boldsymbol{h}^k \left(\boldsymbol{z}^k\right)\right) \mathcal{J} \left(\boldsymbol{z}^k\right) \, d\ell \, d\boldsymbol{z}^k \\
&\qquad = \int_{\Theta^k} \left(\int_{-1}^{1} \left| \partial_\ell \eta_0 \left(t, \boldsymbol{z}^k, \ell\right) \right| \, d\ell \right) \left(\varphi \psi^k\right) \left(\boldsymbol{h}^k \left(\boldsymbol{z}^k\right)\right) \mathcal{J} \left(\boldsymbol{z}^k\right) \, d\boldsymbol{z}^k.
\end{aligned}$$

At this stage, the fourth condition in (2.70b) yields

$$\int_{-1}^{1} \left|\partial_\ell \eta_0\left(t, \boldsymbol{z}^k, \ell\right)\right| \, d\ell = \int_{-1}^{1} \partial_\ell \eta_0\left(t, \boldsymbol{z}^k, \ell\right) \, d\ell$$

and due to (2.71), we find that independent of t and $\boldsymbol{z}^k$,

$$\int_{-1}^{1} \partial_\ell \eta_0\left(t, \boldsymbol{z}^k, \ell\right) \, d\ell = \eta_0\left(t, \boldsymbol{z}^k, 1\right) - \eta_0\left(t, \boldsymbol{z}^k, -1\right) = 1. \tag{2.76}$$

Therefore,

$$\int_{\Theta^k} \left(\int_{-1}^{1} \left|\partial_\ell \eta_0\left(t, \boldsymbol{z}^k, \ell\right)\right| \, d\ell\right) \left(\varphi\psi^k\right) \left(\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)\right) \mathcal{J}\left(\boldsymbol{z}^k\right) \, d\boldsymbol{z}^k = \int_{\Theta^k} \left(\varphi\psi^k\right) \left(\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)\right) \mathcal{J}\left(\boldsymbol{z}^k\right) \, d\boldsymbol{z}^k.$$

Summing over k and performing the coordinate transformation $\Theta^k \ni \boldsymbol{z}^k \mapsto \boldsymbol{y} \in \mathcal{U}^k \cap \Gamma$ we finally obtain

$$\sum_k \int_{\Theta^k} \left(\varphi\psi^k\right) \left(\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)\right) \mathcal{J}\left(\boldsymbol{z}^k\right) \, d\boldsymbol{z}^k = \int_\Gamma \varphi(\boldsymbol{y}) d\mathscr{H}(\boldsymbol{y}) = \langle \delta_\Gamma, \varphi \rangle_\Omega \,,$$

where $\mathscr{H}$ denotes the surface measure on Γ. This completes the proof of

Proposition 2.3. *In* $\mathcal{D}'(\Omega)$,

$$\lim_{\Delta x \to 0} \|\nabla \chi_{\Delta x}\|_2 = \delta_\Gamma.$$

Let us now consider the consistency of $\partial_\alpha \chi_{\Delta x} \partial_\beta \chi_{\Delta x} / \|\nabla \chi_{\Delta x}\|_2$ in $\mathcal{D}'(\Omega)$. Due to definition (2.70), we first obtain in analogy to the above

$$\int_\Omega \frac{\partial_\alpha \chi_{\Delta x}(\boldsymbol{x}) \partial_\gamma \chi_{\Delta x}(\boldsymbol{x})}{\|\nabla \chi_{\Delta x}(\boldsymbol{x})\|_2} \varphi(\boldsymbol{x}) \, d\boldsymbol{x} = \sum_k \int_{\Theta^k} \int_{-1}^{1} \left(\Delta x \partial_\beta \eta_{\Delta x}\left(t, \boldsymbol{z}^k, \ell\right) \partial_\alpha z_\beta^k + \partial_\ell \eta_{\Delta x}\left(t, \boldsymbol{z}^k, \ell\right) n_\alpha^k\left(\boldsymbol{z}^k\right)\right)$$
$$\cdot \frac{\Delta x \partial_\beta \eta_{\Delta x}\left(t, \boldsymbol{z}^k, \ell\right) \partial_\gamma z_\beta^k + \partial_\ell \eta_{\Delta x}\left(t, \boldsymbol{z}^k, \ell\right) n_\gamma^k\left(\boldsymbol{z}^k\right)}{\left\|\Delta x \partial_\beta \eta_{\Delta x}\left(t, \boldsymbol{z}^k, \ell\right) \nabla z_\beta^k + \partial_\ell \eta_{\Delta x}\left(t, \boldsymbol{z}^k, \ell\right) \boldsymbol{n}^k\left(\boldsymbol{z}^k\right)\right\|_2} \left(\varphi\psi^k\right) \left(\boldsymbol{h}^k\left(\boldsymbol{z}^k\right) + \Delta x \ell \boldsymbol{n}^k\left(\boldsymbol{z}^k\right)\right) \widetilde{\mathcal{J}}\left(\boldsymbol{z}^k, \ell\right) \, d\ell \, d\boldsymbol{z}^k,$$

where ∂_α means $\partial/\partial x_\alpha$ and ∂_γ stands for $\partial/\partial x_\gamma$ while ∂_β represents $\partial/\partial z_\beta^k$. We determine

$$\lim_{\Delta x \to 0} \int_{\Theta^k} \int_{-1}^{1} \left(\Delta x \partial_\beta \eta_{\Delta x}\left(t, \boldsymbol{z}^k, \ell\right) \partial_\alpha z_\beta^k + \partial_\ell \eta_{\Delta x}\left(t, \boldsymbol{z}^k, \ell\right) n_\alpha^k\left(\boldsymbol{z}^k\right)\right)$$
$$\cdot \frac{\Delta x \partial_\beta \eta_{\Delta x}\left(t, \boldsymbol{z}^k, \ell\right) \partial_\gamma z_\beta^k + \partial_\ell \eta_{\Delta x}\left(t, \boldsymbol{z}^k, \ell\right) n_\gamma^k\left(\boldsymbol{z}^k\right)}{\left\|\Delta x \partial_\beta \eta_{\Delta x}\left(t, \boldsymbol{z}^k, \ell\right) \nabla z_\beta^k + \partial_\ell \eta_{\Delta x}\left(t, \boldsymbol{z}^k, \ell\right) \boldsymbol{n}^k\left(\boldsymbol{z}^k\right)\right\|_2} \left(\varphi\psi^k\right) \left(\boldsymbol{h}^k\left(\boldsymbol{z}^k\right) + \Delta x \ell \boldsymbol{n}^k\left(\boldsymbol{z}^k\right)\right) \widetilde{\mathcal{J}}\left(\boldsymbol{z}^k, \ell\right) \, d\ell \, d\boldsymbol{z}^k$$
$$= \int_{\Theta^k} \int_{-1}^{1} \frac{\left(\partial_\ell \eta_0\left(t, \boldsymbol{z}^k, \ell\right) n_\alpha^k\left(\boldsymbol{z}^k\right)\right) \left(\partial_\ell \eta_0\left(t, \boldsymbol{z}^k, \ell\right) n_\gamma^k\left(\boldsymbol{z}^k\right)\right)}{\left\|\partial_\ell \eta_0\left(t, \boldsymbol{z}^k, \ell\right) \boldsymbol{n}^k\left(\boldsymbol{z}^k\right)\right\|_2} \left(\varphi\psi^k\right) \left(\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)\right) \mathcal{J}\left(\boldsymbol{z}^k\right) \, d\ell \, d\boldsymbol{z}^k$$

and using now $\left\|\boldsymbol{n}^k\right\|_2 = 1$ together with the second condition in (2.70c) we obtain

$$\int_{\Theta^k} \int_{-1}^{1} \frac{\left(\partial_\ell \eta_0\left(t, \boldsymbol{z}^k, \ell\right) n_\alpha^k\left(\boldsymbol{z}^k\right)\right) \left(\partial_\ell \eta_0\left(t, \boldsymbol{z}^k, \ell\right) n_\gamma^k\left(\boldsymbol{z}^k\right)\right)}{\left\|\partial_\ell \eta_0\left(t, \boldsymbol{z}^k, \ell\right) \boldsymbol{n}^k\left(\boldsymbol{z}^k\right)\right\|_2} \left(\varphi\psi^k\right) \left(\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)\right) \mathcal{J}\left(\boldsymbol{z}^k\right) \, d\ell \, d\boldsymbol{z}^k$$
$$= \int_{\Theta^k} \left(\int_{-1}^{1} \partial_\ell \eta_0\left(t, \boldsymbol{z}^k, \ell\right) \, d\ell\right) n_\alpha^k\left(\boldsymbol{z}^k\right) n_\gamma^k\left(\boldsymbol{z}^k\right) \left(\varphi\psi^k\right) \left(\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)\right) \mathcal{J}\left(\boldsymbol{z}^k\right) \, d\boldsymbol{z}^k.$$

Recalling equation (2.76) we find

$$\int_{\Theta^k} \left(\int_{-1}^{1} \partial_\ell \eta_0\left(t, \boldsymbol{z}^k, \ell\right) \, d\ell\right) n_\alpha^k\left(\boldsymbol{z}^k\right) n_\gamma^k\left(\boldsymbol{z}^k\right) \left(\varphi\psi^k\right) \left(\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)\right) \mathcal{J}\left(\boldsymbol{z}^k\right) \, d\boldsymbol{z}^k$$
$$= \int_{\Theta^k} n_\alpha^k\left(\boldsymbol{z}^k\right) n_\gamma^k\left(\boldsymbol{z}^k\right) \left(\varphi\psi^k\right) \left(\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)\right) \mathcal{J}\left(\boldsymbol{z}^k\right) \, d\boldsymbol{z}^k$$

which leads to

$$\sum_k \int_{\Theta^k} n_\alpha^k\left(\boldsymbol{z}^k\right) n_\gamma^k\left(\boldsymbol{z}^k\right) \left(\varphi\psi^k\right) \left(\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)\right) \mathcal{J}\left(\boldsymbol{z}^k\right) \, d\boldsymbol{z}^k = \int_\Gamma n_\alpha(\boldsymbol{y}) n_\gamma(\boldsymbol{y}) \varphi(\boldsymbol{y}) d\mathscr{H}(\boldsymbol{y}),$$

so we finally gain

Proposition 2.4. *In* $\mathcal{D}'(\Omega)$,

$$\lim_{\Delta x\to 0}\frac{\partial_\alpha\chi_{\Delta x}\partial_\beta\chi_{\Delta x}}{\|\nabla\chi_{\Delta x}\|_2}=n_\alpha n_\gamma\delta_\Gamma.$$

Combining now Propositions 2.3 and 2.4 we have a proof of relation (2.68). Note that defining

$$\mathcal{N}_{\Delta x;\alpha}=\begin{cases}\partial_\alpha\chi_{\Delta x}/\|\nabla\chi_{\Delta x}\|_2 & \text{for } \|\nabla\chi_{\Delta x}\|_2>0\\ 0 & \text{otherwise}\end{cases}$$

we can proceed in complete analogy to the above to show

Proposition 2.5. *In* $\mathcal{D}'(\Omega)$,

$$\lim_{\Delta x\to 0}\mathcal{N}_{\Delta x;\alpha}=n_\alpha\delta_\Gamma.$$

2.4.2. Consistent modelling of surface tension. Let us consider again the lattice BGK equation (2.42). In this equation, surface tension is modelled by the term $S(\boldsymbol{v})$ in the equilibrium particle density f_S^{eq} defined by (2.43). We shall now discuss how to choose this term properly.

In equivalent moment analysis for one-phase lattice BGK (Section 1.1.2), it is striking that the term

$$f^*(\boldsymbol{v})\frac{9\Delta x^2}{2}\rho u_\alpha u_\beta\left(v_\alpha v_\beta-\frac{\delta_{\alpha\beta}}{3}\right)$$

in the equilibrium particle density f^{eq} given by (1.11) produces the term $\partial_\beta(\rho u_\alpha u_\beta)$ in the Navier-Stokes equations (1.8). Inspired by this observation, we choose

$$S(\boldsymbol{v})=f^*(\boldsymbol{v})\frac{9\Delta x^2}{2}\sigma\left(\frac{F_{\Delta x;\alpha}F_{\Delta x;\beta}}{\|\boldsymbol{F}_{\Delta x}\|_2}-\delta_{\alpha\beta}\|\boldsymbol{F}_{\Delta x}\|_2\right)\left(v_\alpha v_\beta-\frac{\delta_{\alpha\beta}}{3}\right)$$

and therefore,

$$\begin{aligned}&f_S^{eq}(\rho,\boldsymbol{u};\boldsymbol{v})=\\&f^*(\boldsymbol{v})\left(\rho+3\Delta x\rho u_\alpha v_\alpha+\frac{9\Delta x^2}{2}\left(\rho u_\alpha u_\beta+\sigma\left(\frac{F_{\Delta x;\alpha}F_{\Delta x;\beta}}{\|\boldsymbol{F}_{\Delta x}\|_2}-\delta_{\alpha\beta}\|\boldsymbol{F}_{\Delta x}\|_2\right)\right)\left(v_\alpha v_\beta-\frac{\delta_{\alpha\beta}}{3}\right)\right).\end{aligned}\tag{2.77}$$

Performing now an equivalent moment analysis of (2.42) we obtain the governing equations

$$\langle\partial_\alpha u_\alpha,\varphi\rangle_\Omega=\mathcal{O}\left(\Delta x^2\right),\tag{2.78a}$$

$$\begin{aligned}&\left\langle\partial_t(\rho u_\alpha)+\partial_\beta\left((1-\omega)\vartheta_{\alpha\beta}+\omega\rho u_\alpha u_\beta+\omega\sigma\left(\frac{F_{\Delta x;\alpha}F_{\Delta x;\beta}}{\|\boldsymbol{F}_{\Delta x}\|_2}-\delta_{\alpha\beta}\|\boldsymbol{F}_{\Delta x}\|_2\right)\right)+\partial_\alpha p,\varphi\right\rangle_\Omega\\&\qquad=\left\langle\frac{1}{6}\Big(\partial_{\beta\beta}(\rho u_\alpha)+2\partial_{\alpha\beta}(\rho u_\beta)\Big),\varphi\right\rangle_\Omega+\mathcal{O}\left(\Delta x^2\right),\end{aligned}\tag{2.78b}$$

and

$$\langle\vartheta_{\alpha\beta},\varphi\rangle_\Omega=\left\langle\rho u_\alpha u_\beta+\sigma\left(\frac{F_{\Delta x;\alpha}F_{\Delta x;\beta}}{\|\boldsymbol{F}_{\Delta x}\|_2}-\delta_{\alpha\beta}\|\boldsymbol{F}_{\Delta x}\|_2\right)-\frac{1}{3\omega}\Big(\partial_\beta(\rho u_\alpha)+\partial_\alpha(\rho u_\beta)\Big),\varphi\right\rangle_\Omega+\mathcal{O}\left(\Delta x^2\right),\tag{2.78c}$$

where $\varphi\in\mathcal{D}(\Omega)$ is an arbitrary test function. Note that sufficient regularity of ϱ and $\tau_{\alpha\beta}$ in Ω is crucial at this point because otherwise, the basic prerequisites of equivalent moment analysis do not hold. Inserting now (2.78c) into (2.78b) and applying (1.9), (1.17) as well as (1.18) we end up with

$$\langle\partial_\alpha u_\alpha,\varphi\rangle_\Omega=\mathcal{O}\left(\Delta x^2\right)$$

and

$$\langle\partial_t(\varrho u_\alpha)+\partial_\beta(\varrho u_\alpha u_\beta),\varphi\rangle_\Omega-\left\langle\sigma\partial_\beta\left(\delta_{\alpha\beta}\|\boldsymbol{F}_{\Delta x}\|_2-\frac{F_{\Delta x;\alpha}F_{\Delta x;\beta}}{\|\boldsymbol{F}_{\Delta x}\|_2}\right),\varphi\right\rangle_\Omega=\langle\partial_\beta\eta_{\alpha\beta},\varphi\rangle_\Omega+\mathcal{O}\left(\Delta x^2\right).$$

Assuming now (2.67) holds we can rewrite this system of equations for $\Delta x\to 0$ in the form

$$\langle\partial_\alpha u_\alpha,\varphi\rangle_\Omega=0\qquad\text{and}\qquad\langle\partial_t(\varrho u_\alpha)+\partial_\beta(\varrho u_\alpha u_\beta),\varphi\rangle_\Omega-\langle\sigma\partial_\beta\left(\left(\delta_{\alpha\beta}-n_\alpha n_\beta\right)\delta_\Gamma\right),\varphi\rangle_\Omega=\langle\partial_\beta\eta_{\alpha\beta},\varphi\rangle_\Omega,$$

so we gain

Proposition 2.6. *In $\mathcal{D}'(\Omega)$, the equivalent moment equations of the lattice BGK scheme* (2.42) *with the equilibrium particle density f_S^{eq} defined by* (2.77) *are consistent to the two-phase incompressible Navier-Stokes equations* (2.31) *if assumption* (2.67) *holds and if both density ϱ and stress tensor $\tau_{\alpha\beta}$ are sufficiently smooth at the interface.*

Note that Nie et al. [**57**] suggest the surface tension term $\widehat{S}(\boldsymbol{v})$ defined by $\widehat{S}(\boldsymbol{c}_0) = 0$ and

$$\widehat{S}(\boldsymbol{c}_j) = f^*(\boldsymbol{c}_j)\frac{9\Delta x^2}{2}\frac{A}{\omega}\left(2\frac{F_{\Delta x;\alpha}F_{\Delta x;\beta}}{\|\boldsymbol{F}_{\Delta x}\|_2}\frac{c_{j;\alpha}c_{j;\beta}}{\|\boldsymbol{c}_j\|_2^2} - 1\right) \qquad \text{for } j = 1,\dots,8$$

instead of S. Following this proposal we obtain the equilibrium particle density $f_{\widehat{S}}^{eq}$ given by $f_{\widehat{S}}^{eq}(\rho,\boldsymbol{u};\boldsymbol{c}_0) = f^*(\boldsymbol{c}_j)\rho$ as well as

$$f_{\widehat{S}}^{eq}(\rho,\boldsymbol{u};\boldsymbol{c}_j) = f^*(\boldsymbol{c}_j)\left\{\rho + 3\Delta x\rho u_\alpha c_{j;\alpha} + \frac{9\Delta x^2}{2}\left(\rho u_\alpha u_\beta\left(c_{j;\alpha}c_{j;\beta} - \frac{\delta_{\alpha\beta}}{3}\right) + \frac{A}{\omega}\left(2\frac{F_{\Delta x;\alpha}F_{\Delta x;\beta}}{\|\boldsymbol{F}_{\Delta x}\|_2}\frac{c_{j;\alpha}c_{j;\beta}}{\|\boldsymbol{c}_j\|_2^2} - 1\right)\right)\right\}$$

for $j = 1,\dots,8$ and using this equilibrium particle density instead of f_S^{eq} we find that the equivalent moment system of (2.42) contains

$$\langle \partial_\alpha u_\alpha, \varphi\rangle_\Omega = \mathcal{O}\left(\Delta x^2\right), \tag{2.79a}$$

$$\left\langle \partial_t(\rho u_\alpha) + \partial_\beta\left((1-\omega)\vartheta_{\alpha\beta} + \omega\rho u_\alpha u_\beta + A\left(\frac{F_{\Delta x;\alpha}F_{\Delta x;\beta}}{\|\boldsymbol{F}_{\Delta x}\|_2} - \Upsilon_{\Delta x;\alpha\beta}\right)\right) + \partial_\alpha p, \varphi\right\rangle_\Omega = \left\langle \frac{1}{6}\Big(\partial_{\beta\beta}(\rho u_\alpha) + 2\partial_{\alpha\beta}(\rho u_\beta)\Big), \varphi\right\rangle_\Omega + \mathcal{O}\left(\Delta x^2\right), \tag{2.79b}$$

and

$$\langle \vartheta_{\alpha\beta}, \varphi\rangle_\Omega = \left\langle \rho u_\alpha u_\beta + \frac{A}{\omega}\left(\frac{F_{\Delta x;\alpha}F_{\Delta x;\beta}}{\|\boldsymbol{F}_{\Delta x}\|_2} - \Upsilon_{\Delta x;\alpha\beta}\right) - \frac{1}{3\omega}\Big(\partial_\beta(\rho u_\alpha) + \partial_\alpha(\rho u_\beta)\Big), \varphi\right\rangle_\Omega + \mathcal{O}\left(\Delta x^2\right), \tag{2.79c}$$

where

$$\boldsymbol{\Upsilon}_{\Delta x} = \begin{pmatrix} F_{\Delta x;2}^2/\|\boldsymbol{F}_{\Delta x}\|_2 & 0 \\ 0 & F_{\Delta x;1}^2/\|\boldsymbol{F}_{\Delta x}\|_2 \end{pmatrix},$$

in place of (2.78). We observe

$$\begin{pmatrix} F_{\Delta x;2}^2/\|\boldsymbol{F}_{\Delta x}\|_2 & 0 \\ 0 & F_{\Delta x;1}^2/\|\boldsymbol{F}_{\Delta x}\|_2 \end{pmatrix} = \begin{pmatrix} \|\boldsymbol{F}_{\Delta x}\|_2 & 0 \\ 0 & \|\boldsymbol{F}_{\Delta x}\|_2 \end{pmatrix} - \begin{pmatrix} F_{\Delta x;1}^2/\|\boldsymbol{F}_{\Delta x}\|_2 & 0 \\ 0 & F_{\Delta x;2}^2/\|\boldsymbol{F}_{\Delta x}\|_2 \end{pmatrix},$$

so assuming

$$\lim_{\Delta x\to 0}\langle \|\boldsymbol{F}_{\Delta x}\|_2, \varphi\rangle_\Omega = \langle \delta_\Gamma, \varphi\rangle_\Omega \qquad \text{and} \qquad \lim_{\Delta x\to 0}\left\langle \frac{F_{\Delta x;\alpha}F_{\Delta x;\beta}}{\|\boldsymbol{F}_{\Delta x}\|_2}, \varphi\right\rangle_\Omega = \langle n_\alpha n_\beta \delta_\Gamma, \varphi\rangle_\Omega$$

(compare Propositions 2.3 and 2.4) we obtain

$$\lim_{\Delta x\to 0}\left\langle \left(\frac{F_{\Delta x;\alpha}F_{\Delta x;\beta}}{\|\boldsymbol{F}_{\Delta x}\|_2} - \Upsilon_{\Delta x;\alpha\beta}\right), \varphi\right\rangle_\Omega = -\langle (\delta_{\alpha\beta} - n_\alpha n_\beta)\delta_\Gamma, \varphi\rangle_\Omega + \langle N_{\alpha\beta}, \varphi\rangle_\Omega,$$

where

$$\boldsymbol{N} = \begin{pmatrix} n_1^2 & 0 \\ 0 & n_2^2 \end{pmatrix},$$

and therefore, in the limit $\Delta x \to 0$, we can rewrite (2.79) in the form $\langle \partial_\alpha u_\alpha\rangle_\Omega = 0$ respectively

$$\left\langle \partial_t(\varrho u_\alpha) + \partial_\beta\left(\varrho u_\alpha u_\beta - \frac{A}{\omega}(\delta_{\alpha\beta} - n_\alpha n_\beta)\delta_\Gamma + \frac{A}{\omega}N_{\alpha\beta}\right), \varphi\right\rangle_\Omega = \langle \partial_\beta \eta_{\alpha\beta}, \varphi\rangle_\Omega.$$

Defining now the surface tension coefficient $\sigma_{\widehat{S}} = A/\omega$ is the best we can do to bring this result as close as possible to the two-phase incompressible Navier-Stokes equations (2.31). However, we can not eliminate the spurious term $\sigma_{\widehat{S}}\partial_\beta N_{\alpha\beta}$ in the momentum equation. Due to this inconsistency, we claim that the

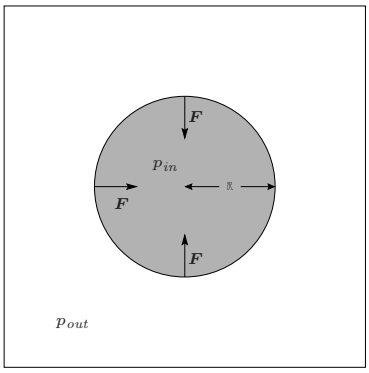


Figure 2.7: The setting of the bubble test with a bubble of radius $\mathbb{R}$ located at the centre of a basin $\Omega = [0, 4\mathbb{R}] \times [0, 4\mathbb{R}]$

surface tension term S is a better choice than $\widehat{S}$. We will now underline this claim with a numerical experiment, the so-called bubble test.

2.4.3. Bubble test. Here, we simulate a circular bubble of red fluid resting in a basin of blue fluid, as illustrated in Figure 2.7, and we use this setting to verify Laplace's law [**49**]

$$p_{in} - p_{out} = \frac{\sigma}{\mathbb{R}},$$

where $\mathbb{R}$ stands for the radius of the bubble while p_{in} and p_{out} denote the pressure inside respectively outside the bubble. The latter are determined by averaging the pressure calculated with (1.17) over all lattice points at an Euclidean distance of less than $0.7\mathbb{R}$ respectively more than $1.3\mathbb{R}$ from the centre of the bubble. We consider a quadratic basin with side length $4\mathbb{R}$ and periodic boundaries and suppose the bubble is located at the centre of this basin, see again Figure 2.7. Furthermore, we assume that

$$\rho(0, \boldsymbol{x}) = \varrho(0, \boldsymbol{x}) = 1, \qquad u_\alpha(0, \boldsymbol{x}) = 0, \qquad \text{and} \qquad f(0, \boldsymbol{x}; \boldsymbol{v}) = f_S^{eq}(\rho(0, \boldsymbol{x}), \boldsymbol{u}(0, \boldsymbol{x}); \boldsymbol{v}) \tag{2.80a}$$

for $\boldsymbol{x} \in \mathcal{X}$ as well as

$$\rho^r(0, \boldsymbol{x}) = \begin{cases} \rho(0, \boldsymbol{x}) & \text{for } \boldsymbol{x} \in \mathcal{X} \cap \Omega^r(0) \\ 0 & \text{for } \boldsymbol{x} \in \mathcal{X} \cap \Omega^b(0). \end{cases} \tag{2.80b}$$

Then, we proceed as described in Algorithm 2.6. Note that due to (2.80), we have $\rho^r = \chi$ in the given situation. For our convenience, we choose $\omega = 1$ for all the computations discussed here.

Simulations are performed with $L = 10$ on a 40×40 lattice and with $L = 20$ on an 80×80 lattice for $\sigma = 0.00005$, 0.0001, 0.0005, 0.001, 0.005, 0.01, 0.05, and 0.1. We always perform 1000 time steps on the 40×40 lattice and 2000 time steps on the 80×80 lattice. Plotting the values of $\Delta p = p_{in} - p_{out}$ versus $\sigma/\mathbb{R}$ in Figure 2.8, we find that on both lattices, the surface tension term S produces significantly better results than $\widehat{S}$ for any choice of σ. We also observe the so-called spurious currents, a well-known numerical artifact that originates from the discretisation of the surface tension term $\partial_\beta((\delta_{\alpha\beta} - n_\alpha n_\beta)\delta_\Gamma)$, see for example Ginzburg and Wittum [**24**] or Lafaurie et al. [**47**]. The velocity fields produced by S and $\widehat{S}$ are exemplified for $\sigma = 0.005$ and $\sigma = 0.01$ in Figures 2.9 (40×40 lattice) and 2.10 (80×80 lattice). It is striking that the flow field produced by $\widehat{S}$ is far more widespread than that produced by S. Furthermore, we learn from Figure 2.11 that for the surface tension term $\widehat{S}$, also $\|\boldsymbol{u}\|_2$ becomes larger than for S. For the sake of completeness, we also show percental mass loss and computing times for the simulations with surface tension term S in Table 2.1 and for those with surface tension term $\widehat{S}$ in Table 2.2. Finally, we remark that total momentum is always zero while the bubble radius equals 10 lattice units in all computations on a 40×40 lattice and 20 lattice units in all computations on an 80×80 lattice.

Altogether, we find the surface tension term S to produce more accurate results and less distinctive numerical artifacts than $\widehat{S}$. Therefore, we will from now on exclusively use S to include surface tension effects into lattice BGK simulations.

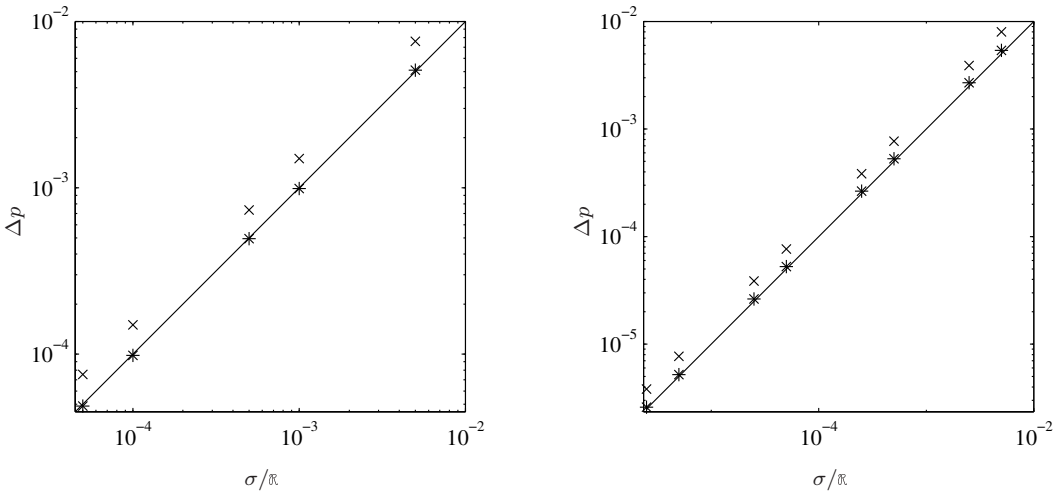


Figure 2.8: Verification of Laplace's law by bubble tests with surface tension terms S ('$*$') and $\widehat{S}$ ('$\times$') on a 40×40 (left) and an 80×80 (right) lattice. The straight line represents the correct solution.

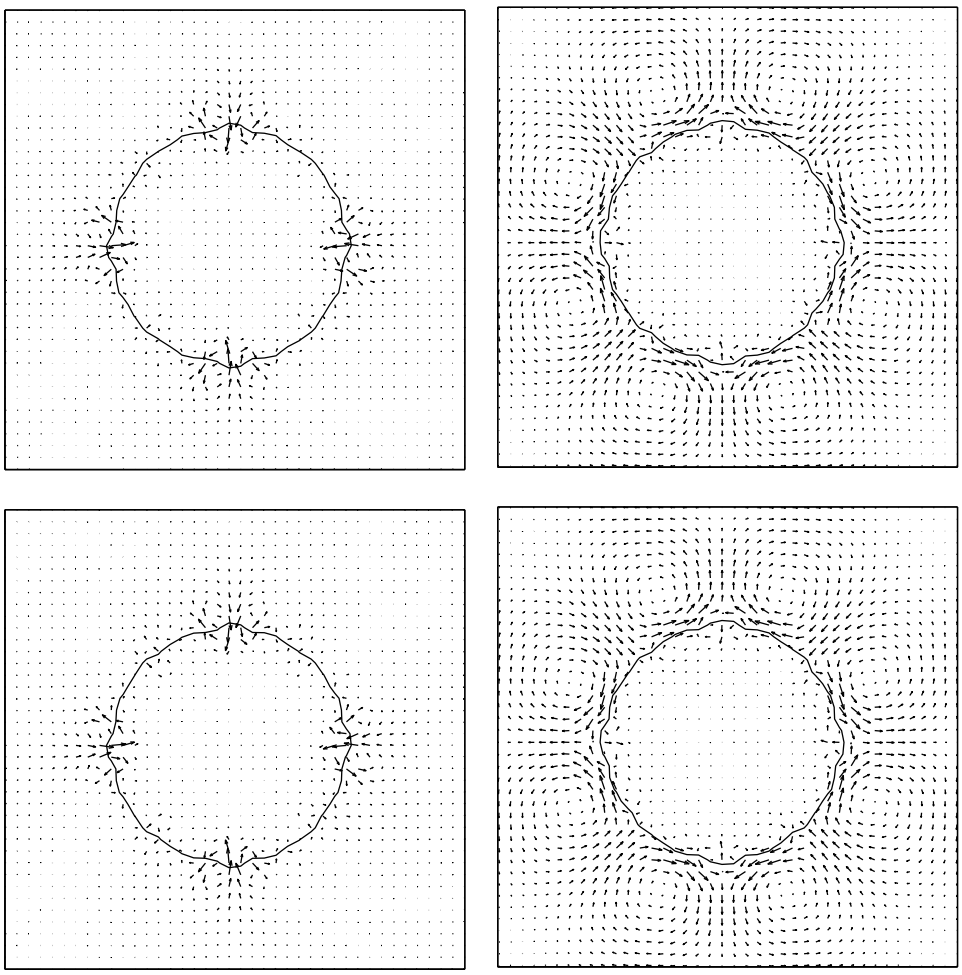


Figure 2.9: Velocity fields produced by the surface tension terms S (left) and $\widehat{S}$ (right) in bubble tests with $\sigma = 0.005$ (top) and $\sigma = 0.01$ (bottom). All computations were performed on a 40×40 lattice.

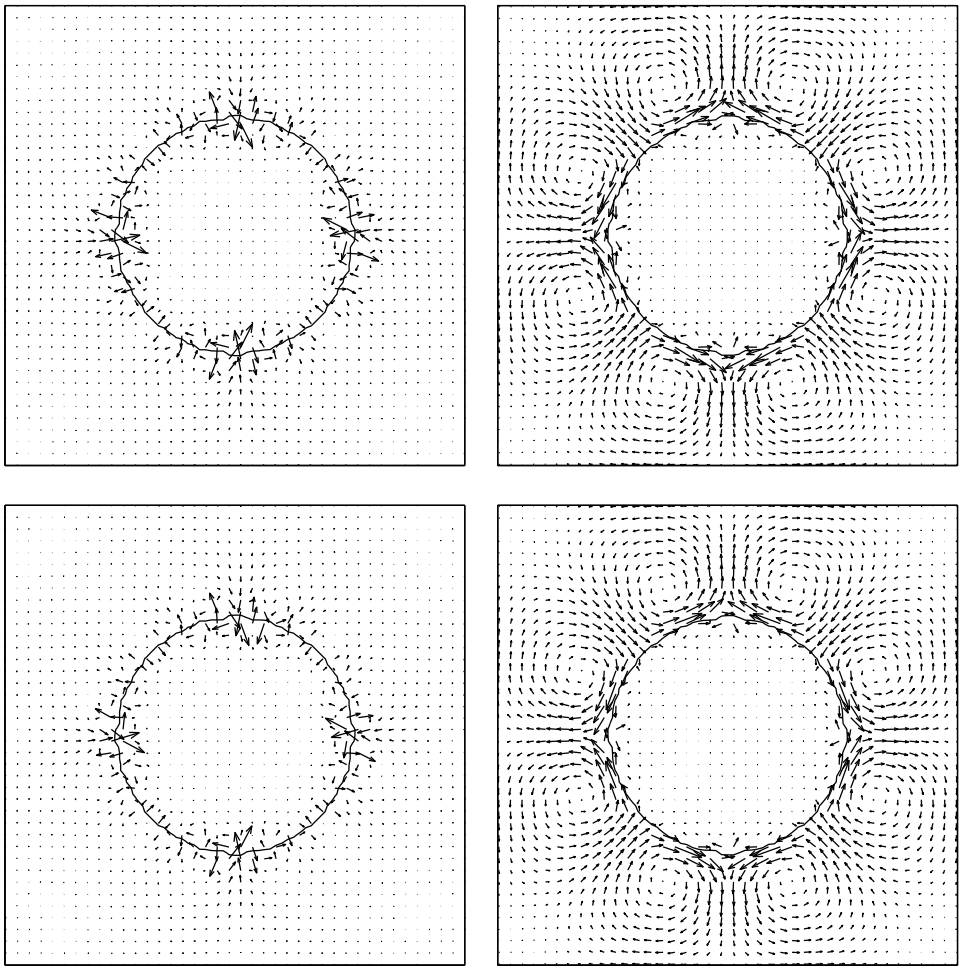


Figure 2.10: Velocity fields produced by the surface tension terms S (left) and $\widehat{S}$ (right) in bubble tests with $\sigma = 0.005$ (top) and $\sigma = 0.01$ (bottom). All computations were performed on a 80×80 lattice.

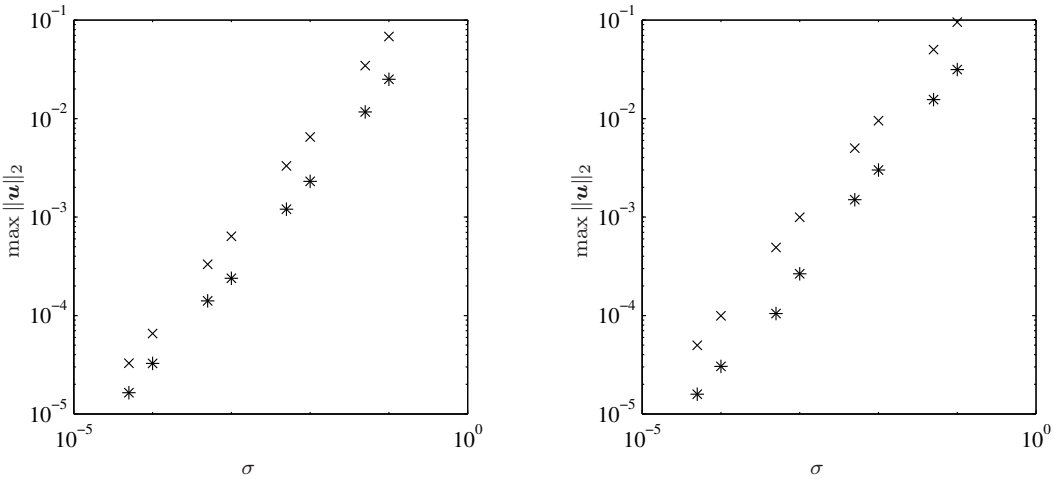


Figure 2.11: Comparison of $\max \|\boldsymbol{u}\|_2$ produced by bubble tests with surface tension terms S ('$*$') and $\widehat{S}$ ('$\times$') on a 40×40 (left) and an 80×80 (right) lattice.

Lattice	σ	Mass loss in %	Number of flops
40×40	0.00005	$6.96088 \cdot 10^{-8}$	$4.44087 \cdot 10^{9}$
40×40	0.0001	$7.14132 \cdot 10^{-8}$	$4.44087 \cdot 10^{9}$
40×40	0.0005	$7.17161 \cdot 10^{-8}$	$4.44085 \cdot 10^{9}$
40×40	0.001	$7.17161 \cdot 10^{-8}$	$4.44085 \cdot 10^{9}$
40×40	0.005	$7.38076 \cdot 10^{-8}$	$4.44085 \cdot 10^{9}$
40×40	0.01	$7.47666 \cdot 10^{-8}$	$4.44085 \cdot 10^{9}$
40×40	0.05	$7.34006 \cdot 10^{-8}$	$4.44081 \cdot 10^{9}$
40×40	0.1	$7.28928 \cdot 10^{-8}$	$4.44074 \cdot 10^{9}$
80×80	0.00005	$1.30909 \cdot 10^{-7}$	$1.00613 \cdot 10^{11}$
80×80	0.0001	$1.31692 \cdot 10^{-7}$	$1.00613 \cdot 10^{11}$
80×80	0.0005	$1.31763 \cdot 10^{-7}$	$1.00613 \cdot 10^{11}$
80×80	0.001	$1.32412 \cdot 10^{-7}$	$1.00613 \cdot 10^{11}$
80×80	0.005	$1.32478 \cdot 10^{-7}$	$1.00613 \cdot 10^{11}$
80×80	0.01	$1.32132 \cdot 10^{-7}$	$1.00613 \cdot 10^{11}$
80×80	0.05	$1.28786 \cdot 10^{-7}$	$1.00612 \cdot 10^{11}$
80×80	0.1	$1.28483 \cdot 10^{-7}$	$1.00612 \cdot 10^{11}$

Table 2.1: Percentage loss of red mass and computing times in the bubble tests with surface tension term S.

Lattice	σ	Mass loss in %	Number of flops
40×40	0.00005	$7.25899 \cdot 10^{-8}$	$4.39853 \cdot 10^{9}$
40×40	0.0001	$7.22619 \cdot 10^{-8}$	$4.39852 \cdot 10^{9}$
40×40	0.0005	$7.33344 \cdot 10^{-8}$	$4.39852 \cdot 10^{9}$
40×40	0.001	$7.26404 \cdot 10^{-8}$	$4.39850 \cdot 10^{9}$
40×40	0.005	$7.42429 \cdot 10^{-8}$	$4.39850 \cdot 10^{9}$
40×40	0.01	$7.41135 \cdot 10^{-8}$	$4.39849 \cdot 10^{9}$
40×40	0.05	$7.43217 \cdot 10^{-8}$	$4.39847 \cdot 10^{9}$
40×40	0.1	$7.14921 \cdot 10^{-8}$	$4.39836 \cdot 10^{9}$
80×80	0.00005	$1.35872 \cdot 10^{-7}$	$1.00290 \cdot 10^{11}$
80×80	0.0001	$1.36000 \cdot 10^{-7}$	$1.00290 \cdot 10^{11}$
80×80	0.0005	$1.36397 \cdot 10^{-7}$	$1.00290 \cdot 10^{11}$
80×80	0.001	$1.36386 \cdot 10^{-7}$	$1.00290 \cdot 10^{11}$
80×80	0.005	$1.34733 \cdot 10^{-7}$	$1.00290 \cdot 10^{11}$
80×80	0.01	$1.33235 \cdot 10^{-7}$	$1.00290 \cdot 10^{11}$
80×80	0.05	$1.31966 \cdot 10^{-7}$	$1.00290 \cdot 10^{11}$
80×80	0.1	$1.29152 \cdot 10^{-7}$	$1.00289 \cdot 10^{11}$

Table 2.2: Percentage loss of red mass and computing times in the bubble tests with surface tension term $\widehat{S}$.

CHAPTER 3

Some numerical experiments

Since ILBGK interface tracking achieved very reasonable results for the essentially one-dimensional model problem discussed in Section 2.3.3 we now want to try this method for really two-dimensional flow. Therefore, we first define a benchmark in Section 3.1, then we simulate it with ILBGK in Section 3.2. Because we find that the only shortcoming of this method are spurious oscillations in the run of the interface we try two approaches for smoothing those oscillations, namely adding anisotropic diffusion to the interface as well as applying artificial surface tension. Those approaches are dealt with in Sections 3.3 and 3.4, respectively.

3.1. A two-dimensional benchmark

In this Section, we shall define a benchmark test for interface tracking in two space dimensions. To stay in the setting of Chapters 1 and 2, we force all quantities except the indicator function χ to be smooth throughout the computational domain. This restriction guarantees a thorough investigation of interface tracking which is not disturbed by numerical difficulties due to jump singularities. Note that the benchmark is chosen with regard to comparison of numerical schemes and not with respect to physical relevance.

Let the domain $\Omega = [0, 2L] \times [0, 2L] \subset \mathbb{R}^2$ contain the open square bubble $\Omega^r = (L/2, 3L/2) \times (L/2, 3L/2)$ of red fluid and let $\Omega^b = \Omega \setminus \overline{\Omega^r}$ be the domain of the blue fluid. We denote the interface between Ω^r and Ω^b by Γ and the unit normal vector of Γ that points into Ω^r by $\boldsymbol{n}$. The whole framework of the benchmark is illustrated in Figure 3.1.

In the setting described above, we assume both fluids have the same constant density ϱ and constant dynamic viscosity μ and suppose their respective flow is governed by the two-phase incompressible Navier-Stokes equations (2.31). Additionally, we consider the transport equation (2.13) for χ. We will perform numerical experiments on 40×40, 80×80, and sometimes 160×160 lattices with initial velocity fields $\boldsymbol{u}^{(1)}(0, \boldsymbol{x}) = (0.1, 0)^T$, $\boldsymbol{u}^{(2)}(0, \boldsymbol{x}) = (0.1, 0.1)^T$, and $\boldsymbol{u}^{(3)}(0, \boldsymbol{x}) = C_{\boldsymbol{u}^{(3)}} \cdot (-x_2, x_1)^T$ as well as surface tension coefficients $\sigma \in \{0, 0.1, 0.01, 0.001\}$. The initial velocity fields are sketched in Figure 3.2. In the definition of $\boldsymbol{u}^{(3)}$, the point $(0, 0)^T$ is assumed to be the centre of the bubble and, therefore, of Ω while the constant $C_{\boldsymbol{u}^{(3)}}$ is given by

$$C_{\boldsymbol{u}^{(3)}} = \begin{cases} \pi/500 & \text{on the } 40 \times 40 \text{ lattice} \\ \pi/1000 & \text{on the } 80 \times 80 \text{ lattice} \\ \pi/2000 & \text{on the } 160 \times 160 \text{ lattice} \end{cases}$$

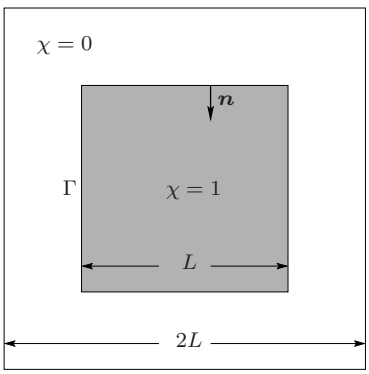


FIGURE 3.1: The initial situation for the two-dimensional benchmark: Ω^r and Ω^b are filled with red respectively blue fluid, $\chi = 1$ in Ω^r and $\chi = 0$ in Ω^b, $\Gamma = \partial\Omega^r$, and $\boldsymbol{n}$ is the unit normal vector of Γ pointing into Ω^r.

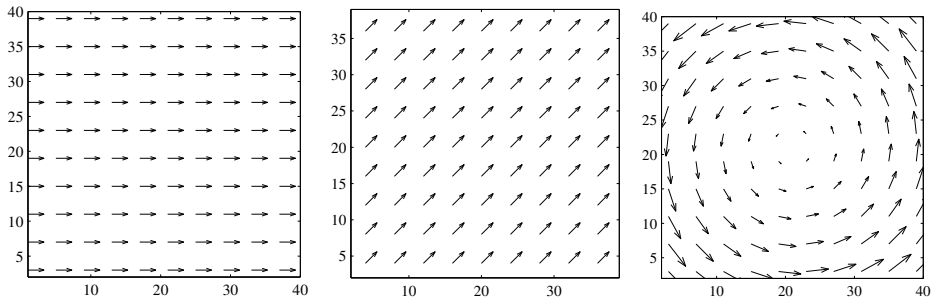


Figure 3.2: The initial velocity fields $\boldsymbol{u}^{(1)}$ (left), $\boldsymbol{u}^{(2)}$ (middle), and $\boldsymbol{u}^{(3)}$ (right) of the benchmark

which yields $\max_{\boldsymbol{x}\in\mathfrak{X}} \left\|\boldsymbol{u}^{(3)}\right\|_2 \approx 0.177715$. However, compressibility effects are not a crucial point in the investigations to follow, so maximum flow speed is not as important in the given situation as it is in other lattice BGK simulations.

Inserting the above velocity fields into the Navier-Stokes equations (2.31) we find that the continuity equation $\partial_\alpha u_\alpha = 0$ is always fulfilled while the momentum equation takes the form

$$\partial_\alpha p = \begin{cases} \sigma\partial_\beta(\delta_{\alpha\beta} - n_\alpha n_\beta) & \text{for } \boldsymbol{u} \in \left\{\boldsymbol{u}^{(1)}, \boldsymbol{u}^{(2)}\right\} \\ \varrho C^2_{\boldsymbol{u}^{(3)}} x_\alpha + \sigma\partial_\beta(\delta_{\alpha\beta} - n_\alpha n_\beta) & \text{for } \boldsymbol{u} = \boldsymbol{u}^{(3)}. \end{cases}$$

In the case $\sigma = 0$, this reduces to $\partial_\alpha p = 0$ respectively $\partial_\alpha p = \varrho C^2_{\boldsymbol{u}^{(3)}} x_\alpha$. Thus, for $\sigma = 0$ each of the initial velocity fields $\boldsymbol{u}^{(1)}$, $\boldsymbol{u}^{(2)}$, $\boldsymbol{u}^{(3)}$ is a stationary solution of the incompressible Navier-Stokes equations (2.31) if we choose the initial pressure field

$$p(0,\boldsymbol{x}) = 0 \quad \text{for } \boldsymbol{u} \in \left\{\boldsymbol{u}^{(1)}, \boldsymbol{u}^{(2)}\right\} \qquad \text{respectively} \qquad p(0,\boldsymbol{x}) = \frac{\varrho C^2_{\boldsymbol{u}^{(3)}}}{2} x_\alpha x_\alpha \quad \text{for } \boldsymbol{u} = \boldsymbol{u}^{(3)}. \tag{3.1}$$

In other words, this choice of initial pressure guarantees that neither $\boldsymbol{u}$ nor p changes in time if surface tension is neglected. For this reason, choosing $\sigma = 0$, $\boldsymbol{u}(0,\boldsymbol{x}) \in \left\{\boldsymbol{u}^{(1)}, \boldsymbol{u}^{(2)}, \boldsymbol{u}^{(3)}\right\}$, and $p(0,\boldsymbol{x})$ according to (3.1) we reduce the Navier-Stokes equations (2.31) to a model of plain interface tracking. Note, however, that this is no longer true for non-zero surface tension.

Let us now assume that $\boldsymbol{u}$ is constant in time. Then, choosing $\boldsymbol{u} = \boldsymbol{u}^{(3)}$ we force the bubble to perform exactly one revolution by running 1000 time steps on the 40×40 lattice, 2000 time steps on the 80×80 lattice, respectively 4000 time steps on the 160×160 lattice. For $\boldsymbol{u} \in \left\{\boldsymbol{u}^{(1)}, \boldsymbol{u}^{(2)}\right\}$ and periodic boundary conditions, the bubble will also perform one revolution if we stop the simulation after 400 time steps on the 40×40 lattice, 800 time steps on the 80×80 lattice, respectively 1600 time steps on the 160×160 lattice. Therefore, the final state of the benchmark will always be identical to the initial situation.

3.2. Plain ILBGK

Here, we consider plain ILBGK interface tracking according to Algorithm 2.9. Since the velocity field is not affected by stand-alone interface tracking we always know the analytical solution of the benchmark, as explained in Section 3.1. Furthermore, we know from Section 3.1 that for $\boldsymbol{u} \in \left\{\boldsymbol{u}^{(1)}, \boldsymbol{u}^{(2)}, \boldsymbol{u}^{(3)}\right\}$ plain interface tracking is equivalent to simulating Navier-Stokes flow with $\sigma = 0$ and pressure field p according to (3.1).

To get a deeper understanding of how ILBGK interface tracking works, we investigate the case of zero velocity before we consider the actual benchmark. Applying $u_\alpha(\boldsymbol{x}) = 0$ for each $\boldsymbol{x} \in \Omega$ we find ILBGK interface tracking to exhibit oscillatory behaviour in t, i. e.

$$\chi(m\Delta t, \boldsymbol{x}) \begin{cases} = \chi(0,\boldsymbol{x}) & \text{if } m \in \mathbb{N} \text{ is even} \\ \neq \chi(0,\boldsymbol{x}) & \text{if } m \in \mathbb{N} \text{ is odd,} \end{cases}$$

see Figure 3.3 for an illustration. These oscillations behave completely analogous to those already observed in Section 2.3.3.

Let us now examine the motion of the square bubble in the velocity fields $\boldsymbol{u}^{(1)}$ (Figure 3.4), $\boldsymbol{u}^{(2)}$ (Figure 3.5), and $\boldsymbol{u}^{(3)}$ (Figure 3.6). For $\boldsymbol{u} = \boldsymbol{u}^{(1)}$, ILBGK advection produces heavy oscillations along the interface sections perpendicular to $\boldsymbol{u}^{(1)}$. Those oscillations decrease with increasing lattice resolution. However, fine lattice computations are expensive, so we should try to avoid or, at least, to damp those

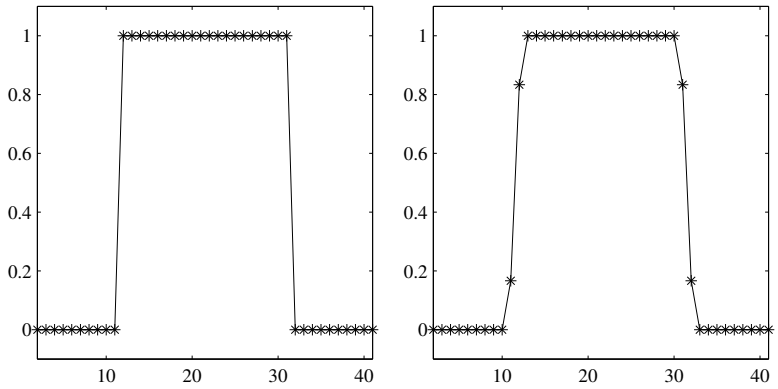


Figure 3.3: Behaviour of a static square bubble on a 40×40 lattice under the influence of the ILBGK interface tracking Algorithm 2.9 for $u_\alpha = 0$. The plots show a cut through the centre of the simulation result, i. e. the fortieth line of the lattice, after 400 time steps (left) and after 401 time steps (right). In the case of zero velocity, the square bubble always oscillates between those two states.

Lattice	Velocity	Mass loss in %	Number of flops
40×40	$\boldsymbol{u}^{(1)}$	$1.70530 \cdot 10^{-13}$	$2.07773 \cdot 10^{8}$
40×40	$\boldsymbol{u}^{(2)}$	$1.29319 \cdot 10^{-12}$	$2.07769 \cdot 10^{8}$
40×40	$\boldsymbol{u}^{(3)}$	$-3.26850 \cdot 10^{-13}$	$5.19433 \cdot 10^{8}$
80×80	$\boldsymbol{u}^{(1)}$	$2.13163 \cdot 10^{-13}$	$1.59739 \cdot 10^{9}$
80×80	$\boldsymbol{u}^{(2)}$	$1.90425 \cdot 10^{-12}$	$1.59743 \cdot 10^{9}$
80×80	$\boldsymbol{u}^{(3)}$	$-9.66338 \cdot 10^{-13}$	$3.99364 \cdot 10^{9}$
160×160	$\boldsymbol{u}^{(1)}$	$2.13163 \cdot 10^{-13}$	$1.25263 \cdot 10^{10}$
160×160	$\boldsymbol{u}^{(2)}$	$3.01270 \cdot 10^{-12}$	$1.25266 \cdot 10^{10}$
160×160	$\boldsymbol{u}^{(3)}$	$-3.26850 \cdot 10^{-12}$	$3.13169 \cdot 10^{10}$

Table 3.1: Percentage loss of red mass and computing times for the simulations shown in Figures 3.4 to 3.6

oscillations. For $\boldsymbol{u} = \boldsymbol{u}^{(2)}$, the spurious oscillations are even worse, covering now the complete interface. At least, they can again be damped by using finer lattices. Lastly, concerning rotation in the velocity field $\boldsymbol{u} = \boldsymbol{u}^{(3)}$, simulation results are most unsatisfactory. Nevertheless, interpreting χ as a pseudo density of the red fluid we find that red mass is perfectly conserved in plain ILBGK interface tracking, as can be seen from Table 3.1. For the sake of completeness, this table also shows the number of floating point operations needed for the different simulations.

It is interesting to observe that keeping the nonlinear term

$$f^*(\boldsymbol{v}) \frac{9\Delta x^2}{2} \chi u_\alpha u_\beta \left(v_\alpha v_\beta - \frac{\delta_{\alpha\beta}}{3} \right)$$

in the equilibrium distribution (2.60) we obtain irregularities away from the interface in the solution of the benchmark problem with $\boldsymbol{u} = \boldsymbol{u}^{(3)}$, see Figure 3.7 for an illustration. Those irregularities represent errors in χ which are considerably larger than $\mathcal{O}\left(\Delta x^2\right)$. Therefore, they cannot be identified with the numerical error produced by the transport scheme (2.62). One possible explanation for their appearance is the recolouring step trying to produce sharp interfaces from small perturbations in the numerical approximation of the indicator function χ. Such a problem does not appear for $\boldsymbol{u} \in \left\{\boldsymbol{u}^{(1)}, \boldsymbol{u}^{(2)}\right\}$, so the velocity field $\boldsymbol{u}^{(3)}$ (which is non-constant in space and features curved streamlines) is most likely to be responsible for perturbing χ. However, there is no rigorous proof of this conjecture, so the explanation of the spurious irregularities of χ must be considered an unsolved problem. We shall face similar difficulties in Sections 3.4 and 5.3.

Note that neither using a symmetrised variant of the recolouring step [**21**] nor decreasing the flow speed results in a considerable improvement of ILBGK interface tracking, corresponding numerical results are shown in Figures 3.8 and 3.9, respectively. Note also that to overcome the perturbation of the interface, D'Ortona et al. [**14**] as well as Tölke [**78**] proposed special modifications of the recolouring step. The

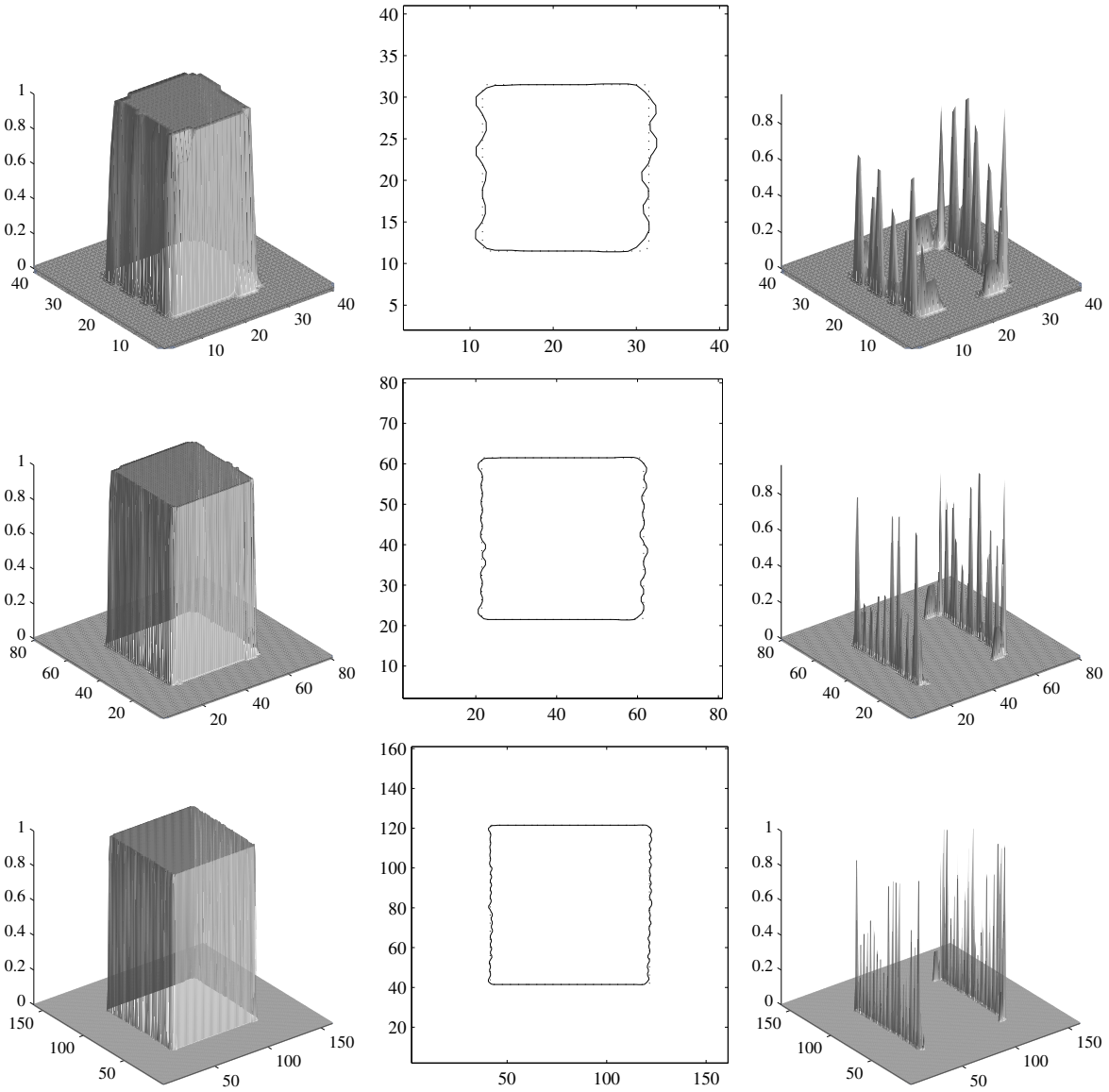


Figure 3.4: Solution of the benchmark problem for $\boldsymbol{u} = \boldsymbol{u}^{(1)}$ with Algorithm 2.9 on a 40×40 (top line), 80×80 (centre line), and 160×160 (bottom line) lattice, all pictures taken after one complete revolution. Left: simulation result, middle: isoline for $\chi = 1/2$ of the true solution (dotted line) and the ILBGK approximation (solid line), right: absolute value of the Euclidean error in the ILBGK approximation.

idea behind both those approaches is to reduce the perturbation of the interface by spreading it over a wider range of lattice nodes. Here, however, we are exclusively interested in sharp interfaces, so we do not consider this idea any further.

3.3. Anisotropic diffusion

In Section 2.3.1, we learned that recolouring serves as artificial anti-diffusion for the viscous numerical scheme (2.61). Unfortunately, for really two-dimensional flow this kind of anti-diffusion has the bad side-effect of disturbing the interface. Since everything works well if the flow is essentially one-dimensional (compare Section 2.3.3) we claim that those defects are not generated by too much anti-diffusion across the interface, which should cause trouble in 1-d, too, but maybe they are provoked by a lack of diffusion along the interface. Thus, it is a natural idea to seek help in the replacement of the recolouring Algorithm 2.8

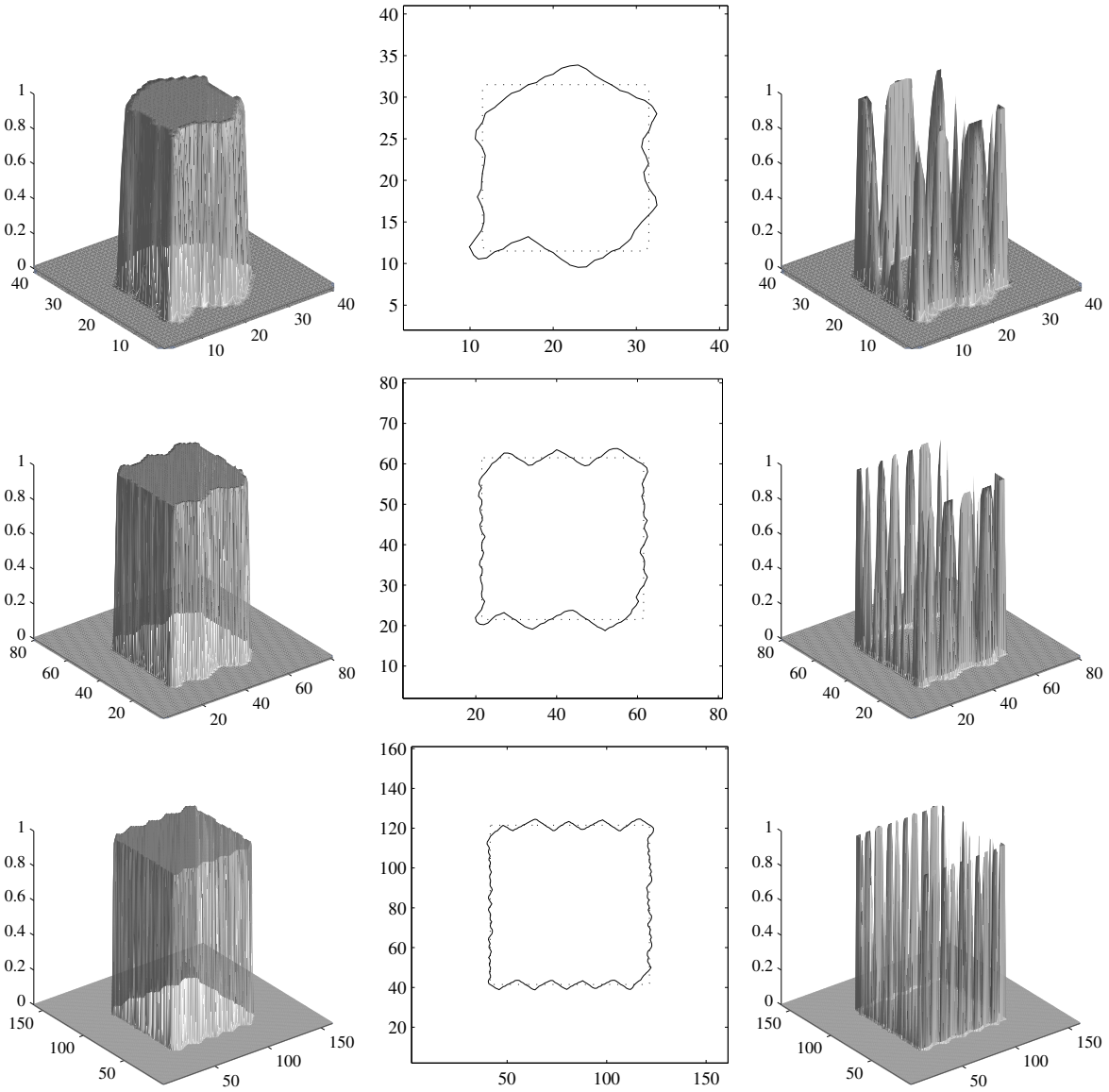


Figure 3.5: Solution of the benchmark problem for $\boldsymbol{u} = \boldsymbol{u}^{(2)}$ with Algorithm 2.9 on a 40×40 (top line), 80×80 (centre line), and 160×160 (bottom line) lattice, all pictures taken after one complete revolution. Left: the simulation result, middle: isoline for $\chi = 1/2$ of the true solution (dotted line) and the ILBGK approximation (solid line), right: absolute value of the Euclidean error in the ILBGK approximation.

with an anisotropic diffusion procedure that applies diffusion along the interface and anti-diffusion across it.

The theory of anisotropic diffusion has been developed in recent years as a powerful tool for image processing. A very good overview of the theory and its application to images is given by Weickert [**79**]. Grahs et al. [**25**] use anisotropic diffusion for the numerical computation of shock solutions of nonlinear hyperbolic conservation laws, and Jawerth et al. [**39**] published two lattice Boltzmann models of the anisotropic diffusion process. However, we do not use one of their models but directly implement a numerical scheme developed by Weickert [**79**].

In the remainder of this Section, we will try to annihilate the negative effects of the spurious diffusion term in the transport equation (2.63) by adopting the ideas of Weickert [**79**] and of Grahs et al [**25**].

3.3.1. Theoretical Background. The basic idea is to solve a nonlinear diffusion equation of the form

$$\partial_{\hat{t}} \hat{\chi} = \partial_\alpha \left(d_{\alpha\beta} \partial_\beta \hat{\chi} \right) \tag{3.2}$$

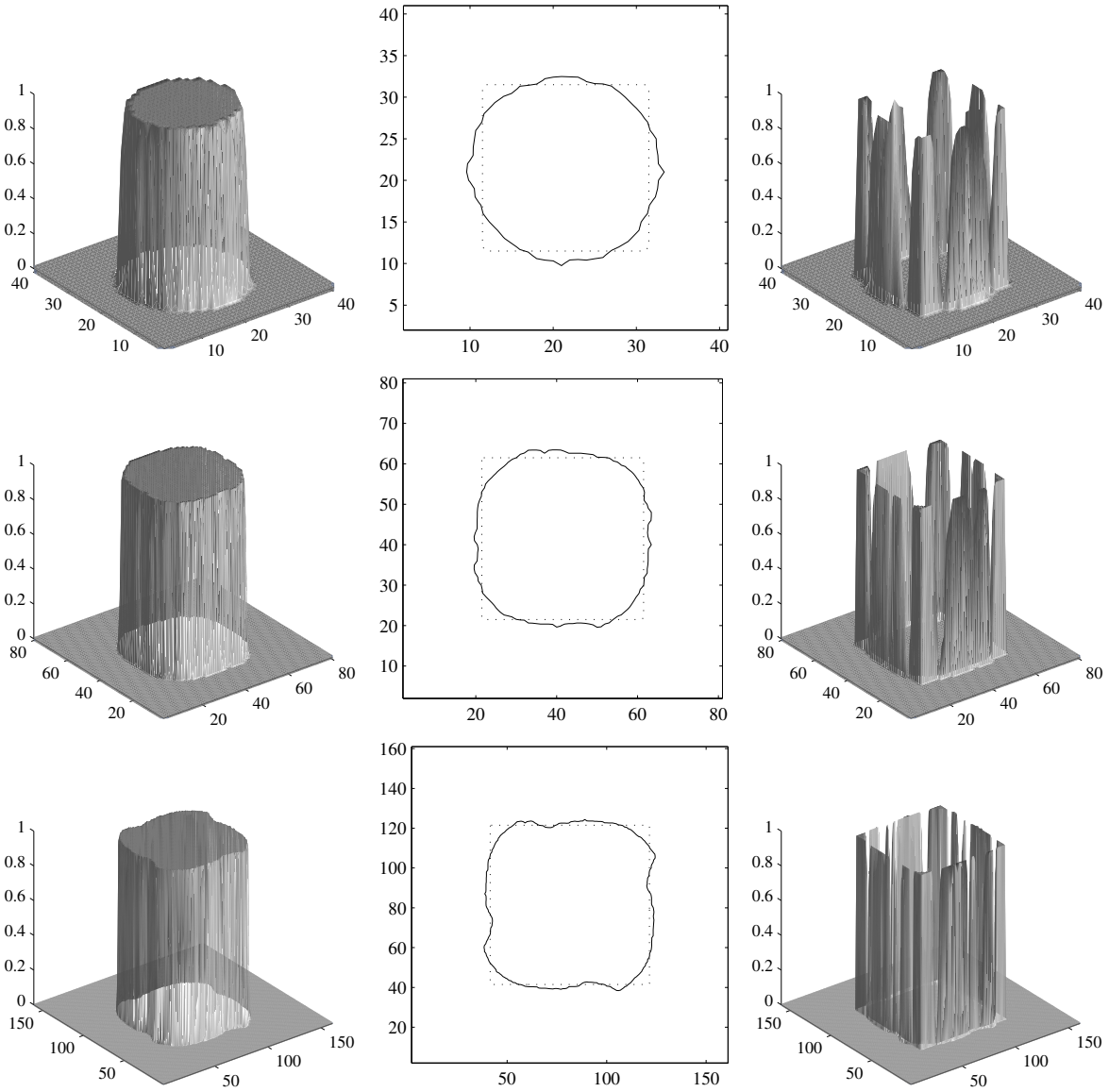


Figure 3.6: Solution of the benchmark problem for $\boldsymbol{u} = \boldsymbol{u}^{(3)}$ with Algorithm 2.9 on a 40×40 (top line), 80×80 (centre line), and 160×160 (bottom line) lattice, all pictures taken after one complete revolution. Left: the simulation result, middle: isoline for $\chi = 1/2$ of the true solution (dotted line) and the ILBGK approximation (solid line), right: absolute value of the Euclidean error in the ILBGK approximation.

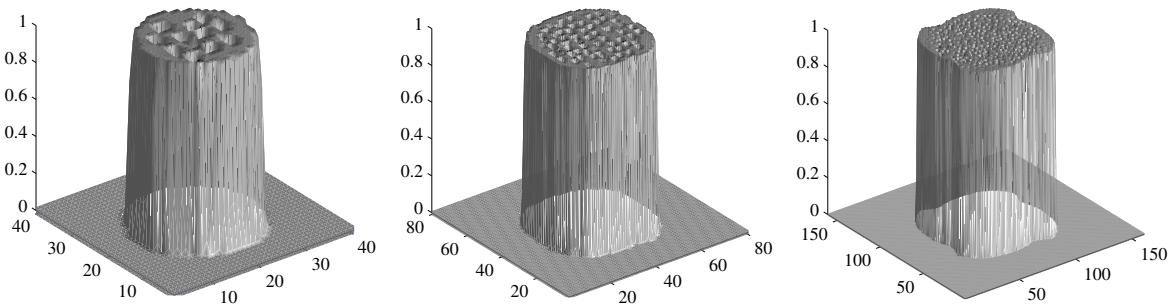


Figure 3.7: Solution of the benchmark problem for $\boldsymbol{u} = \boldsymbol{u}^{(3)}$ with Algorithm 2.9 on a 40×40 (left), 80×80 (middle), and 160×160 (right) lattice if the nonlinear term in the equilibrium distribution is kept.

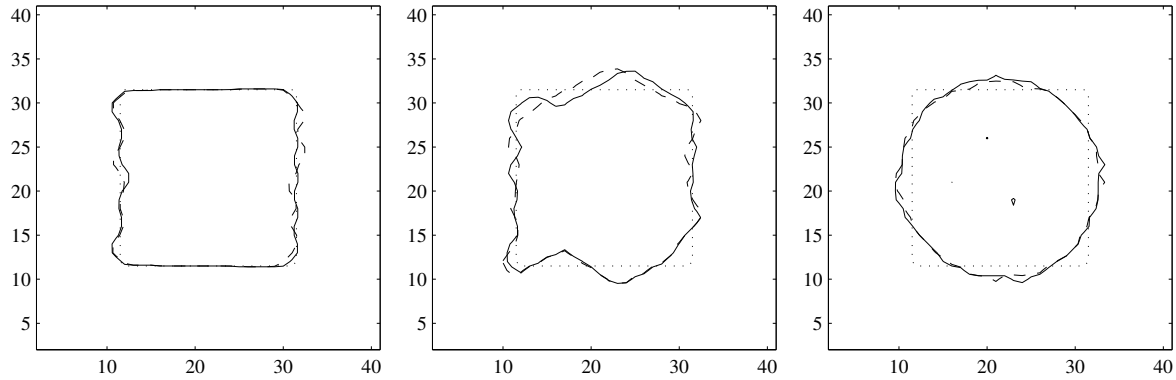


Figure 3.8: Solution of the benchmark problem for $\boldsymbol{u} = \boldsymbol{u}^{(1)}$ (left), $\boldsymbol{u} = \boldsymbol{u}^{(2)}$ (middle), and $\boldsymbol{u} = \boldsymbol{u}^{(3)}$ (right). All pictures show the level line for $\chi = 1/2$ of the analytical solution (dotted line), the ILBGK approximation (dashed line), and the ILBGK result with symmetrised recolouring (solid line) after one complete revolution.

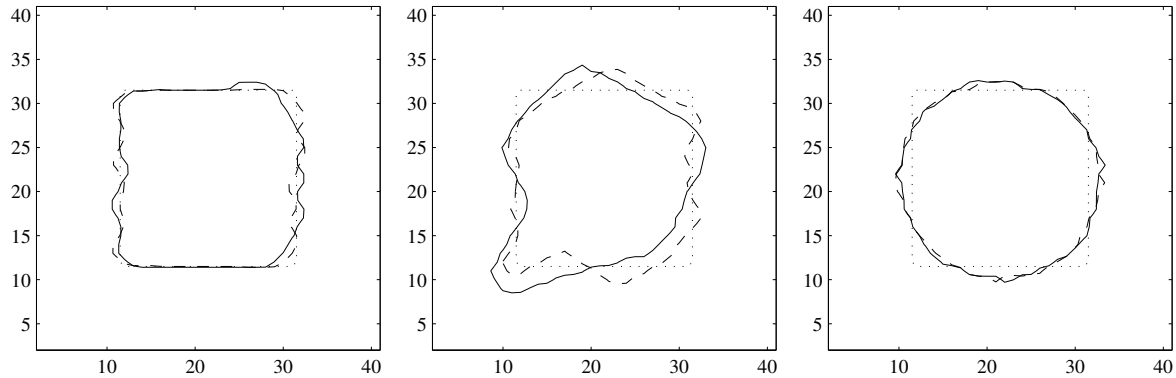


Figure 3.9: Solution of the benchmark problem for $\boldsymbol{u} = \boldsymbol{u}^{(1)}$ (left), $\boldsymbol{u} = \boldsymbol{u}^{(2)}$ (middle), and $\boldsymbol{u} = \boldsymbol{u}^{(3)}$ (right). All pictures show the level line for $\chi = 1/2$ of the analytical solution (dotted line), the ILBGK approximation (dashed line), and the ILBGK result computed with flow speed $\boldsymbol{u}/10$ (solid line) after one complete revolution.

with initial condition $\hat{\chi}(0, \boldsymbol{x}) = \chi(t, \boldsymbol{x})$ until a suitable value of $\hat{t}$ is reached and then use $\hat{\chi}$ as a smoothed version of χ. The diffusion tensor $d_{\alpha\beta}$ is defined such that it produces forward diffusion along the smeared jump in $\hat{\chi}$, i. e. along the interface Γ, and backward diffusion across it. Note that the variable $\hat{t}$ plays the role of time in the sense that it measures the duration of the diffusion process. However, it does not represent physical time t. In the following, we will give a short outline of the techniques involved in putting up $d_{\alpha\beta}$ and solving equation (3.2). See Weickert's book [**79**] and references therein for more details.

We start this overview with defining the structure tensor

$$s^0_{\alpha\beta}(\boldsymbol{x}) = \partial_\alpha \hat{\chi}(\boldsymbol{x}) \partial_\beta \hat{\chi}(\boldsymbol{x}).$$

This tensor implies an orthonormal basis $\left\{\boldsymbol{w}^0_1, \boldsymbol{w}^0_2\right\}$ of $\mathbb{R}^2$ consisting of the eigenvectors $\boldsymbol{w}^0_1 \parallel \nabla\hat{\chi}$ and $\boldsymbol{w}^0_2 \perp \nabla\hat{\chi}$ of $\boldsymbol{s}^0$, where $\nabla = (\partial_1, \partial_2)^T$. The corresponding eigenvalues $\partial_\alpha\hat{\chi}\partial_\alpha\hat{\chi}$ respectively 0 measure the steepness of $\hat{\chi}$ in the eigendirections of $\boldsymbol{s}^0$. To avoid any trouble with jumps, we apply Gaußian smoothing to $s^0_{\alpha\beta}$, i. e. instead of $s^0_{\alpha\beta}$ we use the smoothed structure tensor

$$s^\vartheta_{\alpha\beta}(\boldsymbol{x}) = \Big(G_\vartheta * (\partial_\alpha\hat{\chi}\partial_\beta\hat{\chi})\Big)(\boldsymbol{x}), \tag{3.3}$$

where for any $g \in \mathcal{L}^1\left(\mathbb{R}^2, \mathbb{R}\right)$

$$\left(G_\vartheta * g\right)(\boldsymbol{x}) = \int_{\mathbb{R}^2} G_\vartheta(\boldsymbol{x} - \boldsymbol{y}) g(\boldsymbol{y})\, d\boldsymbol{y} \tag{3.4}$$

denotes the convolution of g with the Gaußian G_ϑ given by

$$G_\vartheta(\boldsymbol{x}) = \frac{1}{2\pi\vartheta^2}\exp\left(-\frac{x_\alpha x_\alpha}{2\vartheta^2}\right) \tag{3.5}$$

with standard deviation $\vartheta > 0$. Note that Gaußian smoothing is equivalent to solving the linear diffusion equation

$$\partial_{\tilde{t}}\tilde{\chi} = \partial_{\alpha\alpha}\tilde{\chi} \tag{3.6}$$

with initial condition $\tilde{\chi}(0,\boldsymbol{x}) = \hat{\chi}(\hat{t},\boldsymbol{x})$ until pseudo time $\tilde{t} = \vartheta^2/2$ and that ϑ is therefore a measure for the intensity of the smoothing.

The symmetric matrix $\boldsymbol{s}^\vartheta \in \mathbb{R}^{2\times 2}$ is positive semi-definite and has two orthonormal eigenvectors,

$$\boldsymbol{w}_1^\vartheta \parallel \begin{pmatrix} 2s_{12}^\vartheta \\ s_{22}^\vartheta - s_{11}^\vartheta + \sqrt{\left(s_{11}^\vartheta - s_{22}^\vartheta\right)^2 + 4\left(s_{12}^\vartheta\right)^2} \end{pmatrix} \qquad \text{and} \qquad \boldsymbol{w}_2^\vartheta \perp \boldsymbol{w}_1^\vartheta. \tag{3.7}$$

The corresponding eigenvalues ν_1 and ν_2 are given by

$$\nu_{1,2} = \frac{1}{2}\left(s_{11}^\vartheta + s_{22}^\vartheta \pm \sqrt{\left(s_{11}^\vartheta - s_{22}^\vartheta\right)^2 + 4\left(s_{12}^\vartheta\right)^2}\right), \tag{3.8}$$

where the '+' in front of the square root belongs to ν_1 and the '−' belongs to ν_2. In order to achieve forward diffusion along the jump and backward diffusion across it, we define the diffusion tensor $\boldsymbol{d}$ such that it possesses the same eigenvectors $\boldsymbol{w}_1^\vartheta$ and $\boldsymbol{w}_2^\vartheta$ as the smoothed structure tensor $\boldsymbol{s}^\vartheta$ while we define its eigenvalues by

$$\lambda_1(\nu_1) = \Psi(\nu_1) \quad \text{and} \quad \lambda_2 = 1, \qquad \text{where} \qquad \Psi(\nu_1) = \begin{cases} 1 & \text{for } \nu_1 \le 0 \\ 1 - \exp\left(\frac{-C_\Psi}{(\nu_1/\nu)^4}\right) & \text{for } \nu_1 > 0. \end{cases} \tag{3.9}$$

The constant C_Ψ must be chosen such that the flux function $\Phi(\nu_1) = \nu_1\Psi(\nu_1)$ is increasing, i. e. $\partial_{\nu_1}\Phi > 0$, for $0 \le \nu_1 < \nu$ and decreasing, i. e. $\partial_{\nu_1}\Phi < 0$, for $\nu < \nu_1 < \infty$ because then, we have forward diffusion for $0 \le \nu_1 < \nu$ and backward diffusion for $\nu < \nu_1 < \infty$.

Weickert [**79**] claims without further explanation that (3.9) implies $C_\Psi \approx 3.31488$. However, we find that for $\nu_1 > 0$

$$\partial_{\nu_1}\Phi = 1 - \left(1 + 4C_\Psi\left(\frac{\nu_1}{\nu}\right)^{-4}\right)\exp\left(\frac{-C_\Psi}{(\nu_1/\nu)^{-4}}\right)$$

and therefore,

$$\left(\partial_{\nu_1}\Phi\right)\Big|_{\nu_1=\nu} = 1 - \left(1 + 4C_\Psi\right)e^{-C_\Psi}.$$

Requiring $\left(\partial_{\nu_1}\Phi\right)\Big|_{\nu_1=\nu} = 0$, which is necessary for a change in the sign of $\partial_{\nu_1}\Phi$ at $\nu_1 = \nu$, we obtain

$$C_\Psi \in \left\{0, W_{-1}\left(-\frac{1}{4}e^{-1/4}\right) - \frac{1}{4}\right\},$$

where W represents the Lambert W function [**11**]. The choice $C_\Psi = 0$ produces a flux function which is neither increasing for $0 \le \nu_1 < \nu$ nor decreasing for $\nu < \nu_1 < \infty$, so we chose

$$C_\Psi = W_{-1}\left(-\frac{1}{4}e^{-1/4}\right) - \frac{1}{4} \approx 2.33666$$

which differs significantly from Weickert's choice. Figure 3.10 shows plots of both Φ and Ψ with $C_\Psi = 2.33666$ and $C_\Psi = 3.31488$ for several values of ν. In the case $C_\Psi = 2.33666$, the change in the sign of $\partial_{\nu_1}\Phi$ always occurs at $\nu_1 = \nu$, so for this choice, Φ produces forward diffusion if $0 < \nu_1 < \nu$ and backward diffusion if $\nu < \nu_1 < \infty$ while the choice $C_\Psi = 3.31488$ slightly perturbs this relation. Even though this perturbation is barely relevant in practice, we shall use $C_\Psi = 2.33666$ for all the simulations below.

The parameter ν must also be chosen with care. If we choose it too large it causes too much forward diffusion perpendicular to the interface, so the jump in χ will smear too much, and if we choose it too small the rapid decrease of the flux function Φ (see again Figure 3.10) will reduce backward diffusion to some negligibly small amount. Obviously, it strongly depends on the given situation whether or not a certain value of ν is reasonable. For the problems considered below, numerical experiments yield that

$$\nu = \min\left\{\max_{\boldsymbol{x}}\{\nu_1\}, \max_{\boldsymbol{x}}\{\nu_2\}\right\} \tag{3.10}$$

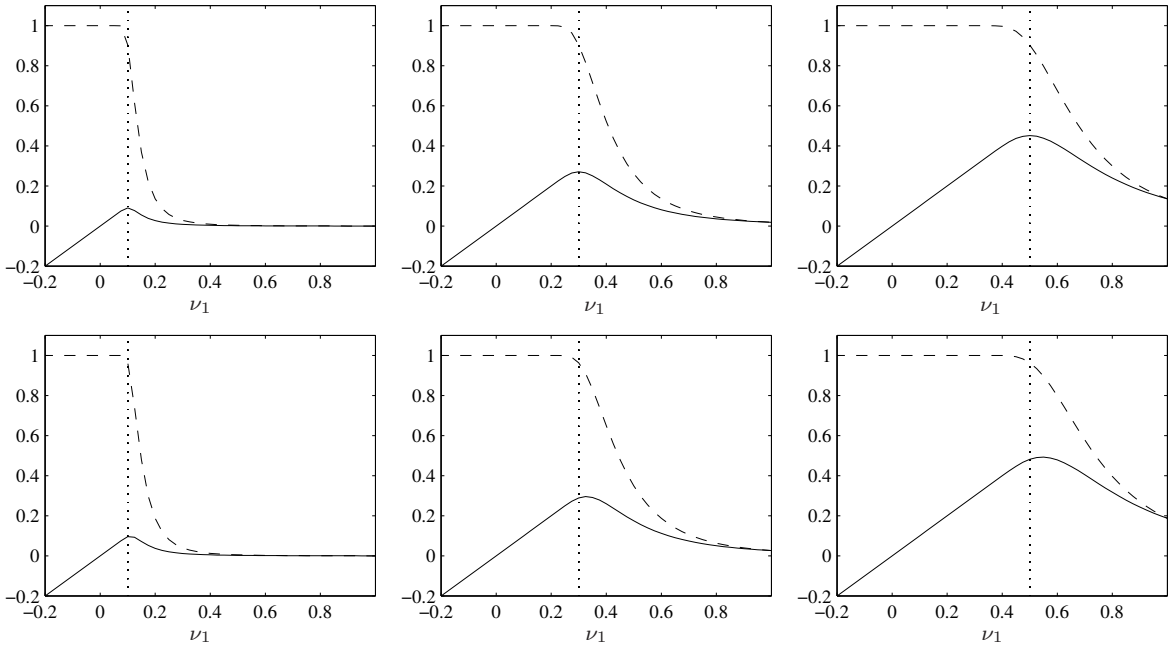


Figure 3.10: The functions Φ (solid line) and Ψ (dashed line) with $C_\Psi = 2.33666$ (top) and $C_\Psi = 3.31488$ (bottom) for $\nu = 0.1$ (left column), $\nu = 0.3$ (middle column), and $\nu = 0.5$ (right column) versus ν_1. In each plot, the axis of ordinates shows the function values, the abscissa represents ν_1, and the dotted vertical line marks the spot where $\nu_1 = \nu$.

is a proper choice.

At this stage, we are finally able to determine the diffusion tensor $\boldsymbol{d}$ by computing the inverse principal axis transform of $\operatorname{diag}(\lambda_1, \lambda_2)$, i. e.

$$d_{\alpha\beta} = w^{\vartheta}_{1;\alpha} w^{\vartheta}_{1;\beta} \lambda_1 + w^{\vartheta}_{2;\alpha} w^{\vartheta}_{2;\beta} \lambda_2. \tag{3.11}$$

This formula guarantees $\boldsymbol{d}$ to have the eigenvalues λ_1 and λ_2 as well as the eigenvectors $\boldsymbol{w}^{\vartheta}_1$ and $\boldsymbol{w}^{\vartheta}_2$. Therefore, it guarantees that $d_{\alpha\beta}$ has exactly the properties we want it to have and thus, we are now able to set up the anisotropic diffusion equation (3.2).

3.3.2. Numerical algorithm. In order to permit easy use of anisotropic diffusion in immiscible lattice BGK, we discretise equation (3.2) on a regular quadratic lattice with spacing Δx, as it is used in ILBGK itself. Weickert [**79**] showed that for any diffusion tensor $\boldsymbol{d}$ with suitable condition number, the finite difference stencil

$$\begin{bmatrix} k_4(\boldsymbol{x}) & k_3(\boldsymbol{x}) & k_2(\boldsymbol{x}) \\ k_5(\boldsymbol{x}) & k_0(\boldsymbol{x}) & k_1(\boldsymbol{x}) \\ k_6(\boldsymbol{x}) & k_7(\boldsymbol{x}) & k_8(\boldsymbol{x}) \end{bmatrix} \leftrightarrow \partial_\alpha \left(d_{\alpha\beta} \partial_\beta \right) + \mathcal{O}\left(\Delta x^2\right)$$

with

$$\begin{aligned} k_0(\boldsymbol{x}) = &- \frac{2d_{11}(\boldsymbol{x}) + d_{11}(\boldsymbol{x} + \boldsymbol{c}_1\Delta x) + d_{11}(\boldsymbol{x} + \boldsymbol{c}_5\Delta x)}{2\Delta x^2} + \frac{2\left|d_{12}(\boldsymbol{x})\right| + \left|d_{12}(\boldsymbol{x} + \boldsymbol{c}_1\Delta x)\right| + \left|d_{12}(\boldsymbol{x} + \boldsymbol{c}_3\Delta x)\right|}{2\Delta x^2} \\ &+ \frac{\left|d_{12}(\boldsymbol{x} + \boldsymbol{c}_5\Delta x)\right| + \left|d_{12}(\boldsymbol{x} + \boldsymbol{c}_7\Delta x)\right|}{2\Delta x^2} - \frac{\left|d_{12}(\boldsymbol{x} + \boldsymbol{c}_2\Delta x)\right| + d_{12}(\boldsymbol{x} + \boldsymbol{c}_2\Delta x)}{4\Delta x^2} \\ &- \frac{\left|d_{12}(\boldsymbol{x} + \boldsymbol{c}_4\Delta x)\right| - d_{12}(\boldsymbol{x} + \boldsymbol{c}_4\Delta x)}{4\Delta x^2} - \frac{\left|d_{12}(\boldsymbol{x} + \boldsymbol{c}_6\Delta x)\right| + d_{12}(\boldsymbol{x} + \boldsymbol{c}_6\Delta x)}{4\Delta x^2} \\ &- \frac{\left|d_{12}(\boldsymbol{x} + \boldsymbol{c}_8\Delta x)\right| - d_{12}(\boldsymbol{x} + \boldsymbol{c}_8\Delta x)}{4\Delta x^2} - \frac{2d_{22}(\boldsymbol{x}) + d_{22}(\boldsymbol{x} + \boldsymbol{c}_3\Delta x) + d_{22}(\boldsymbol{x} + \boldsymbol{c}_7\Delta x)}{2\Delta x^2}, \end{aligned} \tag{3.12a}$$

$$k_1(\boldsymbol{x}) = \frac{d_{11}(\boldsymbol{x} + \boldsymbol{c}_1\Delta x) + d_{11}(\boldsymbol{x}) - \left|d_{12}(\boldsymbol{x} + \boldsymbol{c}_1\Delta x)\right| - \left|d_{12}(\boldsymbol{x})\right|}{2\Delta x^2}, \tag{3.12b}$$

$$k_2(\boldsymbol{x}) = \frac{d_{12}(\boldsymbol{x} + \boldsymbol{c}_2\Delta x) + d_{12}(\boldsymbol{x}) + \left|d_{12}(\boldsymbol{x} + \boldsymbol{c}_2\Delta x)\right| + \left|d_{12}(\boldsymbol{x})\right|}{4\Delta x^2}, \tag{3.12c}$$

```
let χ̂(0, x) = χ(t, x);
at each lattice point,
    compute s^ϑ_αβ according to (3.3);
    compute ν_1,2 according to (3.8);
    compute ν according to (3.10);
    compute λ_1,2 according to (3.9);
    compute d_αβ according to (3.11);
perform t̂_max time steps of the scheme (3.12);
let χ(t, x) = χ̂(t̂_max, x);
```

Algorithm 3.1: Anisotropic diffusion

```
initialise;
while t < t_max
    determine χ(t + Δt, x) according to (2.62);
    do anisotropic diffusion according to Algorithm 3.1;
    t = t + Δt;
```

Algorithm 3.2: ILBGK interface tracking with anisotropic diffusion instead of recolouring

$$k_3(\boldsymbol{x}) = \frac{d_{22}(\boldsymbol{x}+\boldsymbol{c}_3\Delta x) + d_{22}(\boldsymbol{x}) - |d_{12}(\boldsymbol{x}+\boldsymbol{c}_3\Delta x)| - |d_{12}(\boldsymbol{x})|}{2\Delta x^2}, \tag{3.12d}$$

$$k_4(\boldsymbol{x}) = \frac{-d_{12}(\boldsymbol{x}+\boldsymbol{c}_4\Delta x) - d_{12}(\boldsymbol{x}) + |d_{12}(\boldsymbol{x}+\boldsymbol{c}_4\Delta x)| + |d_{12}(\boldsymbol{x})|}{4\Delta x^2}, \tag{3.12e}$$

$$k_5(\boldsymbol{x}) = \frac{d_{11}(\boldsymbol{x}+\boldsymbol{c}_5\Delta x) + d_{11}(\boldsymbol{x}) - |d_{12}(\boldsymbol{x}+\boldsymbol{c}_5\Delta x)| - |d_{12}(\boldsymbol{x})|}{2\Delta x^2}, \tag{3.12f}$$

$$k_6(\boldsymbol{x}) = \frac{d_{12}(\boldsymbol{x}+\boldsymbol{c}_6\Delta x) + d_{12}(\boldsymbol{x}) + |d_{12}(\boldsymbol{x}+\boldsymbol{c}_6\Delta x)| + |d_{12}(\boldsymbol{x})|}{4\Delta x^2}, \tag{3.12g}$$

$$k_7(\boldsymbol{x}) = \frac{d_{22}(\boldsymbol{x}+\boldsymbol{c}_7\Delta x) + d_{22}(\boldsymbol{x}) - |d_{12}(\boldsymbol{x}+\boldsymbol{c}_7\Delta x)| - |d_{12}(\boldsymbol{x})|}{2\Delta x^2}, \tag{3.12h}$$

and

$$k_8(\boldsymbol{x}) = \frac{-d_{12}(\boldsymbol{x}+\boldsymbol{c}_8\Delta x) - d_{12}(\boldsymbol{x}) + |d_{12}(\boldsymbol{x}+\boldsymbol{c}_8\Delta x)| + |d_{12}(\boldsymbol{x})|}{4\Delta x^2} \tag{3.12i}$$

leads to a stable numerical scheme for the anisotropic diffusion equation (3.2), namely

$$\hat{\chi}\left(\hat{t}+\Delta\hat{t},\boldsymbol{x}\right) = \hat{\chi}\left(\hat{t},\boldsymbol{x}\right) + \Delta\hat{t}\begin{bmatrix} k_4(\boldsymbol{x}) & k_3(\boldsymbol{x}) & k_2(\boldsymbol{x}) \\ k_5(\boldsymbol{x}) & k_0(\boldsymbol{x}) & k_1(\boldsymbol{x}) \\ k_6(\boldsymbol{x}) & k_7(\boldsymbol{x}) & k_8(\boldsymbol{x}) \end{bmatrix}\hat{\chi}\left(\hat{t},\boldsymbol{x}\right) \tag{3.12j}$$

if the pseudo-time step size $\Delta\hat{t}$ is properly chosen. Computing $d_{\alpha\beta}$ according to (3.11) and approximating the anisotropic diffusion equation (3.2) by scheme (3.12) we obtain the anisotropic diffusion Algorithm 3.1. Note that a suitable stopping time $\hat{t}_{max}$ is still to be determined.

3.3.3. Anisotropic diffusion instead of recolouring. In the following, we perform some experiments with ILBGK interface tracking, where the recolouring step (Algorithm 2.8) is replaced with anisotropic diffusion (Algorithm 3.1), as outlined in Algorithm 3.2. In particular, we perform one time step of this algorithm for the benchmark problem with $u_\alpha = 0$. In Figures 3.11, 3.12, and 3.13, we present results computed on a 40×40, 80×80, and 160×160 lattice, respectively, with one, 100, and 2000 pseudo-time steps of anisotropic diffusion.

In analogy to the one-dimensional case (compare Section 2.3.3), we observe that during one time step of the transport scheme (2.62), a certain part of the red mass located at those lattice points $\boldsymbol{x} \in \mathcal{X} \cap \Omega^r$ next to the interface diffuses into the blue phase. In particular, away from the corners of Ω^r, exactly

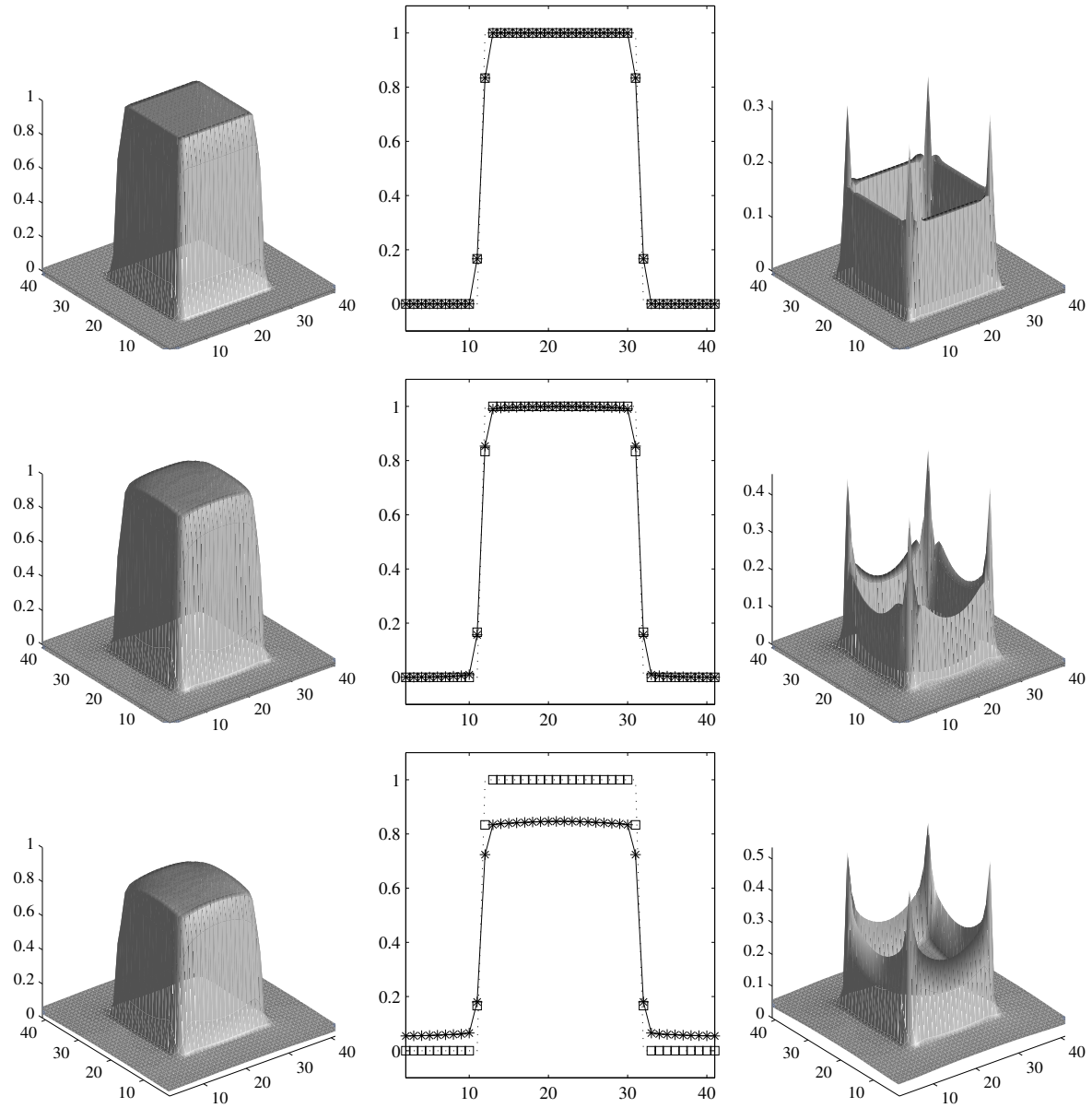


Figure 3.11: Solution of the benchmark problem with Algorithm 3.2 on a 40×40 lattice for $u_\alpha = 0$ and one (top line), 100 (centre line), respectively 2000 (bottom line) iterations of Algorithm 3.1. Left: surface plot of the indicator function χ, middle: horizontal cut through χ (solid line, '$*$' marking the values at the lattice points), taken at the vertical centre of Ω, versus corresponding cuts through the exact solution (dotted line) and the result of unmodified viscous transport according to (2.62) ('$\square$' marking the values at the lattice points), right: absolute value of the Euclidean error in χ.

one sixth of the red mass at $\boldsymbol{x}$ is diffusing into the blue phase which perfectly coincides with the one-dimensional case. Note that in the middle columns of Figures 3.11 to 3.13, the smeared indicator function is represented by '$\square$' marking its values at the lattice points. Now, we would like anisotropic diffusion to return the diffused mass to its original phase. The value of χ after anisotropic diffusion was applied is marked with '$*$' in the middle columns of Figures 3.11 to 3.13, so concerning the plots we would like no '$*$' to be located on any other level than zero or one after anisotropic diffusion was applied.

We clearly see from the Figures that one pseudo-time step of anisotropic diffusion does not yield any appreciable sharpening of the interface. After 100 steps, we find that the diffused lattice points have in fact started to move back to their respective phases but after 2000 steps, we realize that those lattice points are still far away from their destination while the indicator function already starts converging to a constant value. The latter is a typical behaviour if too many pseudo-time steps of anisotropic diffusion are applied, see again Weickert's book [**79**] for more details.

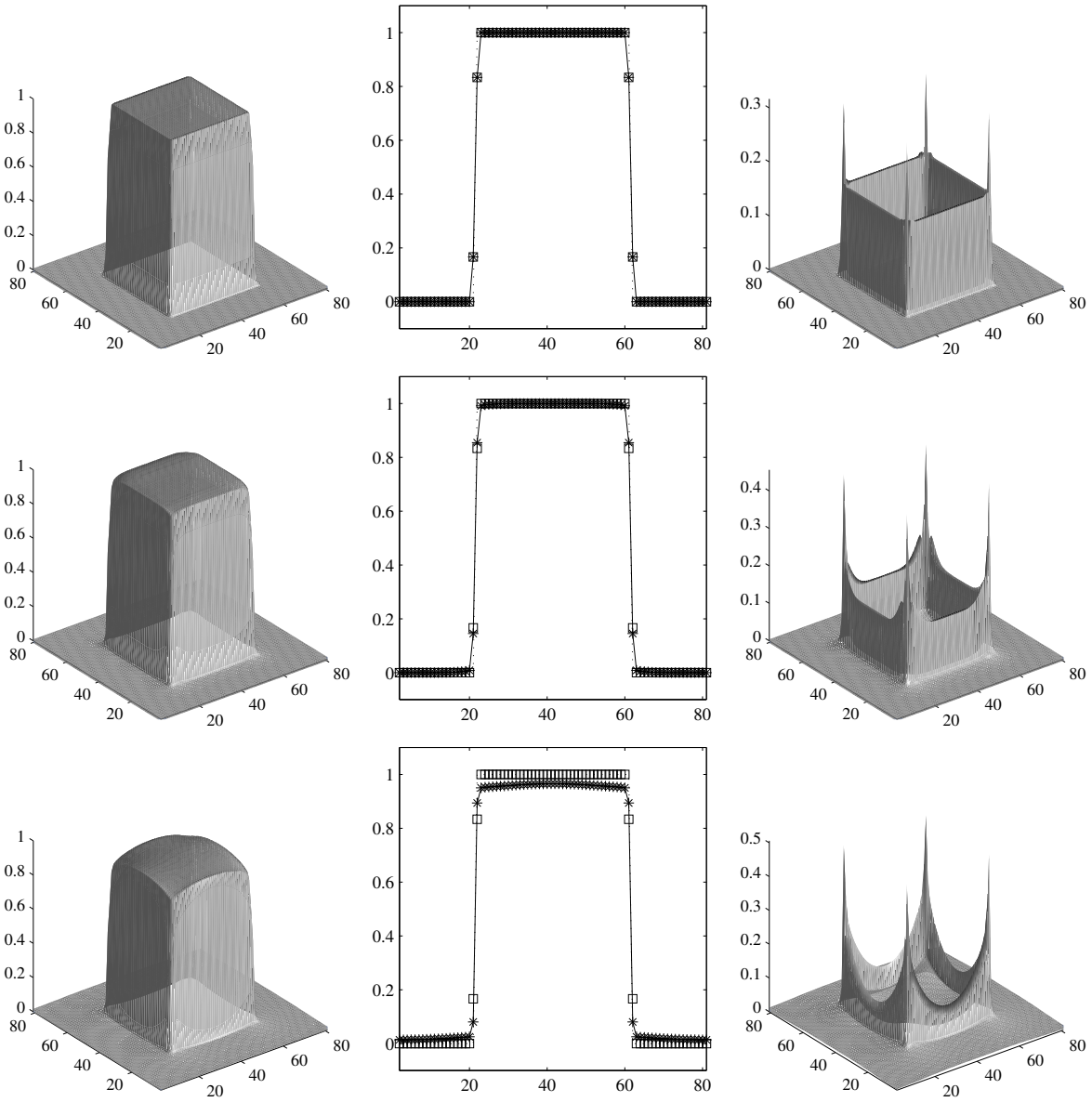


Figure 3.12: Solution of the benchmark problem with Algorithm 3.2 on a 80×80 lattice for $u_\alpha = 0$ and one (top line), 100 (centre line), respectively 2000 (bottom line) iterations of Algorithm 3.1. Left: surface plot of the indicator function χ, middle: horizontal cut through χ (solid line, '$*$' marking the values at the lattice points), taken at the vertical centre of Ω, versus corresponding cuts through the exact solution (dotted line) and the result of unmodified viscous transport according to (2.62) ('$\square$' marking the values at the lattice points), right: absolute value of the Euclidean error in χ.

From the above results, we conclude that anisotropic diffusion is not capable of considerably reducing the strong smearing of the interface produced by the viscous transport scheme (2.62) without introducing other side-effects of similar relevance. Furthermore, we have to remark that the original ILBGK interface tracking Algorithm 2.9 is troublesome only in the case of non-zero velocity but has no real problems with simulating the static case considered here. For those reasons, we decide to give up the idea of replacing the recolouring step with an anisotropic diffusion approach to interface sharpening. However, it should be mentioned that Algorithm 3.2 is perfectly mass conserving for the experiments discussed in this Section, as can be seen from Table 3.2.

3.3.4. Anisotropic diffusion after recolouring. The idea behind this approach is to apply the recolouring step to compensate the strong smearing of the interface and then use anisotropic diffusion to smooth the resulting surface waves, as summarised in Algorithm 3.3.

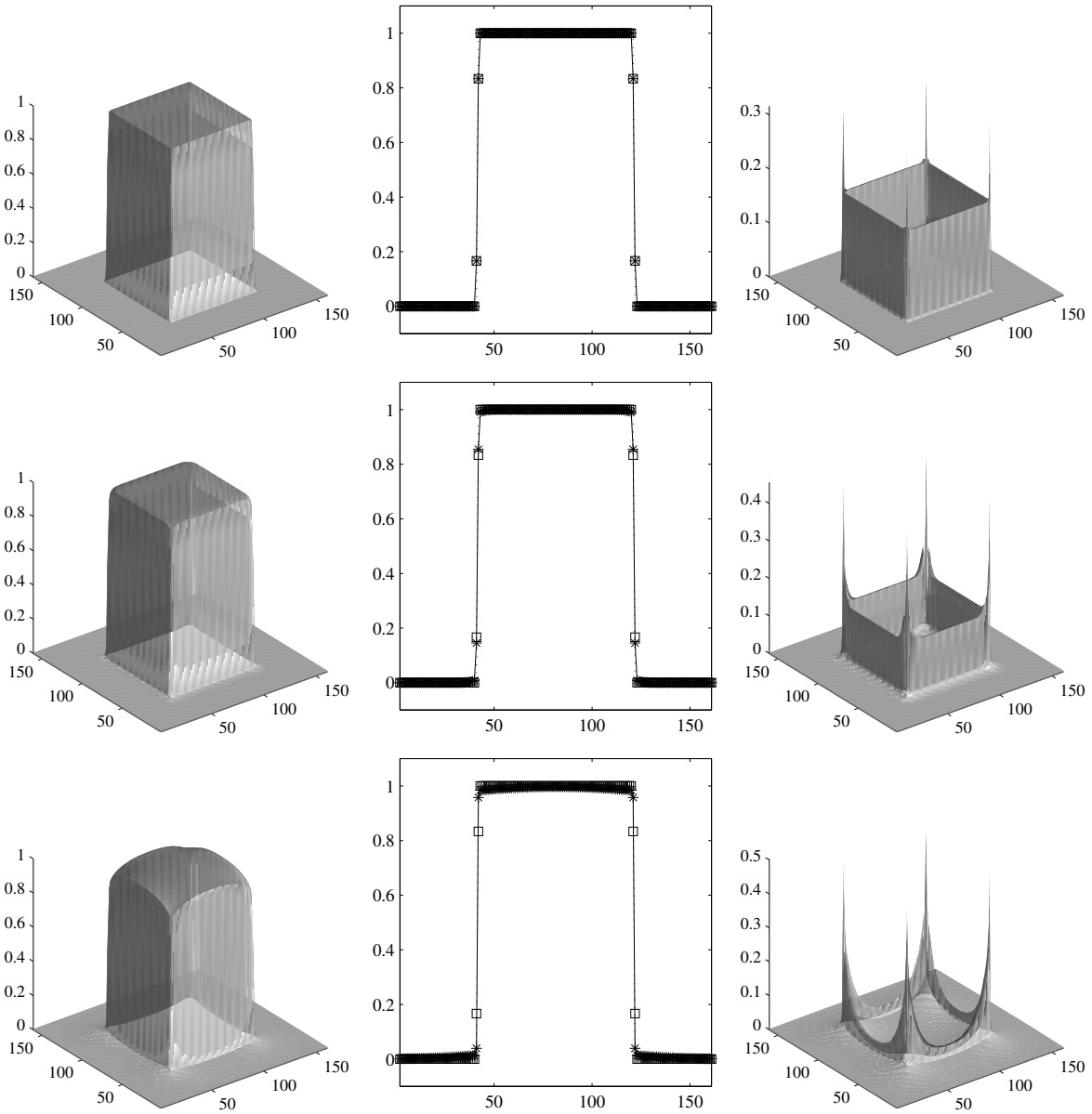


Figure 3.13: Solution of the benchmark problem with Algorithm 3.2 on a 160×160 lattice for $u_\alpha = 0$ and one (top line), 100 (centre line), respectively 2000 (bottom line) iterations of Algorithm 3.1. Left: surface plot of the indicator function χ, middle: horizontal cut through χ (solid line, '$*$' marking the values at the lattice points), taken at the vertical centre of Ω, versus corresponding cuts through the exact solution (dotted line) and the result of unmodified viscous transport according to (2.62) ('$\square$' marking the values at the lattice points), right: absolute value of the Euclidean error in χ.

```
initialise;
while t < t_max
    do recolouring according to Algorithm 2.8;
    determine χ(t + Δt, x) according to (2.66);
    do anisotropic diffusion according to Algorithm 3.1;
    t = t + Δt;
```

Algorithm 3.3: ILBGK interface tracking with anisotropic diffusion after recolouring

Lattice	$\hat{t}_{max}$	Mass loss in %	Number of flops
40×40	1	$-4.26326 \cdot 10^{-14}$	1768277
40×40	100	$-1.42109 \cdot 10^{-14}$	22286423
40×40	2000	$-8.52651 \cdot 10^{-14}$	416069023
80×80	1	0	6840117
80×80	100	$-8.52651 \cdot 10^{-14}$	88425423
80×80	2000	$-4.12115 \cdot 10^{-13}$	$1.6542 \cdot 10^{9}$
160×160	1	0	27022357
160×160	100	$-5.68434 \cdot 10^{-14}$	352393183
160×160	2000	$-3.69482 \cdot 10^{-13}$	$6.5969 \cdot 10^{9}$

Table 3.2: Percentage loss of red mass and computing times for the experiments shown in Figures 3.11 to 3.13

Lattice	Velocity	Mass loss in %	Number of flops
40×40	$\boldsymbol{u}^{(1)}$	-0.0926773	$1.2738 \cdot 10^{11}$
40×40	$\boldsymbol{u}^{(2)}$	-0.0926773	$1.2738 \cdot 10^{11}$
40×40	$\boldsymbol{u}^{(3)}$	-0.0926773	$8.0307 \cdot 10^{11}$
80×80	$\boldsymbol{u}^{(1)}$	-0.0247983	$2.0111 \cdot 10^{12}$
80×80	$\boldsymbol{u}^{(2)}$	-0.0247983	$2.0111 \cdot 10^{12}$
80×80	$\boldsymbol{u}^{(3)}$	-0.0247983	$1.2692 \cdot 10^{13}$
160×160	$\boldsymbol{u}^{(1)}$	-0.00620029	$3.1938 \cdot 10^{13}$
160×160	$\boldsymbol{u}^{(2)}$	-0.00620029	$3.1939 \cdot 10^{13}$
160×160	$\boldsymbol{u}^{(3)}$	-0.00620029	$8.4693 \cdot 10^{13}$

Table 3.3: Percentage loss of red mass and computing times for the experiments with anisotropic diffusion after recolouring

Figures 3.14, 3.15, and 3.16 show benchmark results for $\boldsymbol{u} = \boldsymbol{u}^{(1)}$, $\boldsymbol{u} = \boldsymbol{u}^{(2)}$, and $\boldsymbol{u} = \boldsymbol{u}^{(3)}$, respectively, with $\hat{t}_{max} = 1$ in Algorithm 3.1. All plots show very sharp interfaces without any spurious oscillations, only the corners are smoothed. Unfortunately, looking at Table 3.3 we find that a severe gain of mass occurs in each simulation. This effect depends on lattice size but it seems to be unattached to the velocity field. Anyway, we do not investigate this method any further because mass gain of this magnitude is not acceptable.

3.4. Artificial surface tension

The surface tension term $\sigma\kappa\boldsymbol{n}\delta_\Gamma$ appearing in the incompressible Navier-Stokes equations (2.12) acts only on the interface Γ, its magnitude is proportional to the local curvature κ of Γ, and the direction of its virtue is parallel to the inner unit normal vector $\boldsymbol{n}$ of Γ. Its main effect is the smoothing of strongly curved parts of the boundary. Solving only the transport equation (2.13) we do not have any surface tension, so we do not benefit from such a smoothing effect. However, due to the spurious waves in the interface produced by Algorithm 2.9 it was most welcome. Thus, it is a natural idea to try to smooth those oscillations with some artificial surface tension. Typically, artificial surface tension is used depending on a parameter that measures the disturbance of the interface. In the benchmark, the unperturbed edges of the square are straight lines, i. e. their curvature is zero, so we interpret κ as a governing parameter of artificial surface tension. Note that with this choice, we must accept the corners of the square to be rounded off by surface tension.

In order to apply ILBGK interface tracking with artificial surface tension, we solve the transport equation (2.13) together with the incompressible Navier-Stokes equations (2.31). For this purpose, we proceed as we did for the bubble test in Section 2.4.3, initialising $\boldsymbol{u}(0,\boldsymbol{x}) \in \{\boldsymbol{u}^{(1)}, \boldsymbol{u}^{(2)}, \boldsymbol{u}^{(3)}\}$ instead of $u_\alpha(0,\boldsymbol{x}) = 0$ for each $\boldsymbol{x} \in \mathcal{X}$. Again, we choose the relaxation parameter $\omega = 1$ for all the computations to follow, thus enforcing the dynamic viscosity $\mu = 1/6$. We perform simulations for $\sigma \in \{0.001, 0.01, 0.1\}$ on a 40×40 and an 80×80 lattice. Note that simulations on a 160×160 lattice are no longer performed because of their excessive need of computing time. Benchmark results are shown in Figures 3.17 and 3.18 for

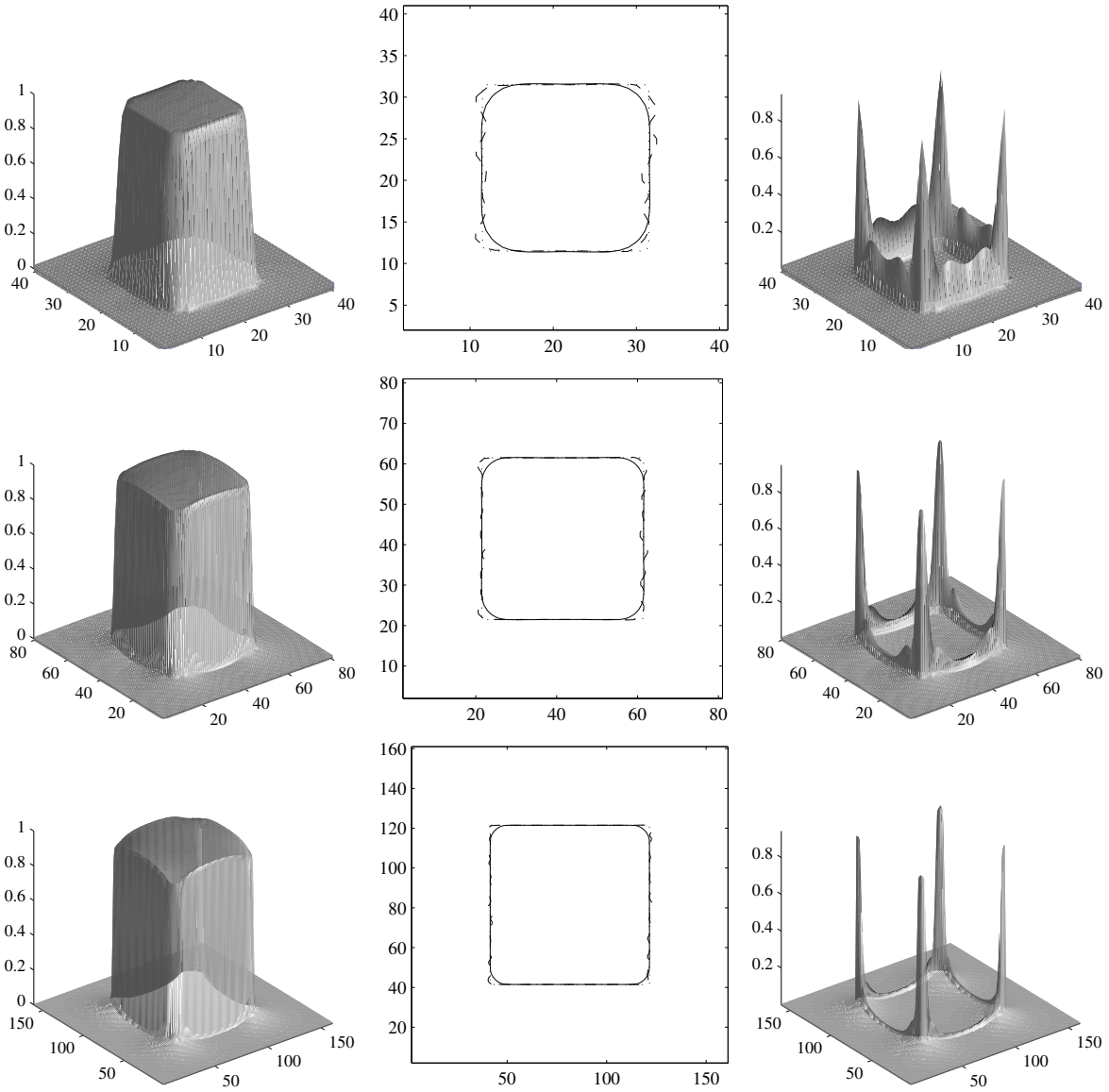


Figure 3.14: Solution of the benchmark problem for $\boldsymbol{u} = \boldsymbol{u}^{(1)}$ with Algorithm 3.3. We present results computed on a 40×40 (top line), 80×80 (centre line), and a 160×160 lattice (bottom line). Left: simulation result, middle: contour line for $\chi = 1/2$ of the analytical solution (dotted line), the plain ILBGK approximation (dashed line) and the simulation result (solid line), right: absolute value of the Euclidean error in χ.

$\boldsymbol{u}(0, \boldsymbol{x}) = \boldsymbol{u}^{(1)}$, in Figures 3.19 and 3.20 for $\boldsymbol{u}(0, \boldsymbol{x}) = \boldsymbol{u}^{(2)}$, and in Figures 3.21 and 3.22 for $\boldsymbol{u}(0, \boldsymbol{x}) = \boldsymbol{u}^{(3)}$. Finally, computing times and percentage loss of red mass are summarised in Table 3.4.

We learn from these experiments that $\sigma = 0.001$ is not enough for smoothing the interface. Also for $\sigma = 0.01$ we still find many spurious waves in Γ, even though the corners of the square are already smoothed a lot. For $\sigma = 0.1$ surface tension already turned the square into a circle. It is especially the contrast between the smoothing of the corners and the non-smoothing of the spurious waves for $\sigma = 0.01$ that teaches us the disability of artificial surface tension to damp the oscillations produced by ILBGK interface tracking.

We also observe severe mass loss in the simulations with $\boldsymbol{u}(0, \boldsymbol{x}) \in \left\{\boldsymbol{u}^{(1)}, \boldsymbol{u}^{(2)}\right\}$ as well as considerably increased (compare Table 3.1) but not really severe mass loss in the simulations with $\boldsymbol{u}(0, \boldsymbol{x}) = \boldsymbol{u}^{(3)}$. In any case, mass loss is increasing with σ. Furthermore, the plots for $\boldsymbol{u}(0, \boldsymbol{x}) = \boldsymbol{u}^{(3)}$ show irregularities in χ which are very similar to those we already observed in Figure 3.7. Due to the latter, it is natural to expect a connection between the difficulties arising here and those discussed at the end of Section 3.2. However, the reason why mass loss and irregularities appear cannot be rigorously explained.

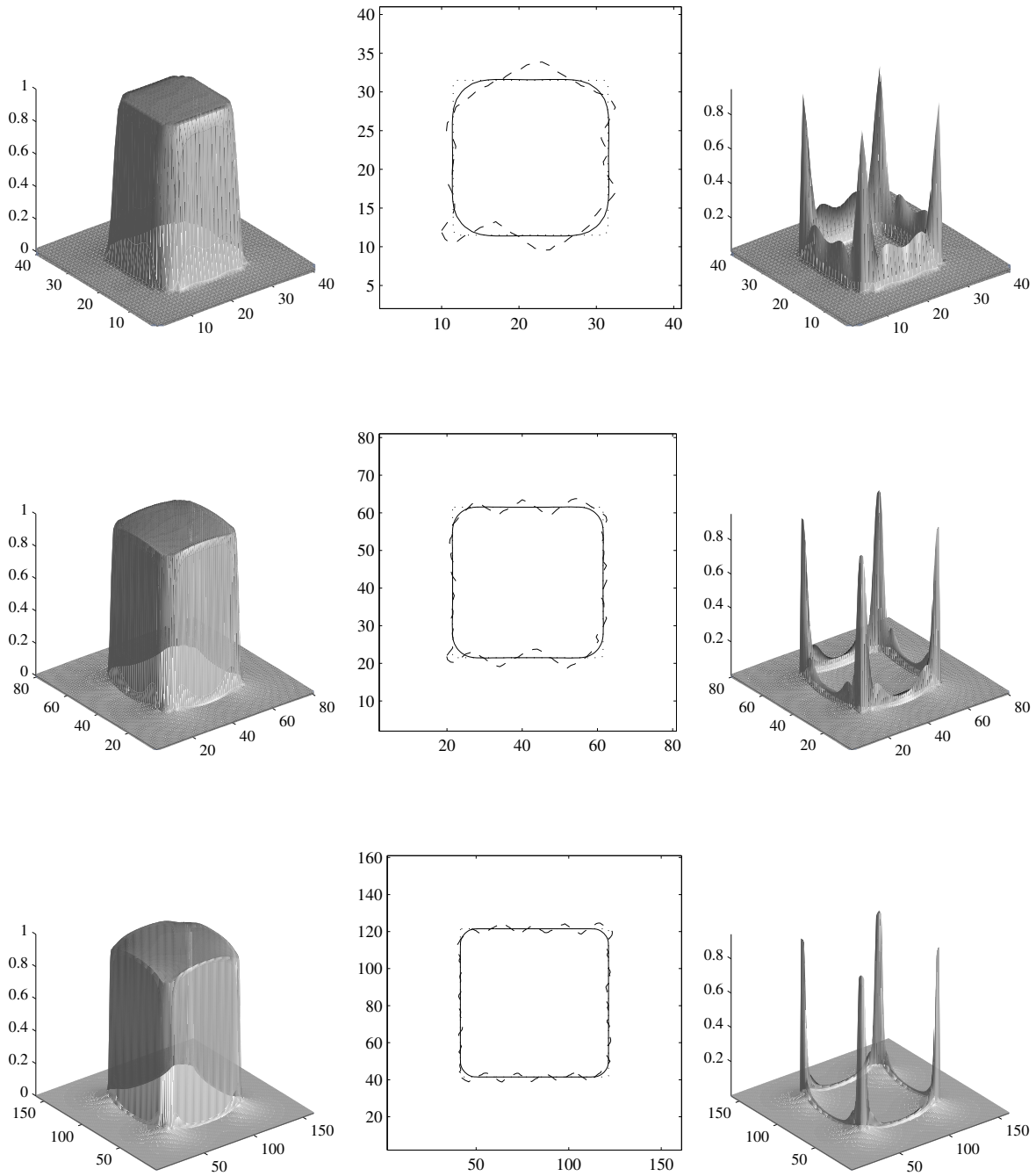


Figure 3.15: Solution of the benchmark problem for $\boldsymbol{u} = \boldsymbol{u}^{(2)}$ with Algorithm 3.3. We present results computed on a 40×40 (top line), 80×80 (centre line), and a 160×160 lattice (bottom line). Left: simulation result, middle: contour line for $\chi = 1/2$ of the analytical solution (dotted line), the plain ILBGK approximation (dashed line) and the simulation result (solid line), right: absolute value of the Euclidean error in χ.

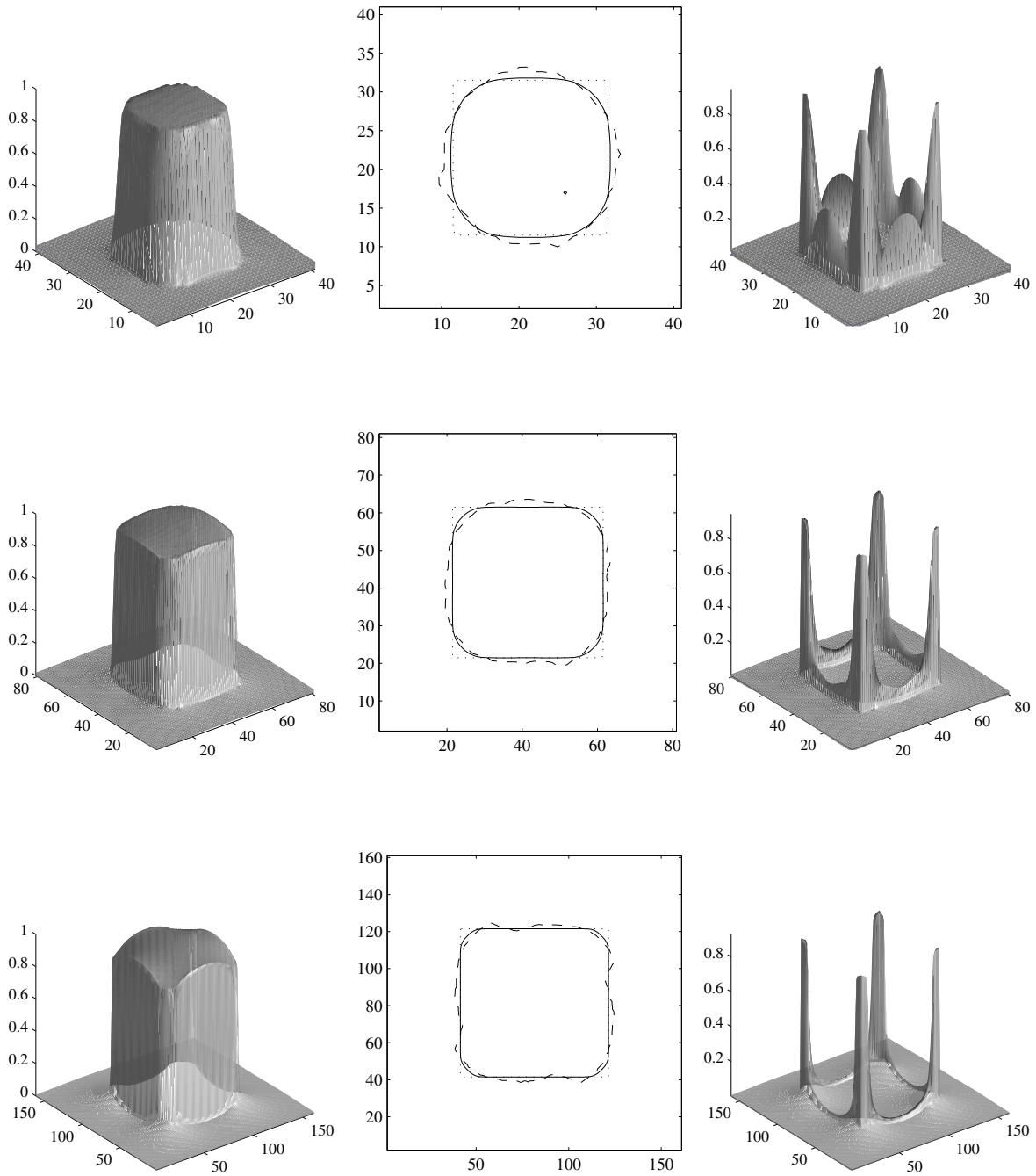


Figure 3.16: Solution of the benchmark problem for $\boldsymbol{u} = \boldsymbol{u}^{(3)}$ with Algorithm 3.3. We present results computed on a 40×40 (top line), 80×80 (centre line), and a 160×160 lattice (bottom line). Left: simulation result, middle: contour line for $\chi = 1/2$ of the analytical solution (dotted line), the plain ILBGK approximation (dashed line) and the simulation result (solid line), right: absolute value of the Euclidean error in χ.

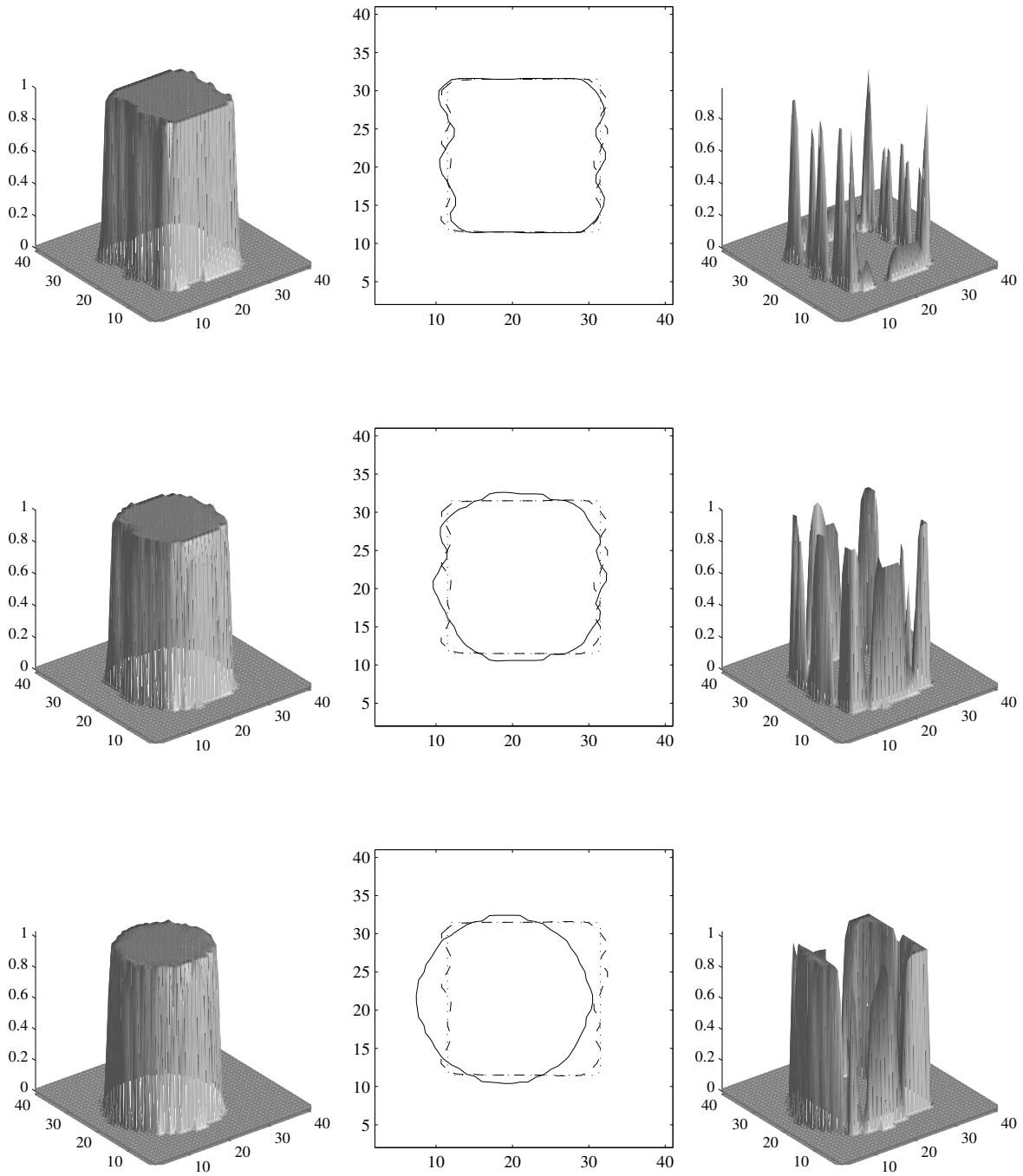


Figure 3.17: ILBGK solution of the benchmark problem with $\sigma = 0.001$ (top), $\sigma = 0.01$ (centre), and $\sigma = 0.1$ (bottom) for $\boldsymbol{u}(0, \boldsymbol{x}) = \boldsymbol{u}^{(1)}$ on a 40×40 lattice. Left: surface plot of χ, middle: level line for $\chi = 1/2$ of the exact solution (dotted line), the ILBGK approximation (dashed line), and the ILBGK result with artificial surface tension (solid line), right: absolute value of the Euclidean error in the ILBGK result with artificial surface tension.

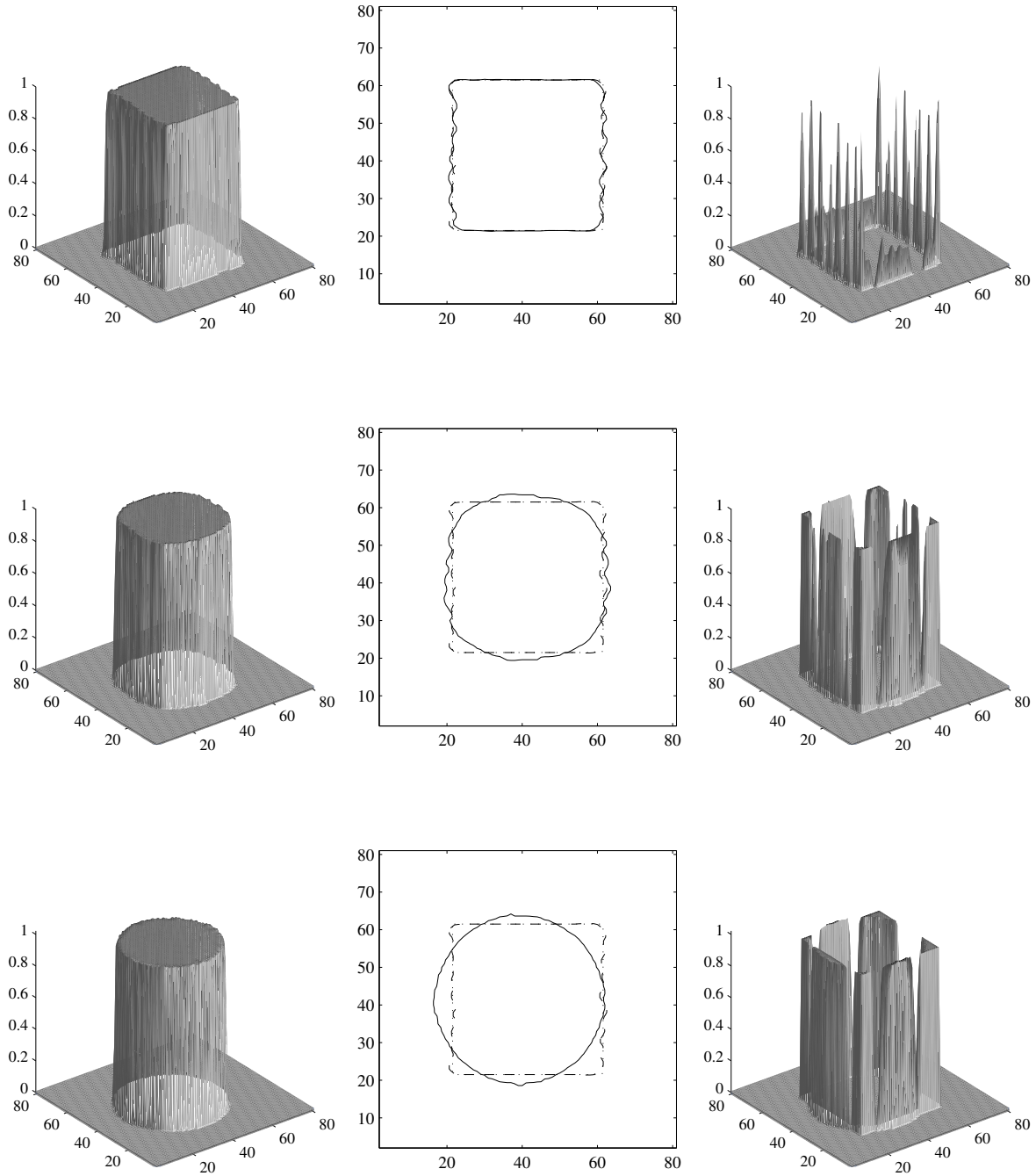


Figure 3.18: ILBGK solution of the benchmark problem with $\sigma = 0.001$ (top), $\sigma = 0.01$ (centre), and $\sigma = 0.1$ (bottom) for $\boldsymbol{u}(0, \boldsymbol{x}) = \boldsymbol{u}^{(1)}$ on a 80×80 lattice. Left: surface plot of χ, middle: level line for $\chi = 1/2$ of the exact solution (dotted line), the ILBGK approximation (dashed line), and the ILBGK result with artificial surface tension (solid line), right: absolute value of the Euclidean error in the ILBGK result with artificial surface tension.

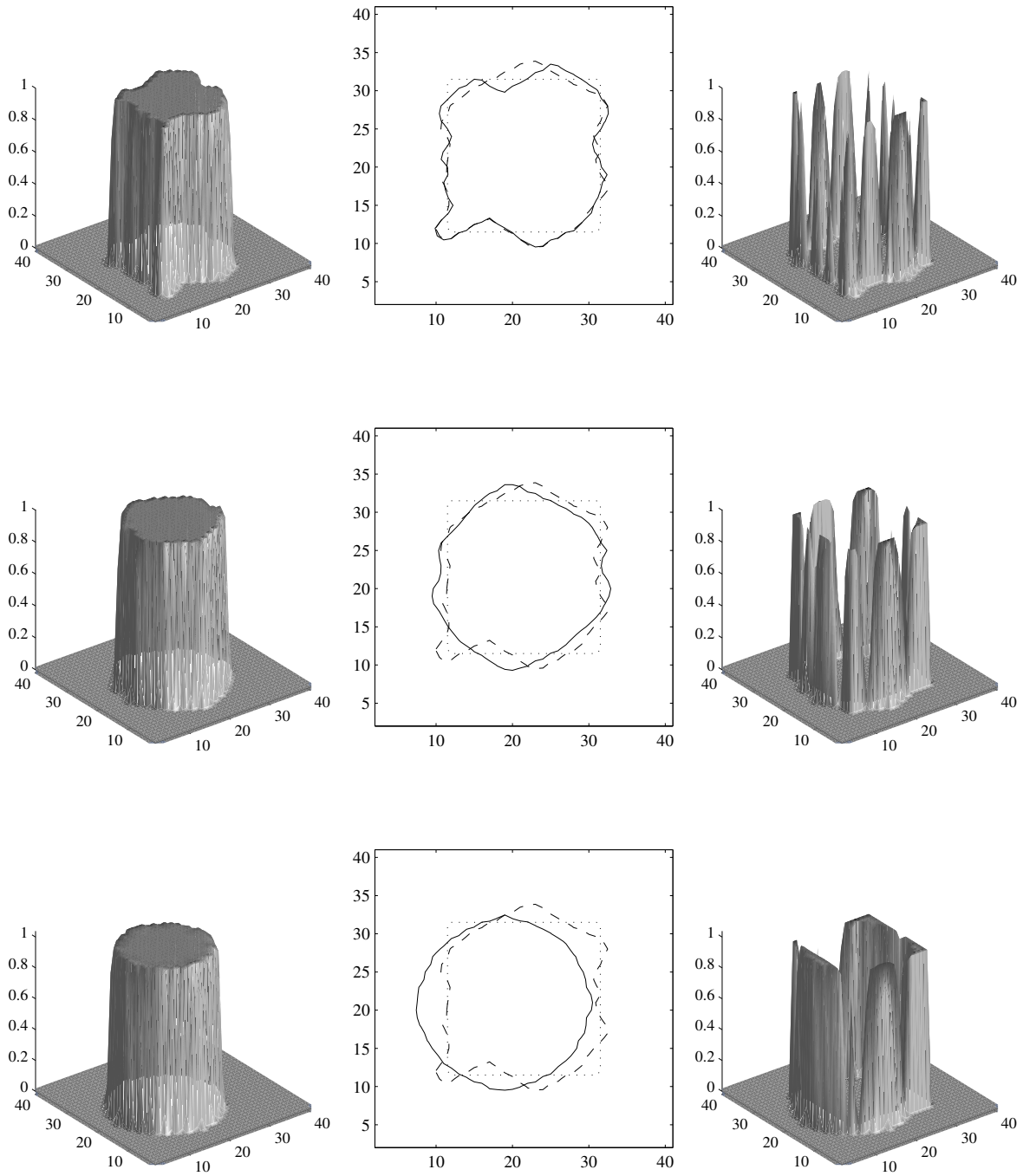


Figure 3.19: ILBGK solution of the benchmark problem with $\sigma = 0.001$ (top), $\sigma = 0.01$ (centre), and $\sigma = 0.1$ (bottom) for $\boldsymbol{u}(0, \boldsymbol{x}) = \boldsymbol{u}^{(2)}$ on a 40×40 lattice. Left: surface plot of χ, middle: level line for $\chi = 1/2$ of the exact solution (dotted line), the ILBGK approximation (dashed line), and the ILBGK result with artificial surface tension (solid line), right: absolute value of the Euclidean error in the ILBGK result with artificial surface tension.

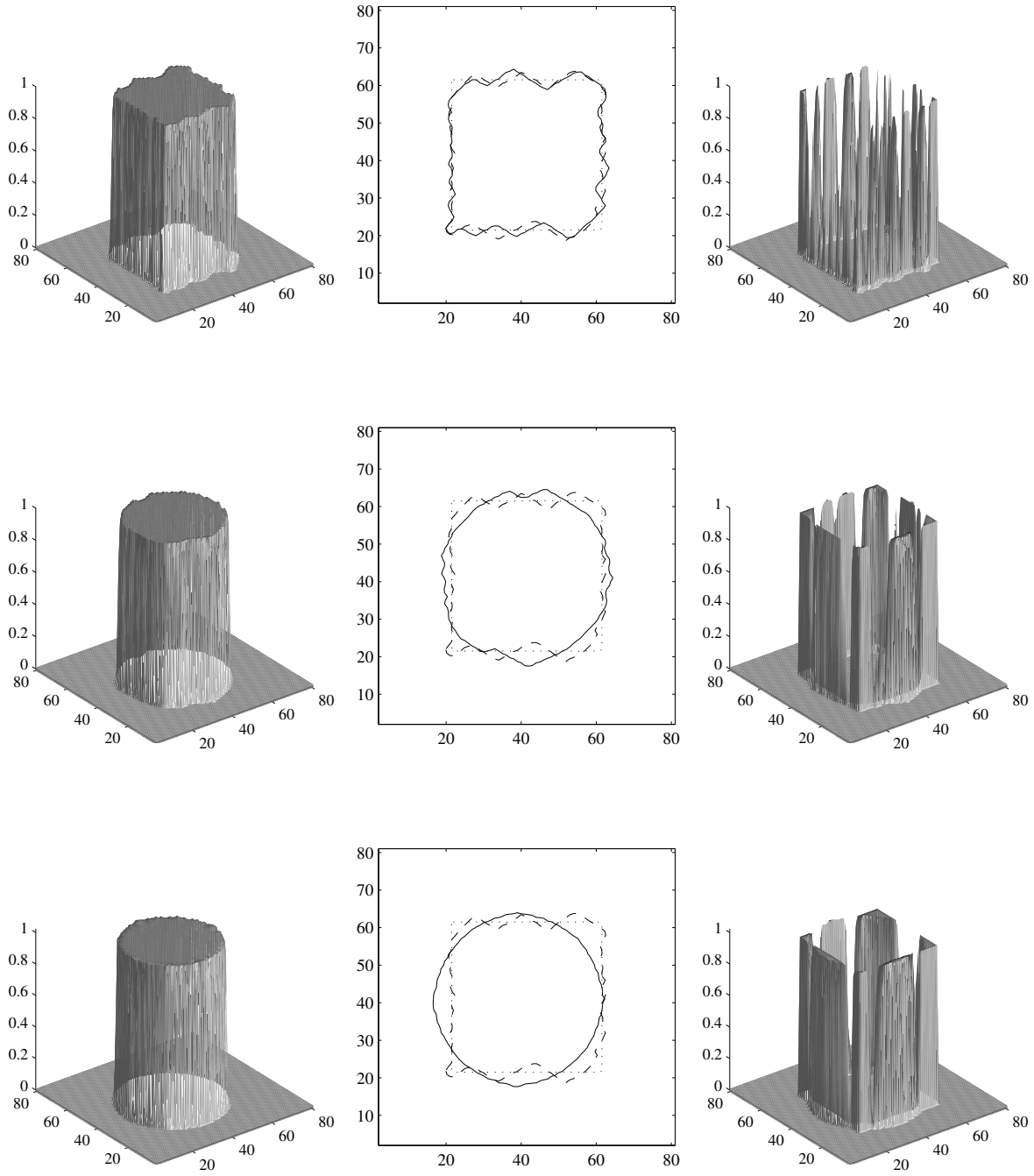


Figure 3.20: ILBGK solution of the benchmark problem with $\sigma = 0.001$ (top), $\sigma = 0.01$ (centre), and $\sigma = 0.1$ (bottom) for $\boldsymbol{u}(0, \boldsymbol{x}) = \boldsymbol{u}^{(2)}$ on a 80×80 lattice. Left: surface plot of χ, middle: level line for $\chi = 1/2$ of the exact solution (dotted line), the ILBGK approximation (dashed line), and the ILBGK result with artificial surface tension (solid line), right: absolute value of the Euclidean error in the ILBGK result with artificial surface tension.

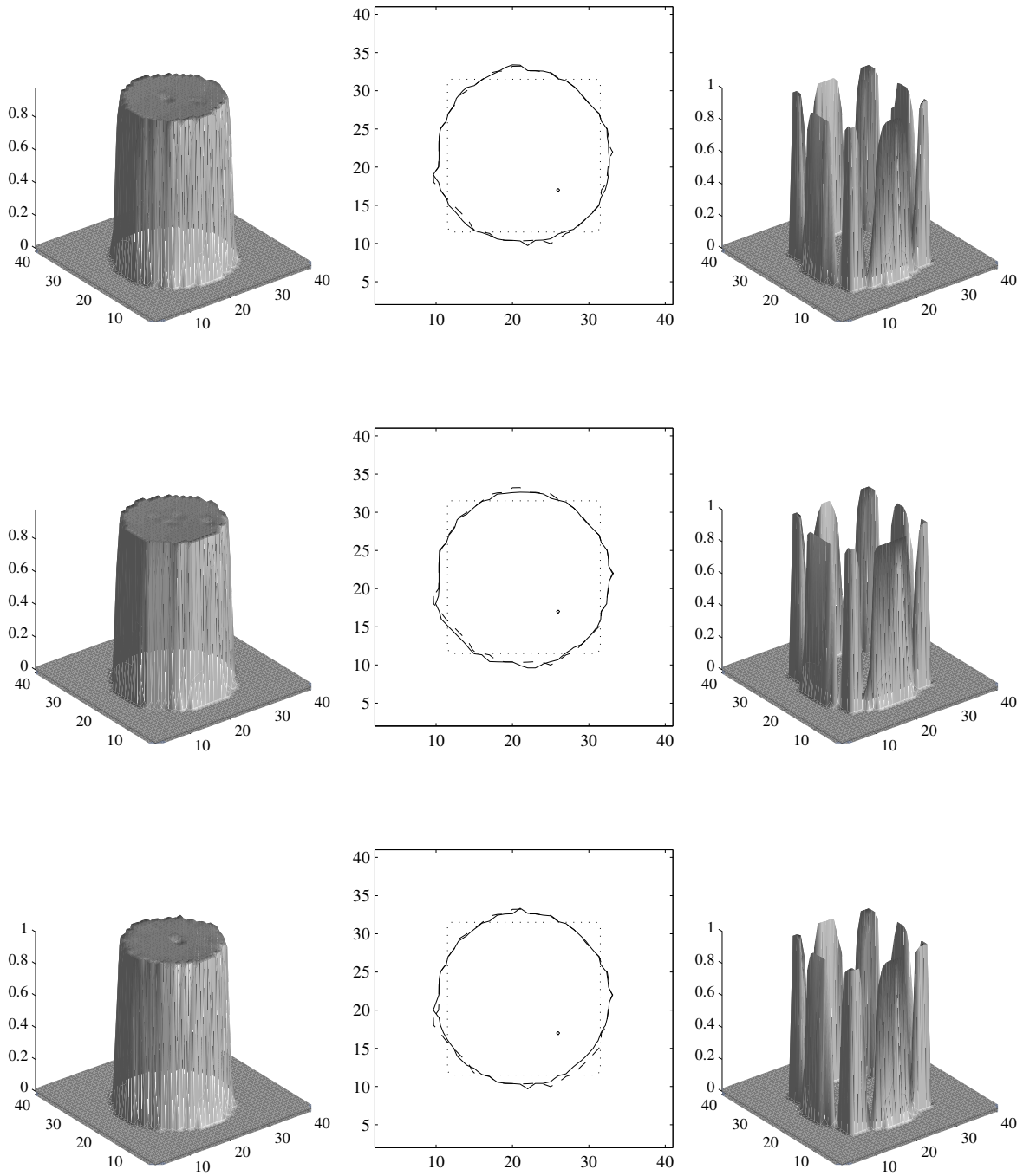


Figure 3.21: ILBGK solution of the benchmark problem with $\sigma = 0.001$ (top), $\sigma = 0.01$ (centre), and $\sigma = 0.1$ (bottom) for $\boldsymbol{u}(0, \boldsymbol{x}) = \boldsymbol{u}^{(3)}$ on a 40×40 lattice. Left: surface plot of χ, middle: level line for $\chi = 1/2$ of the exact solution (dotted line), the ILBGK approximation (dashed line), and the ILBGK result with artificial surface tension (solid line), right: absolute value of the Euclidean error in the ILBGK result with artificial surface tension.

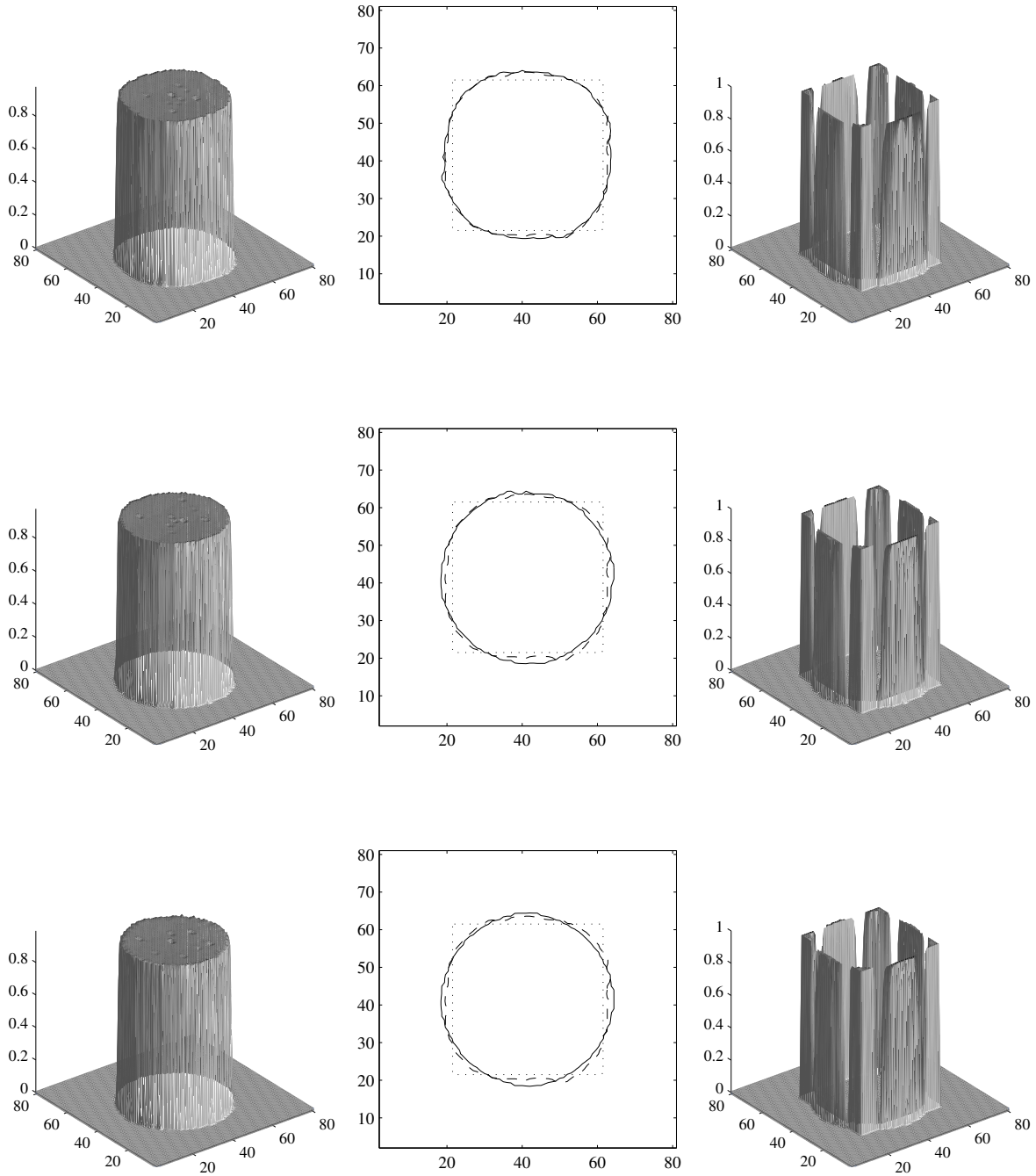


Figure 3.22: ILBGK solution of the benchmark problem with $\sigma = 0.001$ (top), $\sigma = 0.01$ (centre), and $\sigma = 0.1$ (bottom) for $\boldsymbol{u}(0, \boldsymbol{x}) = \boldsymbol{u}^{(3)}$ on a 80×80 lattice. Left: surface plot of χ, middle: level line for $\chi = 1/2$ of the exact solution (dotted line), the ILBGK approximation (dashed line), and the ILBGK result with artificial surface tension (solid line), right: absolute value of the Euclidean error in the ILBGK result with artificial surface tension.

Lattice	$\boldsymbol{u}(0,\boldsymbol{x})$	σ	Mass loss in %	Number of flops
40×40	$\boldsymbol{u}^{(1)}$	0.001	0.00266895	$1.77729 \cdot 10^9$
40×40	$\boldsymbol{u}^{(1)}$	0.01	0.0320708	$6.44152 \cdot 10^9$
40×40	$\boldsymbol{u}^{(1)}$	0.1	0.282062	$6.72455 \cdot 10^9$
40×40	$\boldsymbol{u}^{(2)}$	0.001	0.00479016	$5.33269 \cdot 10^8$
40×40	$\boldsymbol{u}^{(2)}$	0.01	0.0343632	$5.33257 \cdot 10^8$
40×40	$\boldsymbol{u}^{(2)}$	0.1	0.346245	$5.33157 \cdot 10^8$
40×40	$\boldsymbol{u}^{(3)}$	0.001	$7.57425 \cdot 10^{-8}$	$4.44354 \cdot 10^9$
40×40	$\boldsymbol{u}^{(3)}$	0.01	$7.73500 \cdot 10^{-8}$	$4.44341 \cdot 10^9$
40×40	$\boldsymbol{u}^{(3)}$	0.1	$7.4175 \cdot 10^{-8}$	$4.44325 \cdot 10^9$
80×80	$\boldsymbol{u}^{(1)}$	0.001	0.00143724	$6.73781 \cdot 10^{10}$
80×80	$\boldsymbol{u}^{(1)}$	0.01	0.0188019	$6.13699 \cdot 10^{10}$
80×80	$\boldsymbol{u}^{(1)}$	0.1	0.155951	$6.3404 \cdot 10^{10}$
80×80	$\boldsymbol{u}^{(2)}$	0.001	0.00232742	$4.08821 \cdot 10^9$
80×80	$\boldsymbol{u}^{(2)}$	0.01	0.0221576	$4.08812 \cdot 10^9$
80×80	$\boldsymbol{u}^{(2)}$	0.1	0.146429	$4.08773 \cdot 10^9$
80×80	$\boldsymbol{u}^{(3)}$	0.001	$1.34075 \cdot 10^{-7}$	$1.00634 \cdot 10^{11}$
80×80	$\boldsymbol{u}^{(3)}$	0.01	$1.31308 \cdot 10^{-7}$	$1.00634 \cdot 10^{11}$
80×80	$\boldsymbol{u}^{(3)}$	0.1	$1.34066 \cdot 10^{-7}$	$1.00633 \cdot 10^{11}$

Table 3.4: Percentage loss of red mass and computing times for ILBGK interface tracking with artificial surface tension

CHAPTER 4

Numerical analysis of the viscous transport scheme

Due to the unsatisfactory behaviour of ILBGK interface tracking (compare Chapter 3) we now try to modify this method such that it produces more reasonable results. In particular, we generalise the ILBGK transport scheme (2.62) and seek a choice of parameters that produces a stable scheme for interface tracking which is less diffusive than (2.62). The idea behind this approach is that a scheme with only little diffusion requires only little anti-diffusion, i. e. recolouring, to produce a sharp interface and that little recolouring should produce only little disturbance of the interface. In the ideal case, we even ended up with a scheme that does not need any recolouring at all.

We will construct a general transport scheme in Section 4.1 and use its stability properties to find the most reasonable choice of parameters in Section 4.2. Finally, in Section 4.3, we will apply the resulting transport scheme with and without recolouring to the benchmark problem.

4.1. Generalisation of the viscous transport scheme

The ILBGK transport scheme (2.61) yields a solution of the viscous transport equation (2.63). Here, we want to investigate whether it is possible to reduce the undue diffusion in (2.63) by modifying the viscous transport scheme (2.61). Therefore, we consider the general lattice BGK transport equation

$$R(t+\Delta t,\boldsymbol{x};\boldsymbol{v}) = R^{eq}_{gen}\big(\chi(t,\boldsymbol{x}-\boldsymbol{v}\Delta x),\boldsymbol{u}(t,\boldsymbol{x}-\boldsymbol{v}\Delta x);\boldsymbol{v}\big) \tag{4.1}$$

for the pseudo-particle density R with $\langle R,1\rangle_{\boldsymbol{v}} = \chi$. The general equilibrium density R^{eq}_{gen} of pseudo-particles is given by

$$R^{eq}_{gen}\left(\chi,\boldsymbol{u};\boldsymbol{v}\right) = f^*_{gen}(\boldsymbol{v})\left(\chi + a\Delta x\chi u_\alpha v_\alpha\right), \tag{4.2}$$

where $0 < a \in \mathbb{R}$ and f^*_{gen} is a weight function (see Junk [**41**] for guidance in constructing equilibrium (pseudo-)particle densities). Furthermore, we remove the constraint $\Delta t = \Delta x^2$ to gain additional flexibility.

Taking the zeroth order equivalent moment of both sides of equation (4.1) we obtain

$$\chi(t+\Delta t,\boldsymbol{x}) = \left\langle f^*_{gen}, \chi(t,\boldsymbol{x}-\boldsymbol{v}\Delta x) + a\Delta x v_\alpha \chi(t,\boldsymbol{x}-\boldsymbol{v}\Delta x)u_\alpha(t,\boldsymbol{x}-\boldsymbol{v}\Delta x)\right\rangle_{\boldsymbol{v}}, \tag{4.3}$$

compare Section 1.1.2. Taylor expansion yields

$$\chi(t,\boldsymbol{x}-\boldsymbol{v}\Delta x) = \chi(t,\boldsymbol{x}) - \Delta x v_\alpha \partial_\alpha \chi(t,\boldsymbol{x}) + \frac{\Delta x^2}{2} v_\alpha v_\beta \partial_{\alpha\beta}\chi(t,\boldsymbol{x}) - \frac{\Delta x^3}{6} v_\alpha v_\beta v_\gamma \partial_{\alpha\beta\gamma}\chi(t,\boldsymbol{x}) + \mathcal{O}\left(\Delta x^4\right)$$

as well as

$$\boldsymbol{u}_\alpha(t,\boldsymbol{x}-\boldsymbol{v}\Delta x) = \boldsymbol{u}_\alpha(t,\boldsymbol{x}) - \Delta x v_\beta \partial_\beta \boldsymbol{u}_\alpha(t,\boldsymbol{x}) + \frac{\Delta x^2}{2} v_\beta v_\gamma \partial_{\beta\gamma}\boldsymbol{u}_\alpha(t,\boldsymbol{x}) - \frac{\Delta x^3}{6} v_\beta v_\gamma v_\varepsilon \partial_{\beta\gamma\varepsilon} u_\alpha(t,\boldsymbol{x}) + \mathcal{O}\left(\Delta x^4\right)$$

and inserting those results into (4.3) we find that assuming

$$\left\langle f^*_{gen},1\right\rangle_{\boldsymbol{v}} = 1, \qquad \left\langle f^*_{gen}, v_\alpha\right\rangle_{\boldsymbol{v}} = 0, \qquad \left\langle f^*_{gen}, v_\alpha v_\beta\right\rangle_{\boldsymbol{v}} = \frac{\Delta t}{a\Delta x^2}\delta_{\alpha\beta}, \qquad \text{and} \qquad \left\langle v_\alpha v_\beta v_\gamma, f^*_{gen}\right\rangle_{\boldsymbol{v}} = 0 \tag{4.4}$$

equation (4.3) is equivalent to

$$\chi(t+\Delta t,\boldsymbol{x}) - \chi(t,\boldsymbol{x}) + \Delta t u_\alpha(t,\boldsymbol{x})\partial_\alpha\chi(t,\boldsymbol{x}) + \Delta t\partial_\alpha u_\alpha(t,\boldsymbol{x})\chi(t,\boldsymbol{x}) = \frac{\Delta t}{2a}\partial_{\alpha\alpha}\chi(t,\boldsymbol{x}) + \mathcal{O}\left(\Delta x^2\right)$$

which can be interpreted as explicit Euler time discretisation of

$$\partial_t\chi + \partial_\alpha(u_\alpha\chi) = \frac{1}{2a}\partial_{\alpha\alpha}\chi + \mathcal{O}\left(\Delta t,\Delta x^2\right). \tag{4.5}$$

Note that for $a = 3$ and $\Delta t = \Delta x^2$, this coincides exactly with the viscous transport equation (2.63).

Let us now derive the most general weight function f^*_{gen} fulfilling (4.4). Rewriting (4.4) we obtain the linear system of equations

$$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ c_{0;1} & c_{1;1} & c_{2;1} & c_{3;1} & c_{4;1} & c_{5;1} & c_{6;1} & c_{7;1} & c_{8;1} \\ c_{0;2} & c_{1;2} & c_{2;2} & c_{3;2} & c_{4;2} & c_{5;2} & c_{6;2} & c_{7;2} & c_{8;2} \\ c_{0;1}^2 & c_{1;1}^2 & c_{2;1}^2 & c_{3;1}^2 & c_{4;1}^2 & c_{5;1}^2 & c_{6;1}^2 & c_{7;1}^2 & c_{8;1}^2 \\ c_{0;1}c_{0;2} & c_{1;1}c_{1;2} & c_{2;1}c_{2;2} & c_{3;1}c_{3;2} & c_{4;1}c_{4;2} & c_{5;1}c_{5;2} & c_{6;1}c_{6;2} & c_{7;1}c_{7;2} & c_{8;1}c_{8;2} \\ c_{0;2}^2 & c_{1;2}^2 & c_{2;2}^2 & c_{3;2}^2 & c_{4;2}^2 & c_{5;2}^2 & c_{6;2}^2 & c_{7;2}^2 & c_{8;2}^2 \\ c_{0;1}^3 & c_{1;1}^3 & c_{2;1}^3 & c_{3;1}^3 & c_{4;1}^3 & c_{5;1}^3 & c_{6;1}^3 & c_{7;1}^3 & c_{8;1}^3 \\ c_{0;1}^2c_{0;2} & c_{1;1}^2c_{1;2} & c_{2;1}^2c_{2;2} & c_{3;1}^2c_{3;2} & c_{4;1}^2c_{4;2} & c_{5;1}^2c_{5;2} & c_{6;1}^2c_{6;2} & c_{7;1}^2c_{7;2} & c_{8;1}^2c_{8;2} \\ c_{0;1}c_{0;2}^2 & c_{1;1}c_{1;2}^2 & c_{2;1}c_{2;2}^2 & c_{3;1}c_{3;2}^2 & c_{4;1}c_{4;2}^2 & c_{5;1}c_{5;2}^2 & c_{6;1}c_{6;2}^2 & c_{7;1}c_{7;2}^2 & c_{8;1}c_{8;2}^2 \\ c_{0;2}^3 & c_{1;2}^3 & c_{2;2}^3 & c_{3;2}^3 & c_{4;2}^3 & c_{5;2}^3 & c_{6;2}^3 & c_{7;2}^3 & c_{8;2}^3 \end{pmatrix} \begin{pmatrix} f^*_{gen}(\boldsymbol{c}_0) \\ f^*_{gen}(\boldsymbol{c}_1) \\ f^*_{gen}(\boldsymbol{c}_2) \\ f^*_{gen}(\boldsymbol{c}_3) \\ f^*_{gen}(\boldsymbol{c}_4) \\ f^*_{gen}(\boldsymbol{c}_5) \\ f^*_{gen}(\boldsymbol{c}_6) \\ f^*_{gen}(\boldsymbol{c}_7) \\ f^*_{gen}(\boldsymbol{c}_8) \end{pmatrix}$$
$$= \left(1, 0, 0, \Delta t/\left(a\Delta x^2\right), 0, \Delta t/\left(a\Delta x^2\right), 0, 0, 0, 0\right)^T.$$

This is a system of ten equations for nine unknowns. However, explicitly calculating the entries of the system matrix we find

$$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & -1 & -1 & -1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 & -1 & -1 & -1 \\ 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & -1 & 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & -1 & -1 & -1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & -1 & 0 & -1 \\ 0 & 0 & 1 & 0 & -1 & 0 & -1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 & -1 & -1 & -1 \end{pmatrix} \begin{pmatrix} f^*_{gen}(\boldsymbol{c}_0) \\ f^*_{gen}(\boldsymbol{c}_1) \\ f^*_{gen}(\boldsymbol{c}_2) \\ f^*_{gen}(\boldsymbol{c}_3) \\ f^*_{gen}(\boldsymbol{c}_4) \\ f^*_{gen}(\boldsymbol{c}_5) \\ f^*_{gen}(\boldsymbol{c}_6) \\ f^*_{gen}(\boldsymbol{c}_7) \\ f^*_{gen}(\boldsymbol{c}_8) \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ \Delta t/\left(a\Delta x^2\right) \\ 0 \\ \Delta t/\left(a\Delta x^2\right) \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

and at this stage, we observe the second equation to equal the seventh and the third equation to equal the tenth, so actually, the system contains only eight linear independent equations for nine unknowns and it yields the one-parameter solution

$$f^*_{gen}(\boldsymbol{c}_j) = \begin{cases} 1 + 4\frac{\Delta t}{2a\Delta x^2}(\mathscr{a} - 1) & \text{for } j = 0 \\ \frac{\Delta t}{2a\Delta x^2}(1 - 2\mathscr{a}) & \text{for } j = 1, 3, 5, 7 \\ \frac{\Delta t}{2a\Delta x^2}\mathscr{a} & \text{for } j = 2, 4, 6, 8 \end{cases}$$

with parameter $\mathscr{a}$. Note that for $\mathscr{a} = 1/6$, $a = 3$, and $\Delta t = \Delta x^2$ the weight function f^*_{gen} coincides exactly with f^*. The moment equation (4.3) can now be rewritten in the form

$$\chi(t + \Delta t, \boldsymbol{x}) = \chi(t, \boldsymbol{x}) + \frac{\Delta t}{2a\Delta x^2} \begin{bmatrix} \mathscr{a} & 1 - 2\mathscr{a} & \mathscr{a} \\ 1 - 2\mathscr{a} & 4(\mathscr{a} - 1) & 1 - 2\mathscr{a} \\ \mathscr{a} & 1 - 2\mathscr{a} & \mathscr{a} \end{bmatrix} \chi(t, \boldsymbol{x})$$
$$- \frac{\Delta t}{2\Delta x} \begin{bmatrix} -\mathscr{a} & 0 & \mathscr{a} \\ 2\mathscr{a} - 1 & 0 & 1 - 2\mathscr{a} \\ -\mathscr{a} & 0 & \mathscr{a} \end{bmatrix} (\chi u_1)(t, \boldsymbol{x}) - \frac{\Delta t}{2\Delta x} \begin{bmatrix} \mathscr{a} & 1 - 2\mathscr{a} & \mathscr{a} \\ 0 & 0 & 0 \\ -\mathscr{a} & 2\mathscr{a} - 1 & -\mathscr{a} \end{bmatrix} (\chi u_2)(t, \boldsymbol{x}), \quad (4.6)$$

so we just proved

Proposition 4.1. *The finite difference scheme* (4.6) *which is equivalent to equation* (4.1) *is consistent of* $\mathcal{O}\left(\Delta t, \Delta x^2\right)$ *to the viscous transport equation* (4.5). *The classical ILBGK interface tracking scheme* (2.61) *coincides with the choice* $\mathscr{a} = 1/6$, $a = 3$, *and* $\Delta t = \Delta x^2$.

Checking the consistency of the stencils in (4.6) we find by straight Taylor expansion that

$$\frac{1}{\Delta x^2}\begin{bmatrix} \mathscr{a} & 1-2\mathscr{a} & \mathscr{a} \\ 1-2\mathscr{a} & 4(\mathscr{a}-1) & 1-2\mathscr{a} \\ \mathscr{a} & 1-2\mathscr{a} & \mathscr{a} \end{bmatrix} \leftrightarrow \partial_{\alpha\alpha} + \mathcal{O}\left(\Delta x^2\right), \tag{4.7}$$

$$\frac{1}{2\Delta x}\begin{bmatrix} -\mathscr{a} & 0 & \mathscr{a} \\ 2\mathscr{a}-1 & 0 & 1-2\mathscr{a} \\ -\mathscr{a} & 0 & \mathscr{a} \end{bmatrix} \leftrightarrow \partial_1 + \mathcal{O}\left(\Delta x^2\right), \qquad \text{and} \qquad \frac{1}{2\Delta x}\begin{bmatrix} \mathscr{a} & 1-2\mathscr{a} & \mathscr{a} \\ 0 & 0 & 0 \\ -\mathscr{a} & 2\mathscr{a}-1 & -\mathscr{a} \end{bmatrix} \leftrightarrow \partial_2 + \mathcal{O}\left(\Delta x^2\right).$$

Let us think a little bit more about the Laplace stencil (4.7). By Taylor expansion, we find that for a general nine-point stencil and a smooth function $g : [0, t_{max}] \times \Omega \to \mathbb{R}$,

$$\begin{aligned}\begin{bmatrix} \mathscr{b}_4 & \mathscr{b}_3 & \mathscr{b}_2 \\ \mathscr{b}_5 & \mathscr{b}_0 & \mathscr{b}_1 \\ \mathscr{b}_6 & \mathscr{b}_7 & \mathscr{b}_8 \end{bmatrix} g(t,\boldsymbol{x}) &= g(t,\boldsymbol{x})\sum_{j=0}^{8}\mathscr{b}_j + \Delta x\partial_\alpha g(t,\boldsymbol{x})\sum_{j=1}^{8}\mathscr{b}_j c_{j;\alpha} \\ &\quad + \frac{\Delta x^2}{2}\partial_{\alpha\beta} g(t,\boldsymbol{x})\sum_{j=1}^{8}\mathscr{b}_j c_{j;\alpha}c_{j;\beta} + \frac{\Delta x^3}{6}\partial_{\alpha\beta\gamma} g(t,\boldsymbol{x})\sum_{j=1}^{8}\mathscr{b}_j c_{j;\alpha}c_{j;\beta}c_{j;\gamma} + \mathcal{O}\left(\Delta x^4\right)\end{aligned} \tag{4.8}$$

and therefore,

$$\begin{bmatrix} \mathscr{b}_4 & \mathscr{b}_3 & \mathscr{b}_2 \\ \mathscr{b}_5 & \mathscr{b}_0 & \mathscr{b}_1 \\ \mathscr{b}_6 & \mathscr{b}_7 & \mathscr{b}_8 \end{bmatrix} \leftrightarrow \partial_{\alpha\alpha} + \mathcal{O}\left(\Delta x^2\right) \tag{4.9}$$

if and only if

$$\sum_{j=0}^{8}\mathscr{b}_j = 0, \qquad \sum_{j=1}^{8}\mathscr{b}_j c_{j;\alpha} = 0, \qquad \sum_{j=1}^{8}\mathscr{b}_j c_{j;\alpha}c_{j;\beta} = \frac{2}{\Delta x^2}\delta_{\alpha\beta}, \qquad \text{and} \qquad \sum_{j=1}^{8}\mathscr{b}_j c_{j;\alpha}c_{j;\beta}c_{j;\gamma} = 0.$$

In complete analogy to the above, those conditions can be rewritten in the form

$$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & -1 & -1 & -1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 & -1 & -1 & -1 \\ 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & -1 & 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & -1 & -1 & -1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & -1 & 0 & -1 \\ 0 & 0 & 1 & 0 & -1 & 0 & -1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 & -1 & -1 & -1 \end{pmatrix} \begin{pmatrix} \mathscr{b}_0 \\ \mathscr{b}_1 \\ \mathscr{b}_2 \\ \mathscr{b}_3 \\ \mathscr{b}_4 \\ \mathscr{b}_5 \\ \mathscr{b}_6 \\ \mathscr{b}_7 \\ \mathscr{b}_8 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 2/\Delta x^2 \\ 0 \\ 2/\Delta x^2 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}.$$

This system of equations reduces to

$$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & -1 & -1 & -1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 & -1 & -1 & -1 \\ 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & -1 & 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & -1 & 0 & -1 \\ 0 & 0 & 1 & 0 & -1 & 0 & -1 & 0 & 1 \end{pmatrix} \begin{pmatrix} \mathscr{b}_0 \\ \mathscr{b}_1 \\ \mathscr{b}_2 \\ \mathscr{b}_3 \\ \mathscr{b}_4 \\ \mathscr{b}_5 \\ \mathscr{b}_6 \\ \mathscr{b}_7 \\ \mathscr{b}_8 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 2/\Delta x^2 \\ 0 \\ 2/\Delta x^2 \\ 0 \\ 0 \end{pmatrix}$$

which yields the one-parameter solution

$$\mathscr{b}_j = \begin{cases} 4(\mathscr{a}-1)/\Delta x^2 & \text{for } j=0 \\ (1-2\mathscr{a})/\Delta x^2 & \text{for } j=1,3,5,7 \\ \mathscr{a}/\Delta x^2 & \text{for } j=2,4,6,8, \end{cases}$$

so (4.9) holds if and only if the stencil can be written in the form (4.7). Therefore, we just proved

Theorem 4.1. *Any nine-point stencil that is second order consistent to the Laplace operator can be written in the form*

$$\frac{1}{\Delta x^2}\begin{bmatrix} \mathscr{a} & 1-2\mathscr{a} & \mathscr{a} \\ 1-2\mathscr{a} & 4(\mathscr{a}-1) & 1-2\mathscr{a} \\ \mathscr{a} & 1-2\mathscr{a} & \mathscr{a} \end{bmatrix}$$

and any difference stencil of this form is second order consistent to the Laplace operator. Especially, the choice $\mathscr{a}=1/6$ *yields the nine-point Laplace stencil that appears in the original ILBGK transport scheme* (2.62) *and* $\mathscr{a}=0$ *results in the standard five-point stencil for the Laplace operator* [**29**].

4.2. Stability of the general viscous transport scheme

To reduce diffusion in ILBGK interface tracking, we have to choose the parameter a in equation (4.5) and, therefore, in the general viscous transport scheme (4.6) as large as possible. In the following, we will investigate the stability properties of scheme (4.6) to find the largest value of a that can be chosen without causing numerical instability. In (4.6), the parameter a appears only within the fraction $\Delta t/\left(2a\Delta x^2\right)$ in front of the Laplace stencil, so the choice of a is of course connected to the choice of Δx and Δt.

For the beginning, we consider the simple case $u_\alpha = 0$, so we actually study the scheme

$$\chi(t+\Delta t,\boldsymbol{x}) = \chi(t,\boldsymbol{x}) + \frac{\Delta t}{2a\Delta x^2}\begin{bmatrix} \mathscr{a} & 1-2\mathscr{a} & \mathscr{a} \\ 1-2\mathscr{a} & 4(\mathscr{a}-1) & 1-2\mathscr{a} \\ \mathscr{a} & 1-2\mathscr{a} & \mathscr{a} \end{bmatrix}\chi(t,\boldsymbol{x}) \tag{4.10}$$

which is consistent to the diffusion equation

$$\partial_t\chi = \frac{1}{2a}\partial_{\alpha\alpha}\chi + \mathcal{O}\left(\Delta t,\Delta x^2\right).$$

We will investigate under which conditions (4.10) is $\mathcal{L}^\infty$-stable respectively $\mathcal{L}^2$-stable, i. e. we will check under which conditions

$$\|\chi(t+\Delta t)\|_{\mathcal{L}^\infty} \le \|\chi(t)\|_{\mathcal{L}^\infty} \qquad \text{respectively} \qquad \|\chi(t+\Delta t)\|_{\mathcal{L}^2} \le \|\chi(t)\|_{\mathcal{L}^2}$$

for $t\in[0,t_{max}-\Delta t]$.

Let us start with $\mathcal{L}^\infty$-stability. First of all, we recall that by definition,

$$\|\chi(t+\Delta t)\|_{\mathcal{L}^\infty} = \max_{\boldsymbol{x}}|\chi(t+\Delta t,\boldsymbol{x})| \qquad \text{for } t\in[0,t_{max}-\Delta t].$$

From (4.10), we obtain

$$\begin{aligned}\chi(t+\Delta t,\boldsymbol{x}) &= \left(1+\frac{4(\mathscr{a}-1)\Delta t}{2a\Delta x^2}\right)\chi(t,\boldsymbol{x}) \\ &\quad + \frac{(1-2\mathscr{a})\Delta t}{2a\Delta x^2}\Big(\chi(t,\boldsymbol{x}+\boldsymbol{c}_1\Delta x)+\chi(t,\boldsymbol{x}+\boldsymbol{c}_3\Delta x)+\chi(t,\boldsymbol{x}+\boldsymbol{c}_5\Delta x)+\chi(t,\boldsymbol{x}+\boldsymbol{c}_7\Delta x)\Big) \\ &\qquad + \frac{\mathscr{a}\Delta t}{2a\Delta x^2}\Big(\chi(t,\boldsymbol{x}+\boldsymbol{c}_2\Delta x)+\chi(t,\boldsymbol{x}+\boldsymbol{c}_4\Delta x)+\chi(t,\boldsymbol{x}+\boldsymbol{c}_6\Delta x)+\chi(t,\boldsymbol{x}+\boldsymbol{c}_8\Delta x)\Big)\end{aligned}$$

and therefore, $\chi(t+\Delta t,\boldsymbol{x})$ is of the structure

$$\chi(t+\Delta t,\boldsymbol{x}) = \sum_{j=0}^{8}\gamma_j\chi(t,\boldsymbol{x}+\boldsymbol{c}_j\Delta x) \qquad \text{with} \qquad \sum_{j=0}^{8}\gamma_j = 1.$$

Assuming $\chi(t+\Delta t,\boldsymbol{x})$ is a convex combination of the $\chi(t,\boldsymbol{x}+\boldsymbol{c}_j\Delta x)$, i. e. assuming also $\gamma_j\ge 0$ holds for $j=0,\dots,8$, we find that

$$\min_j \chi(t,\boldsymbol{x}+\boldsymbol{c}_j\Delta x) \le \chi(t+\Delta t,\boldsymbol{x}) \le \max_j \chi(t,\boldsymbol{x}+\boldsymbol{c}_j\Delta x). \tag{4.11}$$

Since for any given $\boldsymbol{x} \in \mathcal{X}$

$$\min_{\boldsymbol{y}\in\mathcal{X}} \chi(t,\boldsymbol{y}) \leq \min_j \chi(t,\boldsymbol{x}+\boldsymbol{c}_j\Delta x) \qquad \text{and} \qquad \max_{\boldsymbol{y}\in\mathcal{X}} \chi(t,\boldsymbol{y}) \geq \max_j \chi(t,\boldsymbol{x}+\boldsymbol{c}_j\Delta x)$$

condition (4.11) immediately implies $\mathcal{L}^\infty$-stability. Thus, for $\mathcal{L}^\infty$-stability it is sufficient to require

$$1+\frac{4\Delta t}{2a\Delta x^2}(\mathscr{a}-1) \geq 0, \qquad \frac{\Delta t}{2a\Delta x^2}\mathscr{a} \geq 0, \qquad \text{and} \qquad \frac{\Delta t}{2a\Delta x^2}(1-2\mathscr{a}) \geq 0$$

and simplifying those conditions we can formulate

Proposition 4.2. *Scheme* (4.10) *is* $\mathcal{L}^\infty$*-stable if*

$$0 \leq \mathscr{a} \leq \frac{1}{2} \qquad \textit{and} \qquad \frac{\Delta t}{a\Delta x^2} \leq \frac{1}{2(1-\mathscr{a})}.$$

By definition, $\mathcal{L}^2$-stability means

$$\left(\frac{1}{N_\mathcal{X}}\sum_{\boldsymbol{x}\in\mathcal{X}} |\chi(t+\Delta t,\boldsymbol{x})|^2\right)^{1/2} \leq \left(\frac{1}{N_\mathcal{X}}\sum_{\boldsymbol{x}\in\mathcal{X}} |\chi(t,\boldsymbol{x})|^2\right)^{1/2} \qquad \text{for each } t\in[0,t_{max}-\Delta t],$$

where $N_\mathcal{X}$ denotes the number of lattice points in $\mathcal{X}$. The above inequality reduces to

$$\sum_{\boldsymbol{x}\in\mathcal{X}} |\chi(t+\Delta t,\boldsymbol{x})|^2 \leq \sum_{\boldsymbol{x}\in\mathcal{X}} |\chi(t,\boldsymbol{x})|^2 \qquad \text{for each } t\in[0,t_{max}-\Delta t].$$

To enforce this condition, we proceed as exemplified for the one-dimensional case by Peyret and Taylor [**60**], i. e. we decompose χ in Fourier space, thus obtaining

$$\chi(t,\boldsymbol{x}) = \sum_{\xi_1,\xi_2=-\infty}^{\infty} X(t,\boldsymbol{\xi})\exp\left(i\xi_\alpha x_\alpha\right) \qquad \text{and} \qquad \chi(t+\Delta t,\boldsymbol{x}) = \sum_{\xi_1,\xi_2=-\infty}^{\infty} X(t+\Delta t,\boldsymbol{\xi})\exp\left(i\xi_\alpha x_\alpha\right),$$

where i denotes the imaginary unit. Afterwards, we insert those results into (4.10) and obtain by straight calculation

$$X(t+\Delta t,\boldsymbol{\xi}) = \left(1+\frac{\Delta t}{2a\Delta x^2}\left(4\mathscr{a}-4+\sum_{j=1,3,5,7}\mathscr{a}\exp\left(i\xi_\alpha c_{j;\alpha}\Delta x\right)+\sum_{j=2,4,6,8}(1-2\mathscr{a})\exp\left(i\xi_\alpha c_{j;\alpha}\Delta x\right)\right)\right)X(t,\boldsymbol{\xi}). \tag{4.12}$$

Our aim is now to prevent the amplitudes of $X(t+\Delta t,\xi)$ for each $t\in[0,t_{max}-\Delta t]$ and arbitrary $\xi\in\mathbb{Z}^2$ from being amplified by (4.12). Defining the so-called amplification factor

$$\mathcal{A}(\boldsymbol{\xi}) = 1+\frac{\Delta t}{2a\Delta x^2}\left(4\mathscr{a}-4+\sum_{j=1,3,5,7}\mathscr{a}\exp\left(i\xi_\alpha c_{j;\alpha}\Delta x\right)+\sum_{j=2,4,6,8}(1-2\mathscr{a})\exp\left(i\xi_\alpha c_{j;\alpha}\Delta x\right)\right) \tag{4.13}$$

we can abbreviate (4.12) by $X(t+\Delta t,\xi)=\mathcal{A}(\xi)X(t,\xi)$. Hence, we obtain

$$X(m\Delta t,\xi) = \left(\prod_{\alpha=1}^{m}\mathcal{A}(\xi)\right)X(0,\xi) \qquad \text{for } m\in\mathbb{N}$$

and therefore,

$$|X(m\Delta t,\xi)| = \left(\prod_{\alpha=1}^{m}|\mathcal{A}(\xi)|\right)|X(0,\xi)| \qquad \text{for } m\in\mathbb{N}.$$

To avoid any amplification of amplitudes of X, we now apply the strict von Neumann condition [**60**] $|\mathcal{A}(\boldsymbol{\xi})|\leq 1$ which implies

$$\left|1+\frac{\Delta t}{2a\Delta x^2}\left(4\mathscr{a}-4+\sum_{j=1,3,5,7}\mathscr{a}\exp\left(i\xi_\alpha c_{j;\alpha}\Delta x\right)+\sum_{j=2,4,6,8}(1-2\mathscr{a})\exp\left(i\xi_\alpha c_{j;\alpha}\Delta x\right)\right)\right| \leq 1 \tag{4.14}$$

for each $0\leq\xi_\alpha c_{j;\alpha}\Delta x<2\pi$. This yields

Proposition 4.3. *The scheme* (4.10) *is* $\mathcal{L}^2$*-stable if condition* (4.14) *is fulfilled.*

```
let ɑ = 0 and Δx = 2π/N_X;
while ɑ < 1/2
    let Δt = 0 and A_max(ɑ) = 0;
    repeat
        let Δt = Δt + 1/(10 N_X);
        determine A according to (4.13);
        if max_ξ |A| ≤ 1 then A_max(ɑ) = max_ξ |A|;
    until max_ξ |A| > 1;
    let ɑ = ɑ + 1/(2 N_X);
```

Algorithm 4.1: Test for $\mathcal{L}^2$-stability of scheme (4.10)

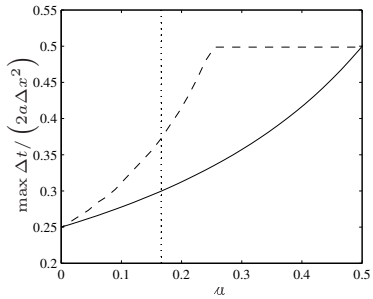


Figure 4.1: Maximal value of $\Delta t/\left(2a\Delta x^2\right)$ for which (4.10) is still $\mathcal{L}^2$-stable (dashed line) respectively $\mathcal{L}^\infty$-stable (solid line). The axis of ordinates represents the values of $\Delta t/\left(2a\Delta x^2\right)$ while the axis of abscissae shows the stencil parameter α. The dotted vertical line marks the original ILBGK scheme with $\alpha = 1/6$ and $a = 3$.

Since (4.14) is very complicated to evaluate we will investigate it numerically using Algorithm 4.1. The result on a 40×40 lattice is compared to the conditions for $\mathcal{L}^\infty$-stability in Figure 4.1. Note that up to a negligibly small discretisation error, calculations on finer grids yield the same result. It is clear that the classical five point stencil for the Laplace operator ($\alpha = 0$) has the worst stability properties while the best stencil is in some sense quite its opposite ($\alpha = 1/2$). The ILBGK scheme ($\alpha = 1/6, a = 3$) is $\mathcal{L}^2$-stable for $\Delta t/\Delta x^2 \lesssim 2.2388$ and $\mathcal{L}^\infty$-stable for $\Delta t/\Delta x^2 \leq 9/5$. Since ILBGK is based on the scaling $\Delta t = \Delta x^2$ it is in fact unconditionally stable in the $\mathcal{L}^2$ norm and the $\mathcal{L}^\infty$ norm.

Let us now consider the general transport scheme (4.6) for a constant non-zero velocity field $\boldsymbol{u}$. Adapting the above arguments we obtain the conditions

$$\frac{\Delta t}{2a\Delta x^2} \leq \frac{1}{4 - 4\alpha},$$

$$\begin{aligned} (1-2\alpha)\left(\frac{1}{a\Delta x} + u_1\right) &\geq 0, \qquad (1-2\alpha)\left(\frac{1}{a\Delta x} - u_1\right) \geq 0, \\ (1-2\alpha)\left(\frac{1}{a\Delta x} + u_2\right) &\geq 0, \qquad (1-2\alpha)\left(\frac{1}{a\Delta x} - u_2\right) \geq 0, \end{aligned} \tag{4.15a}$$

and

$$\begin{aligned} \alpha\left(\frac{1}{a\Delta x} + u_1 + u_2\right) &\geq 0, \qquad \alpha\left(\frac{1}{a\Delta x} - u_1 + u_2\right) \geq 0, \\ \alpha\left(\frac{1}{a\Delta x} - u_1 - u_2\right) &\geq 0, \qquad \alpha\left(\frac{1}{a\Delta x} + u_1 - u_2\right) \geq 0 \end{aligned} \tag{4.15b}$$

for $\mathcal{L}^\infty$-stability. Let us first consider (4.15a) for the case $\alpha > 1/2$. This yields

$$\frac{1}{a\Delta x} + u_\alpha \leq 0 \qquad \text{as well as} \qquad \frac{1}{a\Delta x} - u_\alpha \leq 0$$

and multiplying the latter inequality with -1 we obtain

$$\frac{1}{a\Delta x} \leq u_\alpha \leq -\frac{1}{a\Delta x}.$$

This contradicts the assumption $a > 0$ and therefore, $\mathscr{a} \leq 1/2$ is a first condition for $\mathcal{L}^\infty$-stability of (4.6). Using this result we summarise (4.15a) in the form

$$|u_\alpha| \leq \frac{1}{a\Delta x} \qquad \text{for } \mathscr{a} < 1/2.$$

From (4.15b), we conclude that in the case $\mathscr{a} < 0$ we must require

$$\frac{1}{a\Delta x} \leq u_1 + u_2 \leq -\frac{1}{a\Delta x}$$

and therefore, $\mathscr{a} \geq 0$ is also necessary for $\mathcal{L}^\infty$-stability. Thus, (4.15b) implies

$$|u_1| + |u_2| \leq \frac{1}{a\Delta x}$$

for $\mathscr{a} > 0$ and we can finally formulate

Proposition 4.4. *For a constant velocity field $\boldsymbol{u}$, the general transport scheme* (4.6) *is $\mathcal{L}^\infty$-stable if*

$$0 \leq \mathscr{a} \leq \frac{1}{2}, \qquad \frac{\Delta t}{a\Delta x^2} \leq \frac{1}{2(1-\mathscr{a})},$$

$$|u_\alpha| \leq \frac{1}{a\Delta x} \quad \textit{for } \mathscr{a} \neq \frac{1}{2}, \qquad \textit{and} \qquad |u_1| + |u_2| \leq \frac{1}{a\Delta x} \quad \textit{for } \mathscr{a} \neq 0. \tag{4.16}$$

Thus, the maximal value of a for which (4.6) *is $\mathcal{L}^\infty$-stable in the case $\mathscr{a} \neq 0$ is*

$$a = \frac{1}{\Delta x\,(|u_1| + |u_2|)}.$$

Analysing $\mathcal{L}^2$-stability we find the general transport scheme (4.6) to yield the amplification factor

$$\mathcal{B}(\boldsymbol{\xi}) = 1 + \frac{\Delta t}{2\Delta x}\left(\frac{4\mathscr{a}-4}{a\Delta x} + \sum_{j=1,3,5,7} \mathscr{a}\left(\frac{1}{a\Delta x} + c_{j;\alpha}u_\alpha\right)e^{i\xi_\alpha c_{j;\alpha}\Delta x} + \sum_{j=2,4,6,8}(1-2\mathscr{a})\left(\frac{1}{a\Delta x} + c_{j;\alpha}u_\alpha\right)e^{i\xi_\alpha c_{j;\alpha}\Delta x}\right),$$

for $\boldsymbol{u} = \text{const}$, so we obtain

Proposition 4.5. *For a constant velocity field $\boldsymbol{u}$, the general transport scheme* (4.6) *is $\mathcal{L}^2$-stable if* $|\mathcal{B}| \leq 1$.

However, replacing $\mathcal{A}$ with $\mathcal{B}$ in Algorithm 4.1 has no influence on the result of the analysis. Therefore, as long as conditions (4.16) hold, plotting the maximal value of $\Delta t/\left(2a\Delta x^2\right)$ for which (4.6) is still $\mathcal{L}^2$-stable respectively $\mathcal{L}^\infty$-stable results again in the chart already presented in Figure 4.1. Thus, we conclude that stability of the general viscous transport scheme (4.6) is governed by stability of its parabolic part (4.10).

4.3. Numerical experiments with improved ILBGK interface tracking

From Section 4.2, we know that for a constant velocity field $\boldsymbol{u}$ the general viscous transport scheme (4.6) allows the largest time step size and suffers least diffusion if

$$\mathscr{a} = \frac{1}{2} \qquad \text{and} \qquad a = \frac{1}{\Delta x\,(|u_1| + |u_2|)} \qquad \text{for } \boldsymbol{u} \in \left\{\boldsymbol{u}^{(1)}, \boldsymbol{u}^{(2)}\right\}. \tag{4.17}$$

Furthermore, we know that

$$\max \frac{\Delta t}{2a\Delta x^2} = \frac{1}{2} \qquad \text{and therefore,} \qquad \Delta t_{max} = a\Delta x^2 = \frac{\Delta x}{|u_1| + |u_2|},$$

```
initialise χ and u;
choose ⍼ and a according to (4.17);
let t = 0;
while t < t_max
    determine χ(t + Δt, x) according to (4.18);
    let t = t + Δt;
```

Algorithm 4.2: Interface tracking with scheme (4.18)

so scheme (4.6) takes the form

$$\chi(t+\Delta t,\boldsymbol{x}) = \chi(t,\boldsymbol{x}) + \frac{1}{4}\begin{bmatrix}1 & 0 & 1\\ 0 & -4 & 0\\ 1 & 0 & 1\end{bmatrix}\chi(t,\boldsymbol{x}) - \frac{1}{4\left(|u_1|+|u_2|\right)}\begin{bmatrix}-1 & 0 & 1\\ 0 & 0 & 0\\ -1 & 0 & 1\end{bmatrix}(\chi u_1)(t,\boldsymbol{x})$$
$$- \frac{1}{4\left(|u_1|+|u_2|\right)}\begin{bmatrix}1 & 0 & 1\\ 0 & 0 & 0\\ -1 & 0 & -1\end{bmatrix}(\chi u_2)(t,\boldsymbol{x}). \quad (4.18)$$

We perform benchmark simulations for the above scheme with $\boldsymbol{u} = \boldsymbol{u}^{(1)}$ and $\boldsymbol{u} = \boldsymbol{u}^{(2)}$. The largest possible value of a is then ten respectively five and therefore, we need 40 respectively 80 time steps for one revolution on the 40×40 lattice as well as 80 respectively 160 time steps on the 80×80 lattice. Algorithm 4.2 gives an overview of the whole procedure. Simulation results are collected in Figures 4.2 and 4.3 while mass loss and computing times are shown in Table 4.1. Note that for $\boldsymbol{u} = \boldsymbol{u}^{(1)}$ and $\Delta t = 10\Delta x$ the viscous transport scheme (4.18) reduces to

$$\chi(t+\Delta t,\boldsymbol{x}) = \chi(t,\boldsymbol{x}) + \frac{1}{2}\begin{bmatrix}1 & 0 & 0\\ 0 & -2 & 0\\ 1 & 0 & 0\end{bmatrix}\chi(t,\boldsymbol{x})$$

which, on the given assumptions, approximates

$$\partial_t\chi + \partial_1\chi u_1 = \frac{\Delta x}{20}\partial_{\alpha\alpha}\chi,$$

for which in those directions parallel to the flow field $\boldsymbol{u}^{(1)}$, the transport term clearly dominates diffusion. Nevertheless, we learn from the plots that there is still too much diffusion in the scheme, so we must apply the recolouring step. However, it should be mentioned that both mass loss and computing time for the benchmark are highly competitive.

Following Algorithm 4.3 we simulate the benchmark for $\boldsymbol{u} \in \left\{\boldsymbol{u}^{(1)}, \boldsymbol{u}^{(2)}\right\}$ using (4.18) and the recolouring step, where we replace the uncoloured pseudo-particle density $\not{f}(\boldsymbol{u};\boldsymbol{v})$ with

$$\not{f}_{gen}(\boldsymbol{u};\boldsymbol{v}) = f^*_{gen}\left(1 + a\Delta x u_\alpha v_\alpha\right)$$

for the above choice of parameters. If $\boldsymbol{u} = \boldsymbol{u}^{(3)}$ we cannot use (4.17), so we define

$$⍼ = \frac{1}{2} \qquad \text{and} \qquad a = \frac{1}{\Delta x \max_{\boldsymbol{x}}\left\{|u_1|+|u_2|\right\}} \qquad \text{for } \boldsymbol{u} = \boldsymbol{u}^{(3)}. \quad (4.19)$$

Therefore, we need approximately 264 time steps for one revolution on the 40×40 lattice and approximately 515 time steps for one revolution on the 80×80 lattice.

The corresponding results are just perfect for $\boldsymbol{u} = \boldsymbol{u}^{(1)}$ and quite unsatisfactory for $\boldsymbol{u} \in \left\{\boldsymbol{u}^{(2)}, \boldsymbol{u}^{(3)}\right\}$, as can be seen in Figures 4.4 to 4.6 and Table 4.2. Still, we chose Δt such that $\Delta t/\left(2a\Delta x^2\right) = 1/2$ which is exactly the border between the stable and the unstable regime (recall Figure 4.1) and choosing Δt not quite as large may avoid some severe numerical difficulties. Therefore, we present results computed with Algorithm 4.3 for the lattice BGK time step size $\Delta t = \Delta x^2$ in Figures 4.7 to 4.9 and Table 4.3.

In the given situation, the result for $\boldsymbol{u}(0,\boldsymbol{x}) = \boldsymbol{u}^{(1)}$ is worse than before, that for $\boldsymbol{u}(0,\boldsymbol{x}) = \boldsymbol{u}^{(2)}$ is better, and that for $\boldsymbol{u}(0,\boldsymbol{x}) = \boldsymbol{u}^{(3)}$ is of the same quality as its predecessor. Since we cannot obtain any substantial improvement for the general case we decide to give up this method.

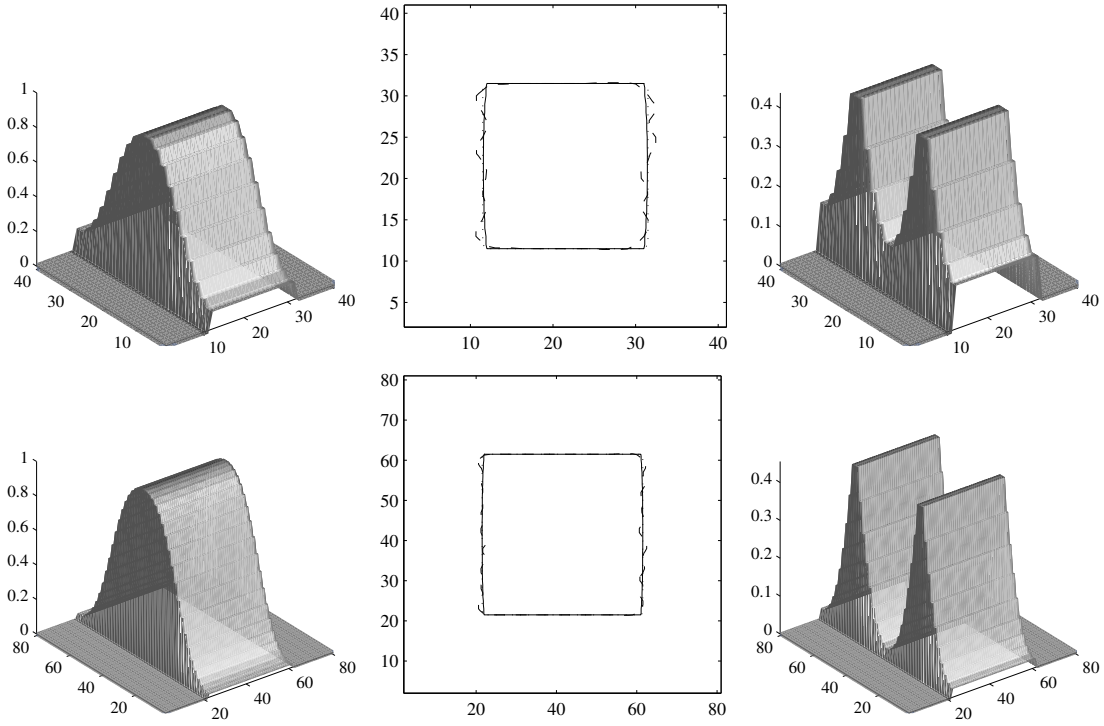


Figure 4.2: Solution of the benchmark problem with Algorithm 4.2 for $\boldsymbol{u} = \boldsymbol{u}^{(1)}$ on a 40×40 (top line) and an 80×80 (bottom line) lattice. Left: surface plot of the simulation result, middle: contour lines for $\chi = 1/2$ of the simulation result (solid line), the correct solution (dotted line), and the outcome of ILBGK interface tracking (dashed line), right: absolute value of the Euclidean error in the simulation result.

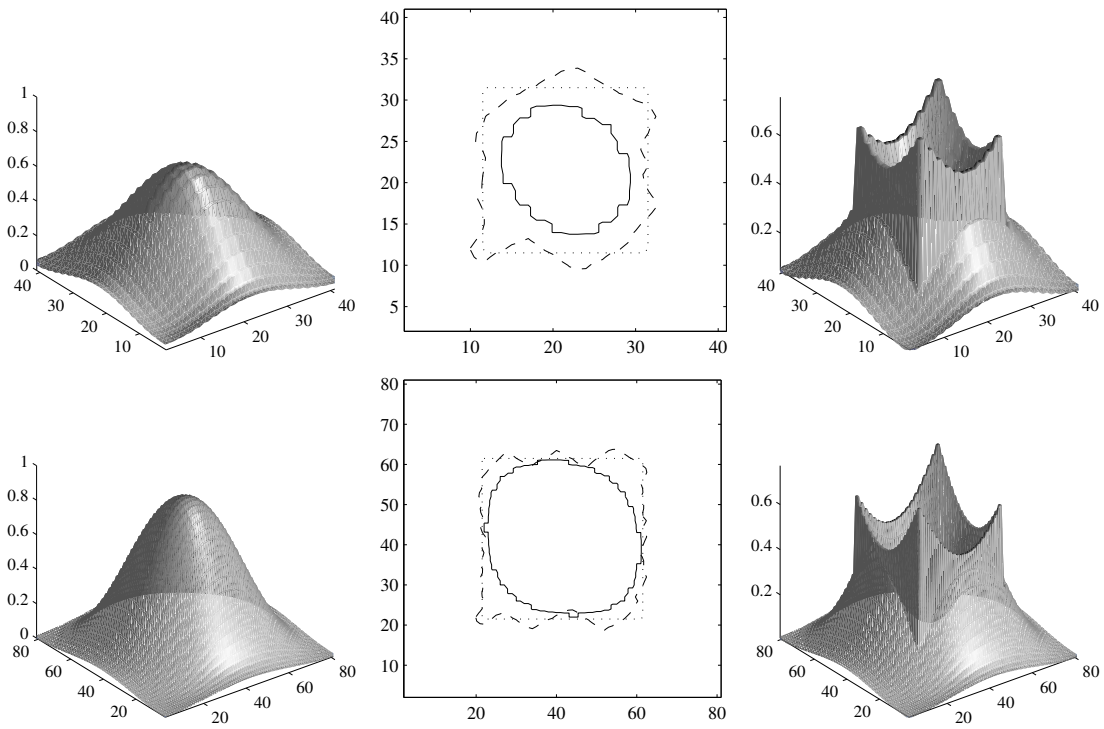


Figure 4.3: Solution of the benchmark problem with Algorithm 4.2 for $\boldsymbol{u} = \boldsymbol{u}^{(2)}$ on a 40×40 (top line) and an 80×80 (bottom line) lattice. Left: surface plot of the simulation result, middle: contour lines for $\chi = 1/2$ of the simulation result (solid line), the correct solution (dotted line), and the outcome of ILBGK interface tracking (dashed line), right: absolute value of the Euclidean error in the simulation result.

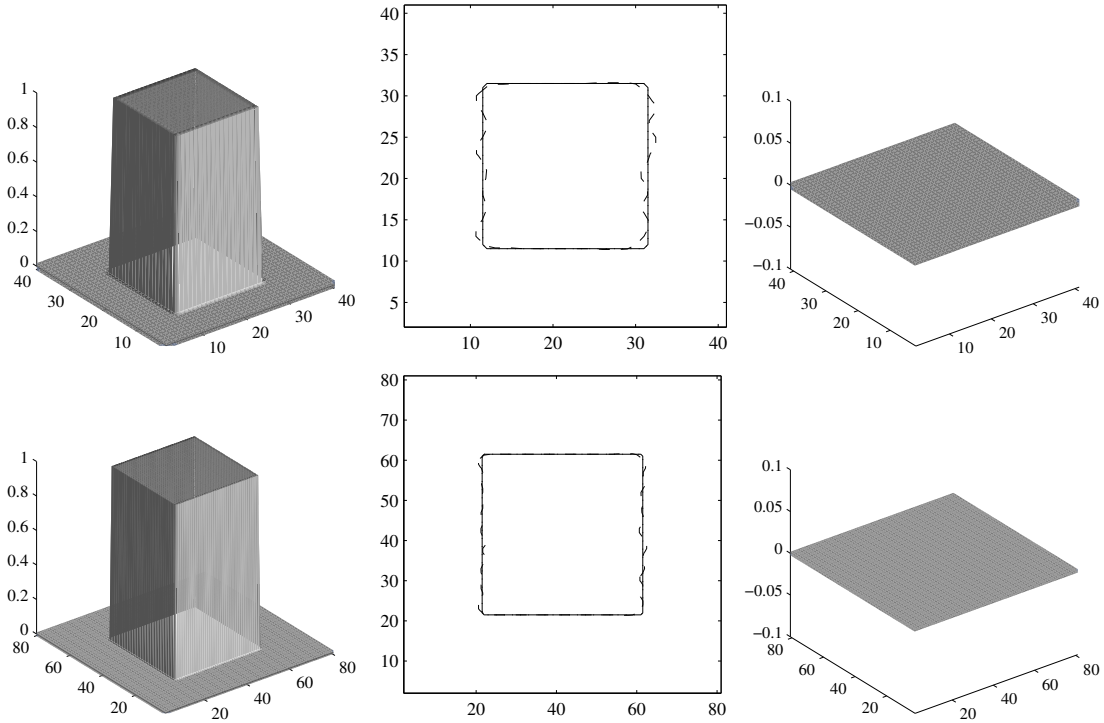


Figure 4.4: Solution of the benchmark problem with Algorithm 4.3 for $\boldsymbol{u} = \boldsymbol{u}^{(1)}$ and $\Delta t = a\Delta x^2$ on a 40×40 (top line) and an 80×80 (bottom line) lattice. Left: surface plot of the simulation result, middle: contour lines for $\chi = 1/2$ of the simulation result (solid line), the correct solution (dotted line), and the outcome of ILBGK interface tracking (dashed line), right: absolute value of the Euclidean error in the simulation result.

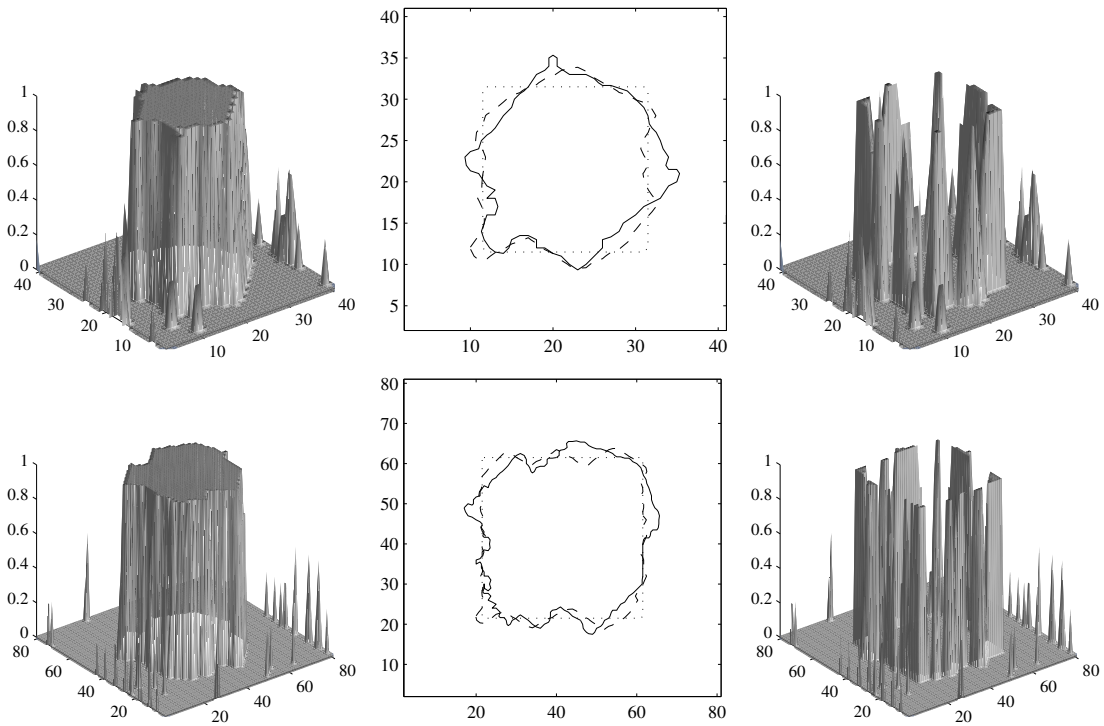


Figure 4.5: Solution of the benchmark problem with Algorithm 4.3 for $\boldsymbol{u} = \boldsymbol{u}^{(2)}$ and $\Delta t = a\Delta x^2$ on a 40×40 (top line) and an 80×80 (bottom line) lattice. Left: surface plot of the simulation result, middle: contour lines for $\chi = 1/2$ of the simulation result (solid line), the correct solution (dotted line), and the outcome of ILBGK interface tracking (dashed line), right: absolute value of the Euclidean error in the simulation result.

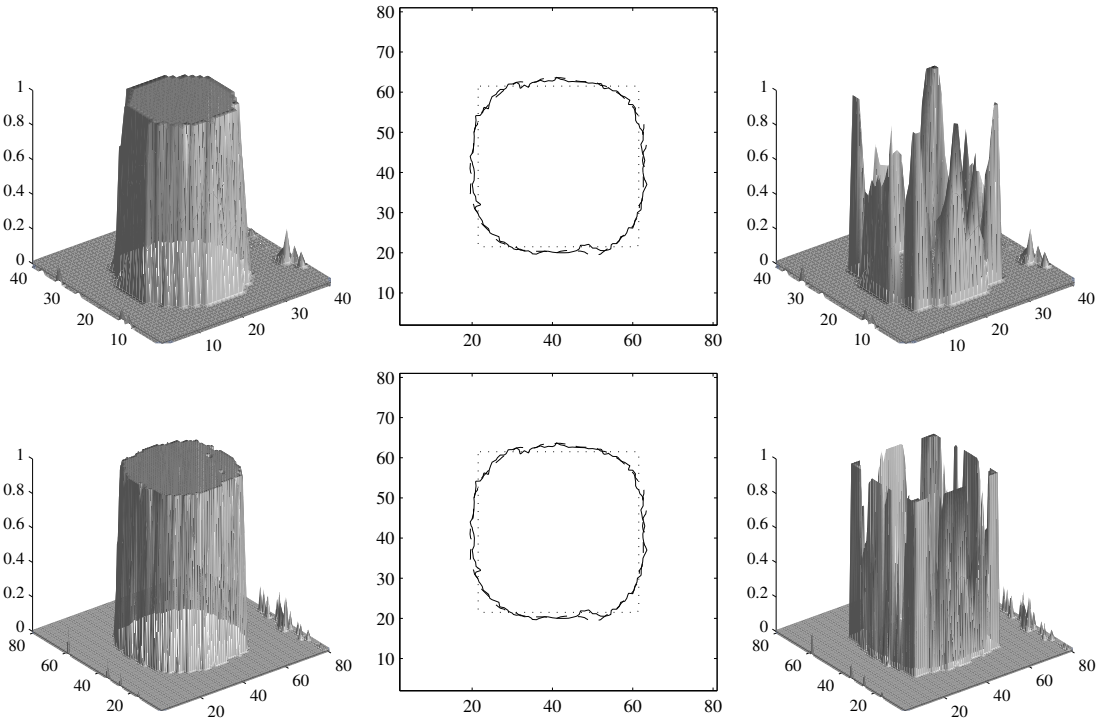


Figure 4.6: Solution of the benchmark problem with Algorithm 4.3 for $\boldsymbol{u} = \boldsymbol{u}^{(3)}$ and $\Delta t = a\Delta x^2$ on a 40×40 (top line) and an 80×80 (bottom line) lattice. Left: surface plot of the simulation result, middle: contour lines for $\chi = 1/2$ of the simulation result (solid line), the correct solution (dotted line), and the outcome of ILBGK interface tracking (dashed line), right: absolute value of the Euclidean error in the simulation result.

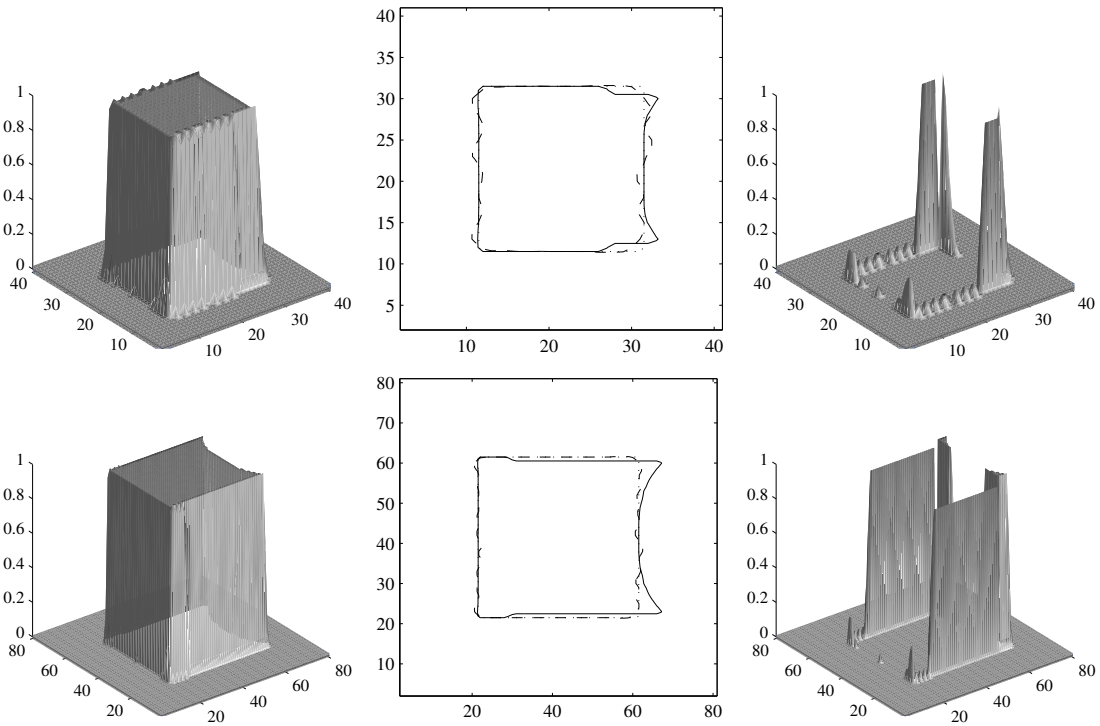


Figure 4.7: Solution of the benchmark problem with Algorithm 4.3 for $\boldsymbol{u} = \boldsymbol{u}^{(1)}$ and $\Delta t = \Delta x^2$ on a 40×40 (top line) and an 80×80 (bottom line) lattice. Left: surface plot of the simulation result, middle: contour lines for $\chi = 1/2$ of the simulation result (solid line), the correct solution (dotted line), and the outcome of ILBGK interface tracking (dashed line), right: absolute value of the Euclidean error in the simulation result.

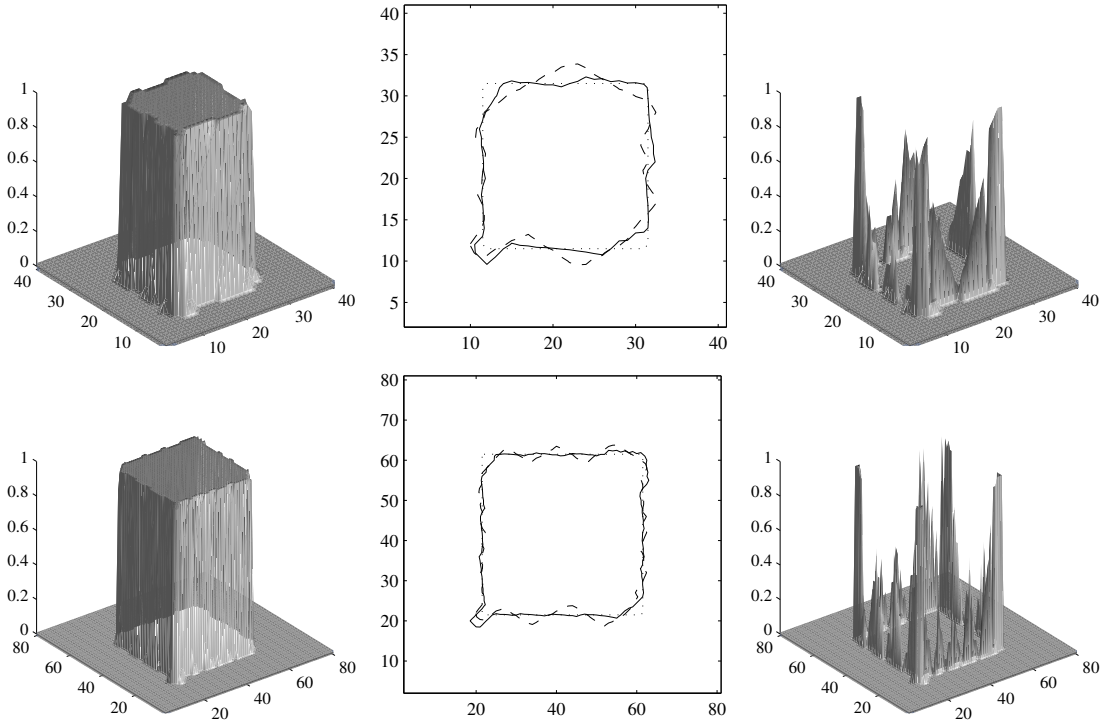


Figure 4.8: Solution of the benchmark problem with Algorithm 4.3 for $\boldsymbol{u} = \boldsymbol{u}^{(2)}$ and $\Delta t = \Delta x^2$ on a 40×40 (top line) and an 80×80 (bottom line) lattice. Left: surface plot of the simulation result, middle: contour lines for $\chi = 1/2$ of the simulation result (solid line), the correct solution (dotted line), and the outcome of ILBGK interface tracking (dashed line), right: absolute value of the Euclidean error in the simulation result.

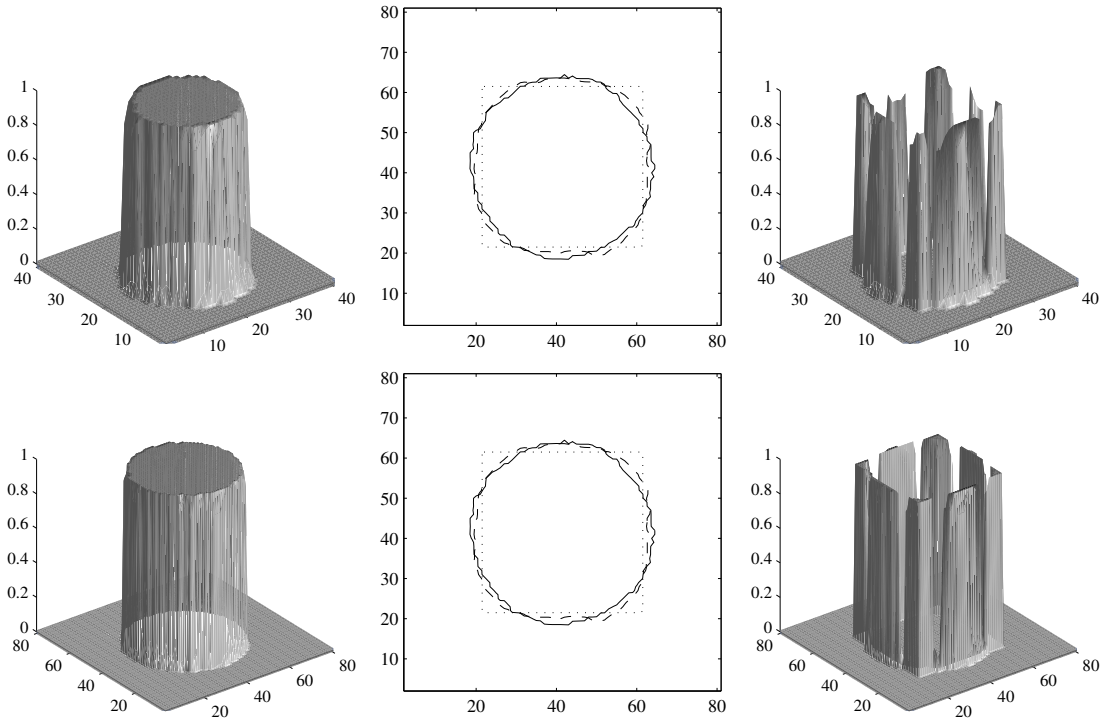


Figure 4.9: Solution of the benchmark problem with Algorithm 4.3 for $\boldsymbol{u} = \boldsymbol{u}^{(3)}$ and $\Delta t = \Delta x^2$ on a 40×40 (top line) and an 80×80 (bottom line) lattice. Left: surface plot of the simulation result, middle: contour lines for $\chi = 1/2$ of the simulation result (solid line), the correct solution (dotted line), and the outcome of ILBGK interface tracking (dashed line), right: absolute value of the Euclidean error in the simulation result.

initialise χ and $\boldsymbol{u}$;
if $\boldsymbol{u} \in \left\{\boldsymbol{u}^{(1)}, \boldsymbol{u}^{(2)}\right\}$ choose $\mathscr{a}$ and a according to (4.17);
if $\boldsymbol{u} = \boldsymbol{u}^{(3)}$ choose $\mathscr{a}$ and a according to (4.19);
let $t = 0$;
while $t < t_{max}$
  do recolouring according to Algorithm 2.8 with ℓ_{gen} instead of ℓ;
  determine $\chi(t + \Delta t, \boldsymbol{x})$ according to (2.66);
  let $t = t + \Delta t$;

Algorithm 4.3: Interface tracking with equation (4.1) and recolouring

Lattice	$\boldsymbol{u}(0, \boldsymbol{x})$	Mass loss in %	Number of flops
40×40	$\boldsymbol{u}^{(1)}$	0	$5.11392 \cdot 10^6$
40×40	$\boldsymbol{u}^{(2)}$	$-4.26326 \cdot 10^{-14}$	$1.02278 \cdot 10^7$
40×40	$\boldsymbol{u}^{(3)}$	0	$3.37519 \cdot 10^7$
80×80	$\boldsymbol{u}^{(1)}$	0	$3.88550 \cdot 10^7$
80×80	$\boldsymbol{u}^{(2)}$	$-4.26326 \cdot 10^{-14}$	$7.77101 \cdot 10^7$
80×80	$\boldsymbol{u}^{(3)}$	$-1.98952 \cdot 10^{-13}$	$2.50129 \cdot 10^8$

Table 4.1: Percentage loss of red mass and computing times for the simulations with Algorithm 4.2

Lattice	$\boldsymbol{u}(0, \boldsymbol{x})$	Mass loss in %	Number of flops
40×40	$\boldsymbol{u}^{(1)}$	0	$1.05043 \cdot 10^7$
40×40	$\boldsymbol{u}^{(2)}$	0	$2.11877 \cdot 10^7$
40×40	$\boldsymbol{u}^{(3)}$	$2.41585 \cdot 10^{-13}$	$7.00696 \cdot 10^7$
80×80	$\boldsymbol{u}^{(1)}$	0	$8.14550 \cdot 10^7$
80×80	$\boldsymbol{u}^{(2)}$	0	$1.64395 \cdot 10^8$
80×80	$\boldsymbol{u}^{(3)}$	$7.38964 \cdot 10^{-13}$	$5.29820 \cdot 10^8$

Table 4.2: Percentage loss of red mass and computing times for the simulations with Algorithm 4.3 for $\Delta t = a\Delta x^2$

Lattice	$\boldsymbol{u}(0, \boldsymbol{x})$	Mass loss in %	Number of flops
40×40	$\boldsymbol{u}^{(1)}$	$5.82645 \cdot 10^{-13}$	$1.05249 \cdot 10^8$
40×40	$\boldsymbol{u}^{(2)}$	$1.73372 \cdot 10^{-12}$	$1.05886 \cdot 10^8$
40×40	$\boldsymbol{u}^{(3)}$	$1.15108 \cdot 10^{-12}$	$2.65433 \cdot 10^8$
80×80	$\boldsymbol{u}^{(1)}$	$5.96856 \cdot 10^{-13}$	$8.15681 \cdot 10^8$
80×80	$\boldsymbol{u}^{(2)}$	$3.42482 \cdot 10^{-12}$	$8.20843 \cdot 10^8$
80×80	$\boldsymbol{u}^{(3)}$	$4.95959 \cdot 10^{-12}$	$2.05742 \cdot 10^9$

Table 4.3: Percentage loss of red mass and computing times for the simulations with Algorithm 4.3 for $\Delta t = \Delta x^2$

CHAPTER 5

Volume tracking lattice BGK

After we separated interface tracking from the Navier-Stokes solver in ILBGK, we now want to combine the ILBGK Navier-Stokes solver with a classical method for interface tracking, namely the conservative volume tracking method. Even though immiscible lattice BGK and conservative volume tracking seem to be based on completely different ideas, we will see that they have a lot of things in common.

We first introduce a conservative volume tracking method and apply it to the benchmark in Section 5.1, then we point out mutualities and differences of ILBGK and conservative volume tracking in Section 5.2. Finally, in Section 5.3, we introduce a volume tracking lattice BGK method which solves the Navier-Stokes equations with lattice BGK and determines the motion of interface with volume tracking. At the end of Section 5.3, this method is applied to the benchmark and the results are compared to those achieved with plain ILBGK.

5.1. Conservative volume tracking

In this Section, we will consider a conservative volume tracking method [**24, 46, 62, 65**] which is based on the ideas of Rider and Kothe [**65**] in their most simple form. Let us therefore interpret each lattice point $\boldsymbol{x} \in \mathcal{X}$ as the centre point of a quadratic grid cell $\mathcal{L}(\boldsymbol{x})$, as shown in Figure 5.1. Choosing all cell edges to be of length Δx we make sure that the uniform lattice $\mathcal{X}$ is then covered by a uniform grid. Now, the basic idea of volume tracking is to interpret $\chi(\boldsymbol{x})$ as that volume fraction of grid cell $\mathcal{L}(\boldsymbol{x})$ which is filled with red fluid and to solve the transport equation (2.13) with a conservative algorithm based on the so-called volume fluxes of red fluid across the cell boundaries.

In Section 5.1.1, we will discuss the basic prerequisite of volume tracking, namely the reconstruction of the interface in a given grid cell. Section 5.1.2 deals with the actual volume tracking algorithm and in Section 5.1.3, we present some numerical results.

5.1.1. Interface reconstruction. We are particularly interested in cells containing both red and blue fluid, i. e. cells with $0 < \chi_{\boldsymbol{x}} < 1$, where

$$\chi_{\boldsymbol{x}}(t) = \frac{1}{\Delta x^2} \int_{\mathcal{L}(\boldsymbol{x})} \chi(t, \boldsymbol{y})\, d\boldsymbol{y}. \tag{5.1}$$

Remember that all edges of $\mathcal{L}(\boldsymbol{x})$ have length Δx, so the volume of $\mathcal{L}(\boldsymbol{x})$ is exactly Δx^2. Let us now assume each $\Gamma_{\boldsymbol{x}} = \mathcal{L}(\boldsymbol{x}) \cap \Gamma$ to be a straight line, so we can describe it with the line equation

$$n_{\boldsymbol{x};\alpha} y_\alpha + L_{\boldsymbol{x}} = 0 \qquad \text{for } \boldsymbol{y} \in \Gamma_{\boldsymbol{x}}, \tag{5.2}$$

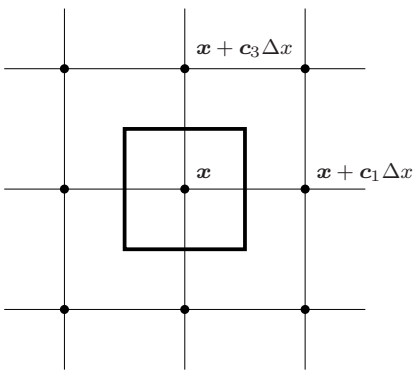


FIGURE 5.1: The quadratic grid cell $\mathcal{L}(\boldsymbol{x})$ (boldface square) around lattice point $\boldsymbol{x}$

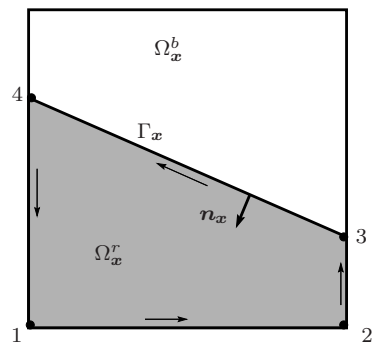


Figure 5.2: Example of a reconstructed interface in a grid cell. Points 1 to 4 are collected counter clockwise for the calculation of $V\left(\Omega_{\boldsymbol{x}}^r\right)$ with equation (5.4).

where $\boldsymbol{n}_{\boldsymbol{x}} = \boldsymbol{n}(\boldsymbol{x})$ is the inner unit normal vector of $\Gamma_{\boldsymbol{x}}$ and the line constant $L_{\boldsymbol{x}}$ is still to be determined. Since by definition, $\boldsymbol{n}$ points into the red liquid we have in general

$$n_{\boldsymbol{x};\alpha} y_\alpha + L_{\boldsymbol{x}} \begin{cases} > 0 & \text{for } \boldsymbol{y} \in \Omega_{\boldsymbol{x}}^r \\ = 0 & \text{for } \boldsymbol{y} \in \Gamma_{\boldsymbol{x}} \\ < 0 & \text{for } \boldsymbol{y} \in \Omega_{\boldsymbol{x}}^b \end{cases} \tag{5.3}$$

with $\Omega_{\boldsymbol{x}}^r = \mathcal{L}(\boldsymbol{x}) \cap \Omega^r$ and $\Omega_{\boldsymbol{x}}^b = \mathcal{L}(\boldsymbol{x}) \cap \Omega^b$.

Suppose for a moment $L_{\boldsymbol{x}}$ is known. Then, we can use (5.3) to check whether or not a given point $\boldsymbol{y} \in \mathcal{L}(\boldsymbol{x})$ belongs to $\Omega_{\boldsymbol{x}}^r$. Since $\Gamma_{\boldsymbol{x}}$ is assumed to be a straight line and $\mathcal{L}(\boldsymbol{x})$ is known to be a square $\Omega_{\boldsymbol{x}}^r$ must be a triangle, a tetragon, or a pentagon. The vertices of the polygon $\Omega_{\boldsymbol{x}}^r$ consist exactly of the vertices of $\mathcal{L}(\boldsymbol{x})$ that lie inside the red phase and the intersection points of $\Gamma_{\boldsymbol{x}}$ with the cell edges of $\mathcal{L}(\boldsymbol{x})$. Once for $m = 1, \ldots, \mathcal{M}$, with $\mathcal{M} \in \{3, 4, 5\}$, the vertices $\boldsymbol{y}^m$ of $\Omega_{\boldsymbol{x}}^r$ are known in Cartesian coordinates, we can calculate the volume $V\left(\Omega_{\boldsymbol{x}}^r\right)$ of $\Omega_{\boldsymbol{x}}^r$ using the formula

$$V\left(\Omega_{\boldsymbol{x}}^r\right) = \frac{1}{2} \sum_{m=1}^{\mathcal{M}} \left(y_1^m y_2^{m+1} - y_1^{m+1} y_2^m\right), \tag{5.4}$$

where $\boldsymbol{y}^{\mathcal{M}+1} = \boldsymbol{y}^1$ is implicitly assumed (see Figure 5.2 for an exemplifying illustration). Afterwards, we express the volume fraction $\chi_{\boldsymbol{x}}$ occupied by red fluid in the form $\chi_{\boldsymbol{x}} = V\left(\Omega_{\boldsymbol{x}}^r\right)/\Delta x^2$ or vice versa, we have $V\left(\Omega_{\boldsymbol{x}}^r\right) = \chi_{\boldsymbol{x}} \Delta x^2$.

To compute $L_{\boldsymbol{x}}$, we first pass a line perpendicular to $\boldsymbol{n}_{\boldsymbol{x}}$ through each vertex of $\mathcal{L}(\boldsymbol{x})$ and denote the corresponding line constant by $L_{\boldsymbol{x}}^m$, where m specifies the vertex (see Figure 5.3 for more details). Together with the cell edges of $\mathcal{L}(\boldsymbol{x})$, those lines form a polygon which is uniquely defined by the condition that $\boldsymbol{n}_{\boldsymbol{x}}$ must point into its interior (recall Figure 5.2). The volume $V\left(L_{\boldsymbol{x}}^m\right)$ of this polygon is named truncation volume. If one of the $L_{\boldsymbol{x}}^m$ solves now $V\left(L_{\boldsymbol{x}}^m\right) = V\left(\Omega_{\boldsymbol{x}}^r\right)$ we are already done. Otherwise, we compute $V\left(L_{\boldsymbol{x}}^m\right) - V\left(\Omega_{\boldsymbol{x}}^r\right)$ for each $L_{\boldsymbol{x}}^m$ and sort the results in ascending order. Then, we pick $L_{\boldsymbol{x}}^+$ such that

$$V\left(L_{\boldsymbol{x}}^+\right) = \min_{m=1\ldots4} \left\{V\left(L_{\boldsymbol{x}}^m\right) : V\left(L_{\boldsymbol{x}}^m\right) - V\left(\Omega_{\boldsymbol{x}}^r\right) > 0\right\}$$

and $L_{\boldsymbol{x}}^-$ such that

$$V\left(L_{\boldsymbol{x}}^-\right) = \max_{m=1\ldots4} \left\{V\left(L_{\boldsymbol{x}}^m\right) : V\left(L_{\boldsymbol{x}}^m\right) - V\left(\Omega_{\boldsymbol{x}}^r\right) < 0\right\}.$$

Afterwards, we interpolate the function $V\left(L_{\boldsymbol{x}}\right) - V\left(\Omega_{\boldsymbol{x}}^r\right)$ between $L_{\boldsymbol{x}}^-$ and $L_{\boldsymbol{x}}^+$. Finally, we use Algorithm 5.1, which is typically referred to as Brent's algorithm [**6, 61**], to solve

$$Vol\left(L_{\boldsymbol{x}}\right) = V\left(L_{\boldsymbol{x}}\right) - V\left(\Omega_{\boldsymbol{x}}^r\right) = 0$$

for $L_{\boldsymbol{x}}$. Knowing $\boldsymbol{n}_{\boldsymbol{x}}$ and $L_{\boldsymbol{x}}$ we are now able to completely describe $\Gamma_{\boldsymbol{x}}$ by equation (5.2).

5.1.2. Volume tracking. The next step will be the determination of the fluxes of red fluid across the cell boundaries. Those fluxes correspond to the amount of red fluid flowing from $\mathcal{L}\boldsymbol{x}$ to a neighbouring

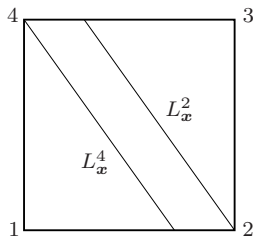


Figure 5.3: Two lines with identical normal vector passing through vertices 2 respectively 4. Those lines are marked with their particular line constants $L_{\boldsymbol{x}}^2$ and $L_{\boldsymbol{x}}^4$. The parallel lines passing through vertices 1 and 3 intersect the grid cell in one point only, so the corresponding truncation volume can only be zero or Δx^2.

```
let ℵ1 = L_x^-, ℵ2 = L_x^+, and ℵ3 = ℵ1;
while Vol(ℵ2) ≠ 0
   if Vol(ℵ2) Vol(ℵ3) > 0 let ℵ3 = ℵ1;
   if |Vol(ℵ3)| < |Vol(ℵ2)| exchange ℵ2 with ℵ3 and let ℵ1 = ℵ3;
   let ϑ = (ℵ3 − ℵ2)/2;
   if ϑ ≈ 0 or Vol(ℵ2) ≈ 0 stop here, ℵ2 is the solution;
   else
      try to update ℵ2 with one step of inverse quadratic interpolation;
      if that works out well go on, everything is fine;
      else
         update ℵ2 with a bisection step;
```

Algorithm 5.1: Brent's algorithm

grid cell. In particular, we apply here a conservative fractional step (or operator split) algorithm [**62, 65**], i. e. we apply a scheme of the form

$$\chi_{\boldsymbol{x}}\left(t+\frac{\Delta t}{2}\right)=\chi_{\boldsymbol{x}}(t)+\frac{\Delta t}{\Delta x}\left(\mathcal{F}\left(t,\boldsymbol{x}+\frac{\boldsymbol{c}_5}{2}\Delta x\right)-\mathcal{F}\left(t,\boldsymbol{x}+\frac{\boldsymbol{c}_1}{2}\Delta x\right)\right) \tag{5.5a}$$

and

$$\chi_{\boldsymbol{x}}(t+\Delta t)=\chi_{\boldsymbol{x}}\left(t+\frac{\Delta t}{2}\right)+\frac{\Delta t}{\Delta x}\left(\mathcal{F}\left(t+\frac{\Delta t}{2},\boldsymbol{x}+\frac{\boldsymbol{c}_7}{2}\Delta x\right)-\mathcal{F}\left(t+\frac{\Delta t}{2},\boldsymbol{x}+\frac{\boldsymbol{c}_3}{2}\Delta x\right)\right), \tag{5.5b}$$

where $\mathcal{F}\left(t,\boldsymbol{x}+\frac{\boldsymbol{c}_1}{2}\Delta x\right)$ represents the normalised volume flux across the eastern edge of $\mathcal{L}(\boldsymbol{x})$ computed at time t. All other normalised volume fluxes are denoted analogously.

There is a simple geometric interpretation of the volume fluxes which we will now exemplify for $\mathcal{F}\left(t,\boldsymbol{x}+\frac{\boldsymbol{c}_1}{2}\Delta x\right)$ in the case $u_1\left(t,\boldsymbol{x}+\frac{\boldsymbol{c}_1}{2}\Delta x\right)\geq 0$. Let us therefore divide the grid cell $\mathcal{L}(\boldsymbol{x})$ into two disjoint rectangles, one of area $u_1\left(t,\boldsymbol{x}+\frac{\boldsymbol{c}_1}{2}\Delta x\right)\Delta t\Delta x$ and one of area $\left(\Delta x-u_1\left(t,\boldsymbol{x}+\frac{\boldsymbol{c}_1}{2}\Delta x\right)\Delta t\right)\Delta x$, as illustrated in Figure 5.4. Then, all the fluid contained in the first rectangle will cross the eastern cell edge in the present time step while all the fluid in the latter one will fail to do so. In particular, this means that the mass flux of red fluid across the eastern cell boundary corresponds exactly to the amount of red fluid contained in the volume $u_1\left(t,\boldsymbol{x}+\frac{\boldsymbol{c}_1}{2}\Delta x\right)\Delta t\Delta x$. This, in turn, coincides with the fluid contained in the dark tetragon in the south-eastern corner of the example grid cell $\mathcal{L}(\boldsymbol{x})$ in Figure 5.4. Therefore, if $V\left(t,\boldsymbol{x}+\frac{\boldsymbol{c}_1}{2}\Delta x\right)$ denotes the volume of red fluid in $u_1\left(t,\boldsymbol{x}+\frac{\boldsymbol{c}_1}{2}\Delta x\right)\Delta t\Delta x$ the volume flux is given by

$$\mathcal{F}^{Vol}\left(t,\boldsymbol{x}+\frac{\boldsymbol{c}_1}{2}\Delta x\right)=V\left(t,\boldsymbol{x}+\frac{\boldsymbol{c}_1}{2}\Delta x\right)$$

and if $Vol_{\boldsymbol{x}}(t)$ denotes the volume of red fluid in grid cell $\mathcal{L}(\boldsymbol{x})$ at time t we have

$$Vol_{\boldsymbol{x}}\left(t+\frac{\Delta t}{2}\right)=Vol_{\boldsymbol{x}}(t)+\left(V\left(t,\boldsymbol{x}+\frac{\boldsymbol{c}_5}{2}\Delta x\right)-V\left(t,\boldsymbol{x}+\frac{\boldsymbol{c}_1}{2}\Delta x\right)\right).$$

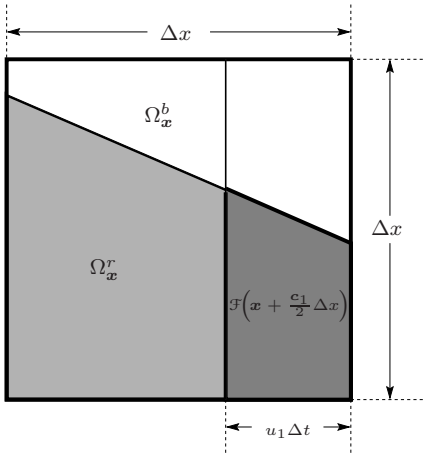


Figure 5.4: The volume flux of red fluid over the eastern cell edge in the case $u_1\left(t,\boldsymbol{x}+\frac{\boldsymbol{c}_1}{2}\Delta x\right)\geq 0$ corresponds exactly to the volume of the dark tetragon at the lower right of the example grid cell $\mathcal{L}(\boldsymbol{x})$. The full, i. e. uncoloured, volume flux over the eastern cell edge is given by the volume of the rectangle $u_1\left(t,\boldsymbol{x}+\frac{\boldsymbol{c}_1}{2}\Delta x\right)\Delta t\Delta x$ while the volume of its complement in $\mathcal{L}(\boldsymbol{x})$, $\left(\Delta x-u_1\left(t,\boldsymbol{x}+\frac{\boldsymbol{c}_1}{2}\Delta x\right)\Delta t\right)\Delta x$, corresponds the volume of (uncoloured) fluid that does not pass the eastern cell edge. Volume fractions and normalised volume fluxes are obtained by dividing the volumes of the polygons and the volume fluxes, respectively, by Δx^2.

Dividing this equation by the volume of the grid cell, Δx^2, we obtain

$$\chi_{\boldsymbol{x}}\left(t+\frac{\Delta t}{2}\right)=\chi_{\boldsymbol{x}}(t)+\frac{\Delta t}{\Delta x}\left(\frac{1}{\Delta t\Delta x}V\left(t,\boldsymbol{x}+\frac{\boldsymbol{c}_5}{2}\Delta x\right)-\frac{1}{\Delta t\Delta x}V\left(t,\boldsymbol{x}+\frac{\boldsymbol{c}_1}{2}\Delta x\right)\right)$$

and comparing this equation with (5.5a) we find that

$$\mathcal{F}\left(t,\boldsymbol{x}+\frac{\boldsymbol{c}_5}{2}\Delta x\right)=\frac{1}{\Delta t\Delta x}V\left(t,\boldsymbol{x}+\frac{\boldsymbol{c}_5}{2}\Delta x\right) \tag{5.6a}$$

and

$$\mathcal{F}\left(t,\boldsymbol{x}+\frac{\boldsymbol{c}_1}{2}\Delta x\right)=\frac{1}{\Delta t\Delta x}V\left(t,\boldsymbol{x}+\frac{\boldsymbol{c}_1}{2}\Delta x\right). \tag{5.6b}$$

In exactly the same way, we gain

$$\mathcal{F}\left(t+\frac{\Delta t}{2},\boldsymbol{x}+\frac{\boldsymbol{c}_7}{2}\Delta x\right)=\frac{1}{\Delta t\Delta x}V\left(t+\frac{\Delta t}{2},\boldsymbol{x}+\frac{\boldsymbol{c}_7}{2}\Delta x\right) \tag{5.6c}$$

and

$$\mathcal{F}\left(t+\frac{\Delta t}{2},\boldsymbol{x}+\frac{\boldsymbol{c}_3}{2}\Delta x\right)=\frac{1}{\Delta t\Delta x}V\left(t+\frac{\Delta t}{2},\boldsymbol{x}+\frac{\boldsymbol{c}_3}{2}\Delta x\right), \tag{5.6d}$$

so we can apply (5.6a) as well as (5.6b) to update $\chi_{\boldsymbol{x}}(t+\Delta t/2)$ using equation (5.5a) and then exploit (5.6c) as well as (5.6d) to update $\chi_{\boldsymbol{x}}(t)$ using equation (5.5b).

Let us now give a more classical interpretation of (5.6). Knowing from (5.1) that $\chi_{\boldsymbol{x}}(t)=Vol_{\boldsymbol{x}}(t)/\Delta x^2$ we define

$$\chi_{\boldsymbol{x}+\frac{\boldsymbol{c}_1}{2}\Delta x}(t)=\frac{V\left(t,\boldsymbol{x}+\frac{\boldsymbol{c}_1}{2}\Delta x\right)}{u_1\left(t,\boldsymbol{x}+\frac{\boldsymbol{c}_1}{2}\Delta x\right)\Delta t\Delta x},$$

i. e. we let $\chi_{\boldsymbol{x}+\frac{\boldsymbol{c}_1}{2}\Delta x}(t)$ equal the volume fraction of the red fluid in the rectangle $u_1\left(t,\boldsymbol{x}+\frac{\boldsymbol{c}_1}{2}\Delta x\right)\Delta t\Delta x$ (see again Figure 5.4), so we gain

$$u_1\left(t,\boldsymbol{x}+\frac{\boldsymbol{c}_1}{2}\Delta x\right)\chi_{\boldsymbol{x}+\frac{\boldsymbol{c}_1}{2}\Delta x}(t)=\frac{1}{\Delta t\Delta x}V\left(t,\boldsymbol{x}+\frac{\boldsymbol{c}_1}{2}\Delta x\right).$$

Since analogous results hold at each cell edge of $\mathcal{L}(\boldsymbol{x})$ and

$$\partial_1\left(u_1\chi\right)_{\boldsymbol{x}}(t)=\frac{1}{\Delta x}\left(u_1\left(t,\boldsymbol{x}+\frac{\boldsymbol{c}_1}{2}\Delta x\right)\chi_{\boldsymbol{x}+\frac{\boldsymbol{c}_1}{2}\Delta x}(t)-u_1\left(t,\boldsymbol{x}+\frac{\boldsymbol{c}_5}{2}\Delta x\right)\chi_{\boldsymbol{x}+\frac{\boldsymbol{c}_5}{2}\Delta x}(t)\right)+\mathcal{O}\left(\Delta x^2\right)$$

```
initialise χ_x(0) and u(0, x);
let t = 0;
while t < t_max
   reconstruct the interface in each grid cell as described in Section 5.1.1;
   if t mod 2 = 1
      compute F(t, x + c_1 Δx/2) in each grid cell according to (5.6b);
      compute χ_x(t + Δt/2) in each grid cell according to (5.5a);
      compute F(t + Δt/2, x + c_3 Δx/2) in each grid cell according to (5.6d);
      compute χ_x(t + Δt) in each grid cell according to (5.5b);
   else
      compute F(t, x + c_3 Δx/2) in each grid cell according to (5.6d);
      compute χ_x(t + Δt/2) in each grid cell according to (5.7a);
      compute F(t + Δt/2, x + c_1 Δx/2) in each grid cell according to (5.6b);
      compute χ_x(t + Δt) in each grid cell according to (5.7b);
   let t = t + Δt;
   if necessary, update u(t, x) with any method;
```

Algorithm 5.2: Conservative volume tracking

respectively

$$\partial_2 \left(u_2 \chi\right)_{\boldsymbol{x}} \left(t + \frac{\Delta t}{2}\right) = \frac{1}{\Delta x} \Bigg(u_2 \left(t + \frac{\Delta t}{2}, \boldsymbol{x} + \frac{\boldsymbol{c}_3}{2} \Delta x\right) \chi_{\boldsymbol{x} + \frac{\boldsymbol{c}_3}{2} \Delta x} \left(t + \frac{\Delta t}{2}\right) - u_2 \left(t + \frac{\Delta t}{2}, \boldsymbol{x} + \frac{\boldsymbol{c}_7}{2} \Delta x\right) \chi_{\boldsymbol{x} + \frac{\boldsymbol{c}_7}{2} \Delta x} \left(t + \frac{\Delta t}{2}\right) \Bigg) + \mathcal{O}\left(\Delta x^2\right),$$

where $\left(u_\alpha \chi\right)_{\boldsymbol{x}}(t) = u_\alpha(t, \boldsymbol{x}) \chi_{\boldsymbol{x}}(t)$, we find that (5.5) is in fact a proper discretisation of the transport equation (2.13) if (5.6) is true.

Note that we can dramatically improve the accuracy of scheme (5.5) by employing the so-called Strang splitting [**75**], i. e. by applying (5.5) only in uneven time steps and sweeping the directions otherwise. This means that in even time steps, we have to solve

$$\chi_{\boldsymbol{x}} \left(t + \frac{\Delta t}{2}\right) = \chi_{\boldsymbol{x}}(t) + \frac{\Delta t}{\Delta x} \left(\mathcal{F}\left(t, \boldsymbol{x} + \frac{\boldsymbol{c}_7}{2} \Delta x\right) - \mathcal{F}\left(t, \boldsymbol{x} + \frac{\boldsymbol{c}_3}{2} \Delta x\right) \right) \tag{5.7a}$$

and

$$\chi_{\boldsymbol{x}}(t + \Delta t) = \chi_{\boldsymbol{x}} \left(t + \frac{\Delta t}{2}\right) + \frac{\Delta t}{\Delta x} \left(\mathcal{F}\left(t + \frac{\Delta t}{2}, \boldsymbol{x} + \frac{\boldsymbol{c}_5}{2} \Delta x\right) - \mathcal{F}\left(t + \frac{\Delta t}{2}, \boldsymbol{x} + \frac{\boldsymbol{c}_1}{2} \Delta x\right) \right) \tag{5.7b}$$

while in uneven time steps, we stick to (5.5). The complete volume tracking procedure is summarised in Algorithm 5.2.

5.1.3. Numerical experiments with conservative volume tracking. Let us now apply Algorithm 5.2 to the benchmark problem. Figures 5.5, 5.6, and 5.7 show the results computed with $\boldsymbol{u} = \boldsymbol{u}^{(1)}$, $\boldsymbol{u} = \boldsymbol{u}^{(2)}$, respectively $\boldsymbol{u} = \boldsymbol{u}^{(3)}$. We find that for all simulations, conservative volume tracking yields very accurate results and clearly outperforms the ILBGK interface tracking Algorithm 2.9. However, looking at Table 5.1 we find that in return for its poor accuracy, ILBGK interface tracking needs less computing time than conservative volume tracking. Furthermore, this table teaches us that none of these methods suffers from considerable mass loss.

5.2. ILBGK interface tracking versus conservative volume tracking

Let us now have a closer look at the relation between the ILBGK interface tracking Algorithm 2.9 and the conservative volume tracking Algorithm 5.2. Both have in common that they use the indicator function

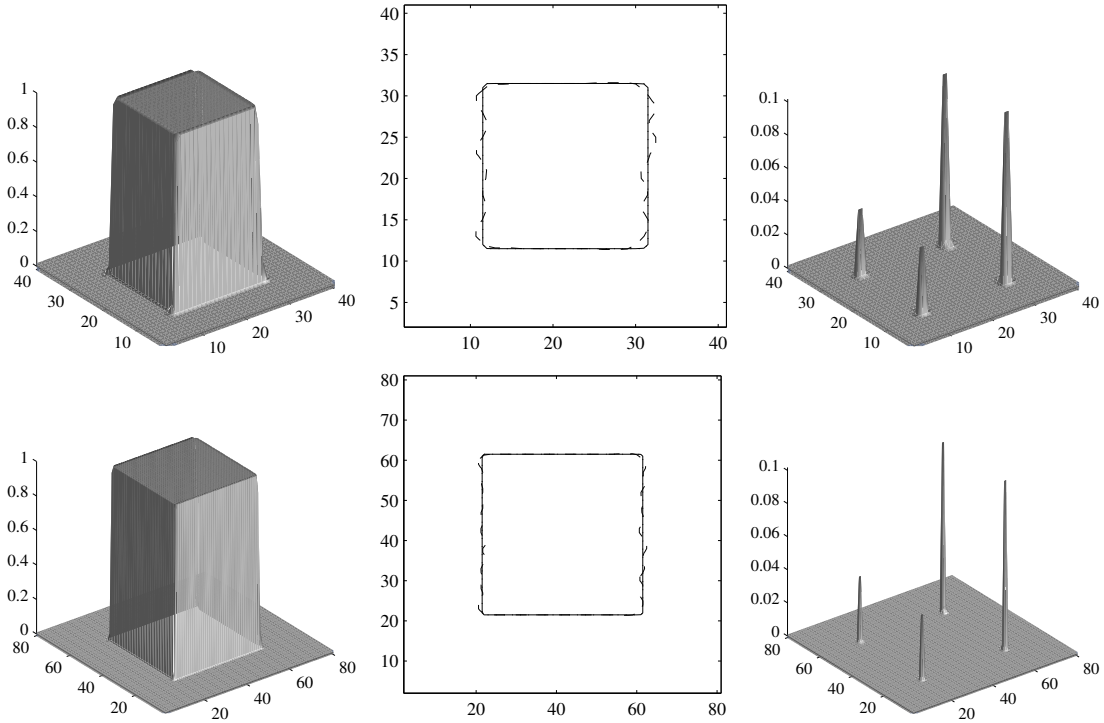


Figure 5.5: Solution of the benchmark problem for $\boldsymbol{u} = \boldsymbol{u}^{(1)}$ with Algorithm 5.2 on a 40×40 (top) and an 80×80 (bottom) lattice, all pictures taken after one complete revolution. Left: volume fraction $\chi_{\boldsymbol{x}}$ of red fluid, middle: level line for $\chi = 1/2$ respectively $\chi_{\boldsymbol{x}} = 1/2$ of the correct solution (dotted line), the ILBGK approximation (dashed line), and the volume tracking result (solid line), right: absolute value of the Euclidean error in the volume tracking result.

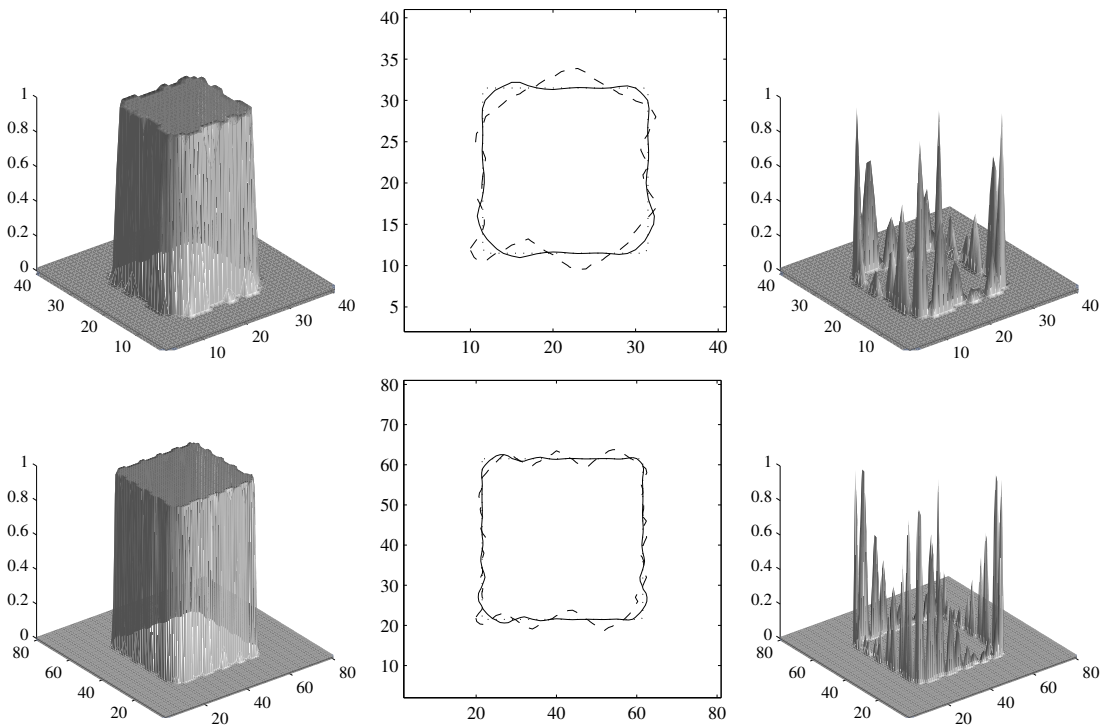


Figure 5.6: Solution of the benchmark problem for $\boldsymbol{u} = \boldsymbol{u}^{(2)}$ with Algorithm 5.2 on a 40×40 (top) and an 80×80 (bottom) lattice, all pictures taken after one complete revolution. Left: volume fraction $\chi_{\boldsymbol{x}}$ of red fluid, middle: level line for $\chi = 1/2$ respectively $\chi_{\boldsymbol{x}} = 1/2$ of the correct solution (dotted line), the ILBGK approximation (dashed line), and the volume tracking result (solid line), right: absolute value of the Euclidean error in the volume tracking result.

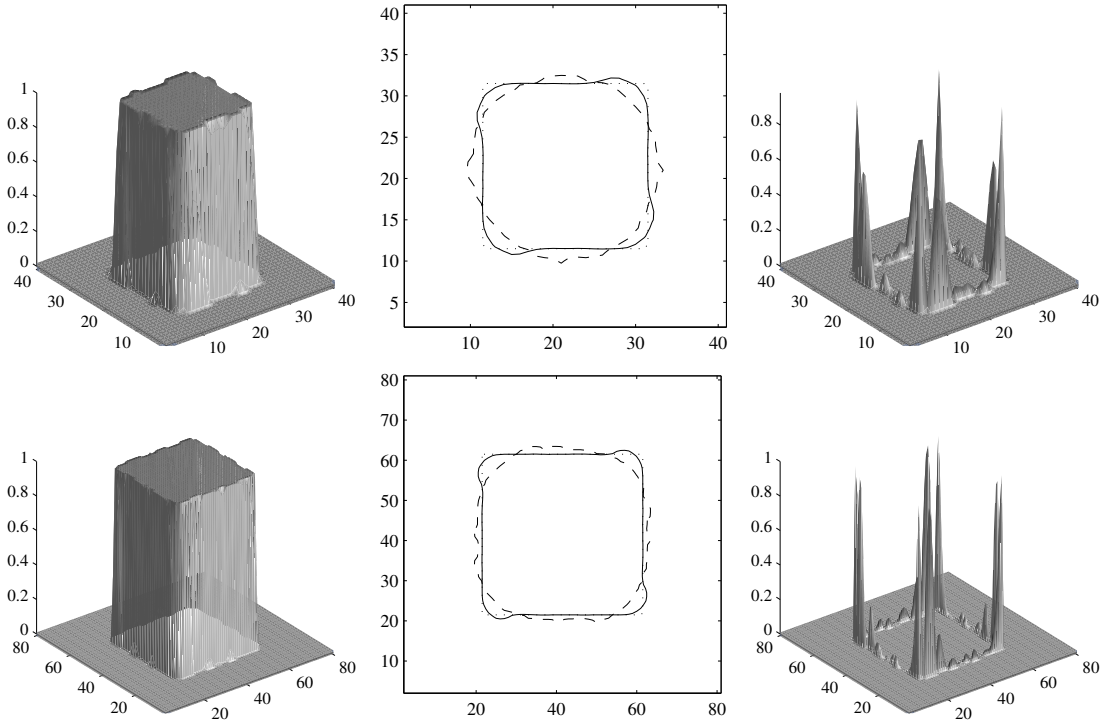


FIGURE 5.7: Solution of the benchmark problem for $\boldsymbol{u} = \boldsymbol{u}^{(3)}$ with Algorithm 5.2 on a 40×40 (top) and an 80×80 (bottom) lattice, all pictures taken after one complete revolution. Left: volume fraction $\chi_{\boldsymbol{x}}$ of red fluid, middle: level line for $\chi = 1/2$ respectively $\chi_{\boldsymbol{x}} = 1/2$ of the correct solution (dotted line), the ILBGK approximation (dashed line), and the volume tracking result (solid line), right: absolute value of the Euclidean error in the volume tracking result.

Lattice	$\boldsymbol{u}$	Mass loss (CVT)	Mass loss (ILBGK)	Flops (CVT)	Flops (ILBGK)
40×40	$\boldsymbol{u}^{(1)}$	$-5.00222 \cdot 10^{-12}$	$1.70530 \cdot 10^{-13}$	$9.29160 \cdot 10^{8}$	$2.07773 \cdot 10^{8}$
40×40	$\boldsymbol{u}^{(2)}$	$-1.42478 \cdot 10^{-10}$	$1.29319 \cdot 10^{-12}$	$2.49489 \cdot 10^{9}$	$2.07769 \cdot 10^{8}$
40×40	$\boldsymbol{u}^{(3)}$	$-1.87498 \cdot 10^{-10}$	$-3.26850 \cdot 10^{-13}$	$9.18812 \cdot 10^{9}$	$5.19433 \cdot 10^{8}$
80×80	$\boldsymbol{u}^{(1)}$	$-1.25056 \cdot 10^{-12}$	$2.13163 \cdot 10^{-13}$	$3.63342 \cdot 10^{9}$	$1.59739 \cdot 10^{9}$
80×80	$\boldsymbol{u}^{(2)}$	$-6.43610 \cdot 10^{-11}$	$1.90425 \cdot 10^{-12}$	$9.76144 \cdot 10^{9}$	$1.59743 \cdot 10^{9}$
80×80	$\boldsymbol{u}^{(3)}$	$-1.70616 \cdot 10^{-10}$	$-9.66338 \cdot 10^{-13}$	$3.83166 \cdot 10^{10}$	$3.99364 \cdot 10^{9}$

TABLE 5.1: Percentage loss of red mass and computing times for the simulations with the conservative volume tracking Algorithm 5.2 (CVT) and the ILBGK interface tracking Algorithm 2.9 (ILBGK)

χ of red fluid to determine the position of the interface Γ but they use very different approaches to solve the transport equation (2.13).

ILBGK interface tracking determines the local colour gradient $\boldsymbol{F}$, which is an inner normal vector of Γ, and distributes red pseudo-mass, i. e. χ, into q different directions (for example $q = 9$ in the D2Q9 model used here) such that as much red pseudo-mass as possible is distributed into the direction closest to $\boldsymbol{F}$. If there is some red pseudo-mass left the direction which is second closest to $\boldsymbol{F}$ obtains as much as possible, and so on until there is no more red pseudo-mass left. This procedure results, in fact, in a perfectly sharp interface, never more than two grid cells wide, but the run of the interface is not kept under control. Actually, this is not very surprising since the whole procedure is designed to keep the interface sharp but no care at all is taken about its correct shape.

In contrast to ILBGK interface tracking, the conservative volume tracking Algorithm 5.2 is more sophisticated. It calculates the inner unit normal vector $\boldsymbol{n}$ of Γ according to Proposition 2.5 and then utilises $\boldsymbol{n}_{\boldsymbol{x}}$ and $\chi_{\boldsymbol{x}}$ to find a reasonable approximation to Γ in each grid cell. After this, the volume tracking method retraces in some sense the characteristics of the transport equation (2.13) to find the full flux of fluid volume over the cell boundaries. Since in each grid cell, the position of Γ is known the volume tracking Algorithm can now determine the fluxes of red fluid over the cell boundaries. The

```
initialise ρ(0,x), χ_x(0), and u(0,x);
let f(0,x;v) = f_S^eq(ρ(0,x), u(0,x)) as defined by (2.43);
while t < t_max
   determine f_S^eq(ρ(t,x), u(t,x)) according to (2.43);
   do uncoloured collision according to (2.44);
   do uncoloured propagation according to (2.45);
    determine χ_x(t + Δt) according to Algorithm 5.2;
    let t = t + Δt;
   compute ρ(t,x) and ρu_α(t,x) according to (2.34);
   let u_α(t,x) = ρu_α(t,x)/ρ(t,x);
```

Algorithm 5.3: Volume tracking lattice BGK

change in $\chi_{\boldsymbol{x}}$ is then given by the sum of the normalised fluxes of red volume over all boundaries of the grid cell. Altogether, we see that volume tracking provides a very detailed treatment of all aspects of interface motion by carefully determining the amount of red fluid to be moved to a given direction. This guarantees not only the sharpness of the interface but also its correct motion.

Comparing both methods we find that ILBGK is in some sense a crude version of conservative volume tracking. This interpretation coincides with the results presented in Section 5.1, where we found that ILBGK interface tracking is faster but less accurate than conservative volume tracking.

5.3. Combination of conservative volume tracking and lattice BGK

Our goal is now to design a lattice BGK method for two-phase flow which does not suffer from too much disturbance of the interface. For this reason, we construct a volume tracking lattice BGK (VTLBGK) method by combining the volume tracking Algorithm 5.2 with the Navier-Stokes solver of the immiscible lattice BGK Algorithm 2.6. The whole procedure is summarised in Algorithm 5.3.

Let us now apply this Algorithm to the benchmark problem. In analogy to Section 3.4, we assume $\varrho = 1$ and $\omega = 1$, which yields the dynamic viscosity $\mu = 1/6$, and we run simulations for $\sigma \in \{0.001, 0.01, 0.1\}$. Results for $\sigma = 0.001$ are shown in Figures 5.8 to 5.10, those for $\sigma = 0.01$ are collected in Figures 5.11 to 5.13, and those for $\sigma = 0.1$ can finally be found in Figures 5.14 to 5.16. Note that we do not know the analytical solution for any of those experiments . Mass loss and computing times for all simulations are collected in Table 5.2.

In analogy to the results presented in Section 5.1.3, we find that for $\sigma = 0.001$ VTLBGK yields more accurate results than ILBGK. For $\sigma = 0.1$ surface tension turns the square into a smooth circle in the VTLBGK simulations and into a perturbed circle in the ILBGK computations, i. e, surface tension is already quite strong but still too weak to cure the spurious oscillations produced by ILBGK interface tracking. In the simulations for $\sigma = 0.1$, both methods produce a smooth circle from which we conclude that surface tension does no longer allow any other shape of the interface. Note that computing time in VTLBGK simulations is always higher than in ILBGK calculations.

The simulations for $\boldsymbol{u}(0, \boldsymbol{x}) = \boldsymbol{u}^{(3)}$ produce the kind of irregularities we already observed in Sections 3.2 and 3.4. Still, there is no rigorous explanation for this phenomenon. Note that this time, we also observe perturbations of $\chi_{\boldsymbol{x}}$ in the results for $\boldsymbol{u}(0, \boldsymbol{x}) = \boldsymbol{u}^{(1)}$ but their order of magnitude is much smaller. Also the rigorous investigation of their origin is a task for future research. One possible explanation for the irregularities in $\chi_{\boldsymbol{x}}$ might be the incompressibility error in the velocity field causing perturbations of $\chi_{\boldsymbol{x}}$ which are then amplified in later time steps. However, this is pure speculation. Fortunately, mass loss is only of $\mathcal{O}\left(10^{-10}\right)$ for all VTLBGK computations.

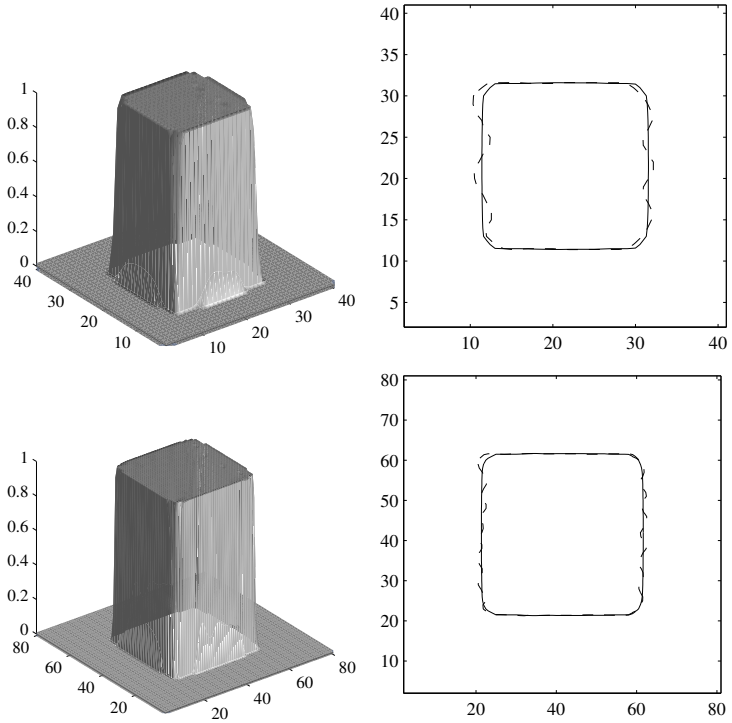


Figure 5.8: Solution of the benchmark problem for $\boldsymbol{u}(0,\boldsymbol{x}) = \boldsymbol{u}^{(1)}$ and $\sigma = 0.001$ using Algorithm 5.3 on a 40×40 (top line) and an 80×80 (bottom line) lattice, all pictures taken after one revolution. Left: the volume fraction $\chi_{\boldsymbol{x}}$, right: level line for $\chi = 1/2$ respectively $\chi_{\boldsymbol{x}} = 1/2$ of the ILBGK approximation (dashed line) and the VTLBGK result (solid line).

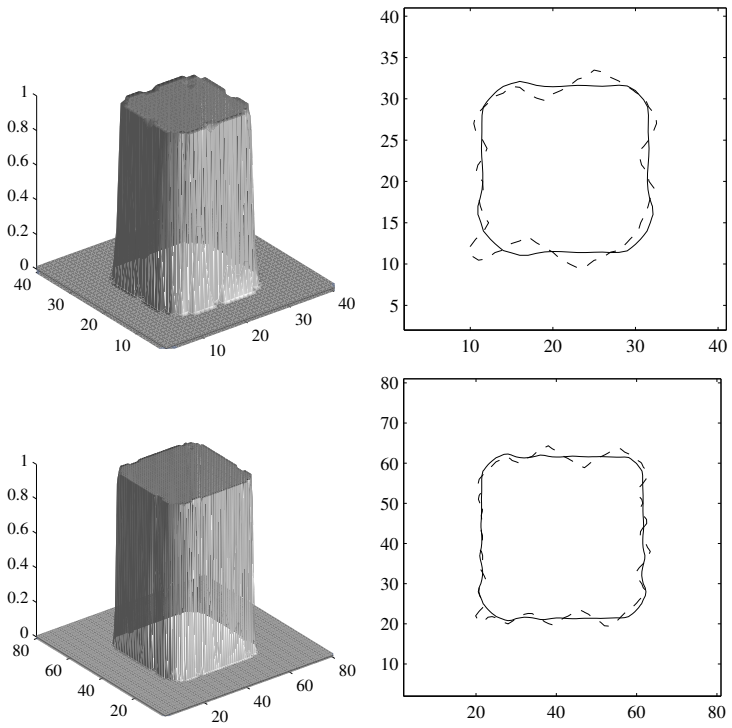


Figure 5.9: Solution of the benchmark problem for $\boldsymbol{u}(0,\boldsymbol{x}) = \boldsymbol{u}^{(2)}$ and $\sigma = 0.001$ using Algorithm 5.3 on a 40×40 (top line) and an 80×80 (bottom line) lattice, all pictures taken after one revolution. Left: the volume fraction $\chi_{\boldsymbol{x}}$, right: level line for $\chi = 1/2$ respectively $\chi_{\boldsymbol{x}} = 1/2$ of the ILBGK approximation (dashed line) and the VTLBGK result (solid line).

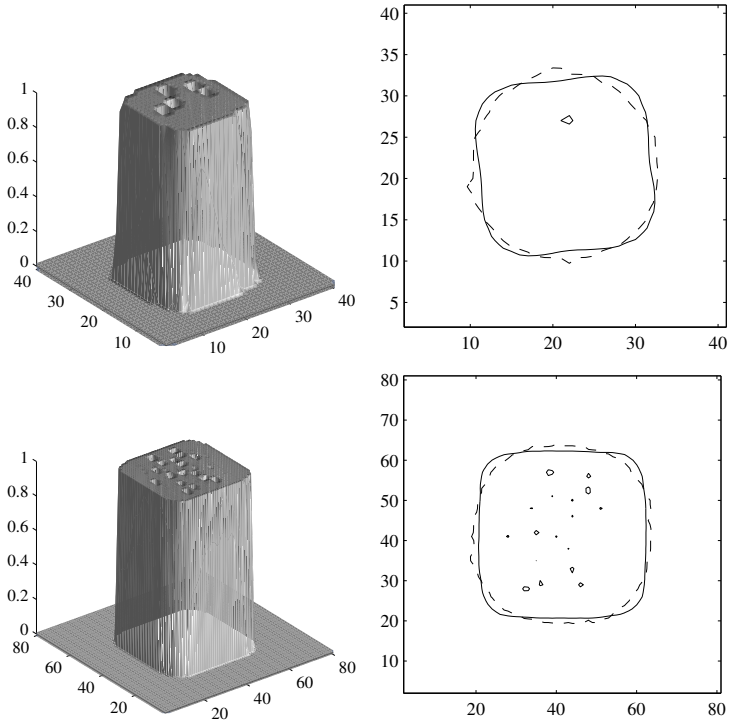


Figure 5.10: Solution of the benchmark problem for $\boldsymbol{u}(0,\boldsymbol{x}) = \boldsymbol{u}^{(3)}$ and $\sigma = 0.001$ using Algorithm 5.3 on a 40×40 (top line) and an 80×80 (bottom line) lattice, all pictures taken after one revolution. Left: the volume fraction $\chi_{\boldsymbol{x}}$, right: level line for $\chi = 1/2$ respectively $\chi_{\boldsymbol{x}} = 1/2$ of the ILBGK approximation (dashed line) and the VTLBGK result (solid line).

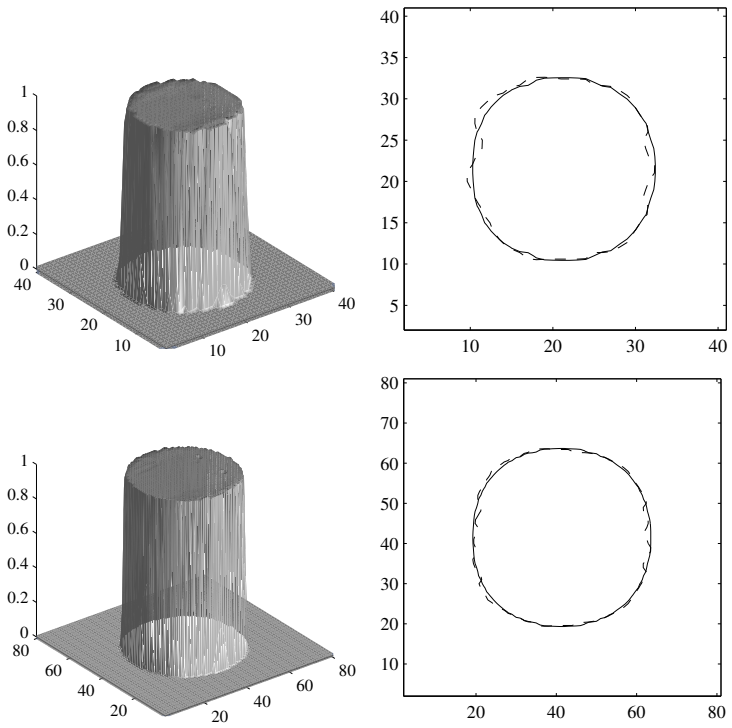


Figure 5.11: Solution of the benchmark problem for $\boldsymbol{u}(0,\boldsymbol{x}) = \boldsymbol{u}^{(1)}$ and $\sigma = 0.01$ using Algorithm 5.3 on a 40×40 (top line) and an 80×80 (bottom line) lattice, all pictures taken after one revolution. Left: the volume fraction $\chi_{\boldsymbol{x}}$, right: level line for $\chi = 1/2$ respectively $\chi_{\boldsymbol{x}} = 1/2$ of the ILBGK approximation (dashed line) and the VTLBGK result (solid line).

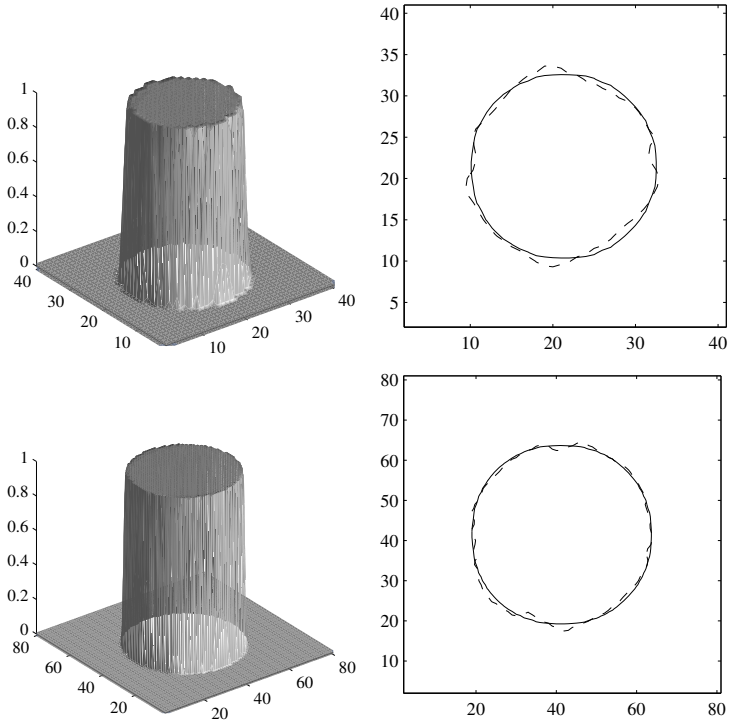


Figure 5.12: Solution of the benchmark problem for $\boldsymbol{u}(0,\boldsymbol{x}) = \boldsymbol{u}^{(2)}$ and $\sigma = 0.01$ using Algorithm 5.3 on a 40×40 (top line) and an 80×80 (bottom line) lattice, all pictures taken after one revolution. Left: the volume fraction $\chi_{\boldsymbol{x}}$, right: level line for $\chi = 1/2$ respectively $\chi_{\boldsymbol{x}} = 1/2$ of the ILBGK approximation (dashed line) and the VTLBGK result (solid line).

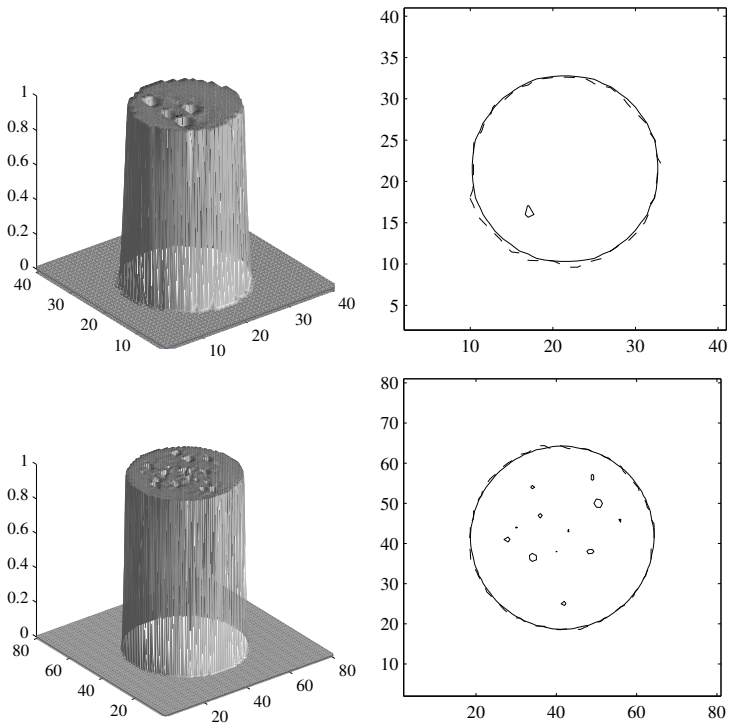


Figure 5.13: Solution of the benchmark problem for $\boldsymbol{u}(0,\boldsymbol{x}) = \boldsymbol{u}^{(3)}$ and $\sigma = 0.01$ using Algorithm 5.3 on a 40×40 (top line) and an 80×80 (bottom line) lattice, all pictures taken after one revolution. Left: the volume fraction $\chi_{\boldsymbol{x}}$, right: level line for $\chi = 1/2$ respectively $\chi_{\boldsymbol{x}} = 1/2$ of the ILBGK approximation (dashed line) and the VTLBGK result (solid line).

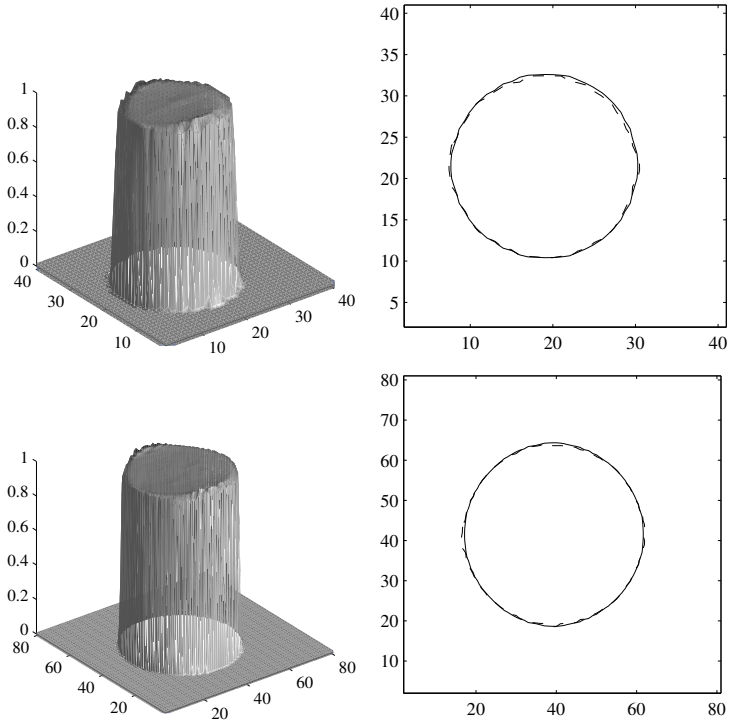


Figure 5.14: Solution of the benchmark problem for $\boldsymbol{u}(0,\boldsymbol{x}) = \boldsymbol{u}^{(1)}$ and $\sigma = 0.1$ using Algorithm 5.3 on a 40×40 (top line) and an 80×80 (bottom line) lattice, all pictures taken after one revolution. Left: the volume fraction $\chi_{\boldsymbol{x}}$, right: level line for $\chi = 1/2$ respectively $\chi_{\boldsymbol{x}} = 1/2$ of the ILBGK approximation (dashed line) and the VTLBGK result (solid line).

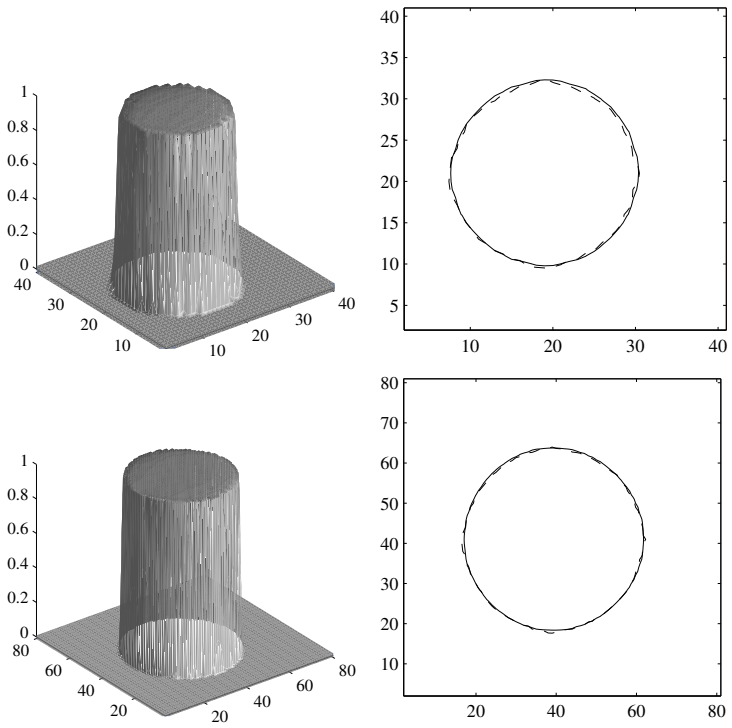


Figure 5.15: Solution of the benchmark problem for $\boldsymbol{u}(0,\boldsymbol{x}) = \boldsymbol{u}^{(2)}$ and $\sigma = 0.1$ using Algorithm 5.3 on a 40×40 (top line) and an 80×80 (bottom line) lattice, all pictures taken after one revolution. Left: the volume fraction $\chi_{\boldsymbol{x}}$, right: level line for $\chi = 1/2$ respectively $\chi_{\boldsymbol{x}} = 1/2$ of the ILBGK approximation (dashed line) and the VTLBGK result (solid line).

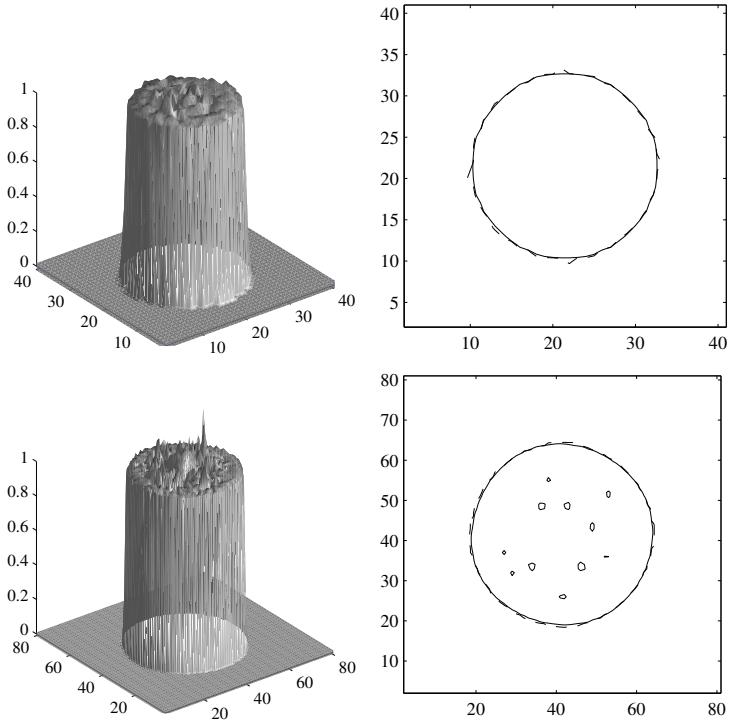


Figure 5.16: Solution of the benchmark problem for $\boldsymbol{u}(0,\boldsymbol{x}) = \boldsymbol{u}^{(3)}$ and $\sigma = 0.1$ using Algorithm 5.3 on a 40×40 (top line) and an 80×80 (bottom line) lattice, all pictures taken after one revolution. Left: the volume fraction $\chi_{\boldsymbol{x}}$, right: level line for $\chi = 1/2$ respectively $\chi_{\boldsymbol{x}} = 1/2$ of the ILBGK approximation (dashed line) and the VTLBGK result (solid line).

Lattice	$\boldsymbol{u}(0,\boldsymbol{x})$	σ	Mass loss in %	Number of flops
40×40	$\boldsymbol{u}^{(1)}$	0.001	$-5.75014 \cdot 10^{-10}$	$7.88354 \cdot 10^{9}$
40×40	$\boldsymbol{u}^{(1)}$	0.01	$2.80039 \cdot 10^{-10}$	$6.44152 \cdot 10^{9}$
40×40	$\boldsymbol{u}^{(1)}$	0.1	$-2.80011 \cdot 10^{-10}$	$6.72455 \cdot 10^{9}$
40×40	$\boldsymbol{u}^{(2)}$	0.001	$1.72520 \cdot 10^{-10}$	$5.28694 \cdot 10^{9}$
40×40	$\boldsymbol{u}^{(2)}$	0.01	$-1.15008 \cdot 10^{-10}$	$4.64590 \cdot 10^{9}$
40×40	$\boldsymbol{u}^{(2)}$	0.1	$-1.22498 \cdot 10^{-10}$	$5.34097 \cdot 10^{9}$
40×40	$\boldsymbol{u}^{(3)}$	0.001	$-8.77520 \cdot 10^{-10}$	$2.91615 \cdot 10^{10}$
40×40	$\boldsymbol{u}^{(3)}$	0.01	$3.34978 \cdot 10^{-10}$	$2.86207 \cdot 10^{10}$
40×40	$\boldsymbol{u}^{(3)}$	0.1	$-9.22469 \cdot 10^{-10}$	$2.27901 \cdot 10^{10}$
80×80	$\boldsymbol{u}^{(1)}$	0.001	$1.48745 \cdot 10^{-10}$	$6.73781 \cdot 10^{10}$
80×80	$\boldsymbol{u}^{(1)}$	0.01	$2.90640 \cdot 10^{-10}$	$6.13699 \cdot 10^{10}$
80×80	$\boldsymbol{u}^{(1)}$	0.1	$-2.25015 \cdot 10^{-10}$	$6.34040 \cdot 10^{10}$
80×80	$\boldsymbol{u}^{(2)}$	0.001	$-3.09370 \cdot 10^{-10}$	$2.72286 \cdot 10^{10}$
80×80	$\boldsymbol{u}^{(2)}$	0.01	$1.88123 \cdot 10^{-10}$	$2.34716 \cdot 10^{10}$
80×80	$\boldsymbol{u}^{(2)}$	0.1	$-5.81366 \cdot 10^{-10}$	$2.73493 \cdot 10^{10}$
80×80	$\boldsymbol{u}^{(3)}$	0.001	$2.23096 \cdot 10^{-10}$	$2.85516 \cdot 10^{11}$
80×80	$\boldsymbol{u}^{(3)}$	0.01	$-3.22501 \cdot 10^{-10}$	$2.53958 \cdot 10^{11}$
80×80	$\boldsymbol{u}^{(3)}$	0.1	$-2.03130 \cdot 10^{-10}$	$2.12220 \cdot 10^{11}$

Table 5.2: Percentage loss of red mass and computing times for the simulations with Algorithm 5.3

Conclusions

Assuming the absence of jumps in density and stress at the interface we have studied the immiscible lattice BGK model of incompressible two-phase flow. Even though many parts of the analysis have been performed for the particular D2Q9 model, the underlying ideas apply directly to other lattices structures because they are based only on general properties of lattice BGK. Adopting those ideas to more general immiscible lattice Boltzmann models is also straightforward.

By detailed mathematical analysis, we have found a recently published [**22, 23**] variant of ILBGK to be more reasonable than the original. Furthermore, we have presented a model of surface tension for which immiscible lattice BGK is formally consistent to the full-space form of the incompressible Navier-Stokes equations. The proof of this consistency has turned out to be not at all straightforward.

Particular focus was given on the intrinsic interface tracking method of immiscible lattice BGK. This method allows a sharp interface to smear out during one time step and corrects the smearing during the next one. This correction is done by the recolouring step. In one space dimension, this method works very well but in 2-d, the recolouring step produces severe perturbation in the interface. Furthermore, for a special velocity field we have observed irregularities in the solution which we have not yet been able to rigorously explain. However, it should be mentioned that ILBGK interface tracking performs definitely better than a Riemann solver because Riemann solvers are not able to maintain a sharp interface in more than one space dimension.

We have tried two approaches to cure the perturbation of the interface, anisotropic diffusion and artificial surface tension. Using anisotropic diffusion we have managed to remove the disturbance of the interface but the price we had to pay for this was unacceptable mass loss. The artificial surface tension approach has turned out to be of no help for removing the perturbation of the interface but produced unacceptable mass loss as well. With artificial surface tension, we have again observed the above irregularities in the solution for a special velocity field.

Furthermore, we have found that ILBGK interface tracking is only a special case of a more general method which first solves a parameter-dependent viscous transport equation for the indicator function of one phase and then applies the recolouring step to correct the resulting diffusion of the phases into each other. Detailed analysis of the stability properties of the general viscous transport scheme has yielded a numerical method with improved theoretical properties but even for this improved scheme, the recolouring step has turned out to produce severe disturbance of the interface.

As an alternative to the unreliable ILBGK interface tracking, we have presented a relatively simple conservative volume tracking method. In numerical experiments, this method has turned out to be more accurate but also more expensive than ILBGK interface tracking. Furthermore, we have realized a close connection between the concepts of ILBGK interface tracking and conservative volume tracking. Coupling conservative volume tracking with the Navier-Stokes solver of ILBGK we have found the numerical results computed with this coupled method to be quite promising, even though again, we have observed irregularities in the solution for a special velocity field.

Once the reason why those irregularities appear is rigorously explored, it may be possible to use this knowledge for the improvement of both the (general) ILBGK interface tracking scheme and the coupled volume tracking lattice BGK method. Also the reason why mass loss appears if anisotropic diffusion is used to remove the disturbance of the interface is not yet investigated, so it might also be possible to cure this defect and to construct a reliable transport scheme from ILBGK interface tracking in this way.

In addition, it should also be examined in how far the difficulties of ILBGK interface tracking can be reduced by allowing the interface to smear out over a few lattice points, as already proposed by some authors. Furthermore, also the other lattice Boltzmann models of two-phase flow (interacting potential and free energy) have not yet been mathematically investigated. Finally, to investigate the consequences of jumps in density and stress is another challenging task for future research.

APPENDIX

Example of a smooth indicator function

Let us now consider an example of an indicator function $\chi_{\Delta x} \in \mathcal{C}^1\left([0, t_{max}] \times \Omega, \mathbb{R}\right)$ fulfilling

$$\chi_{\Delta x}(t, \boldsymbol{x}) = \begin{cases} 1 & \text{for } \boldsymbol{x} \in \Omega^r \setminus \Gamma_{\Delta x} \\ \eta_{\Delta x}\left(t, \boldsymbol{z}^k, \ell\right) & \text{for } \boldsymbol{x} \in \Gamma_{\Delta x} \cap \mathcal{U}^k \\ 0 & \text{for } \boldsymbol{x} \in \Omega^b \setminus \Gamma_{\Delta x} \end{cases} \tag{A.1a}$$

as well as

$$\eta_{\Delta x} \in \mathcal{C}^1\left([0, t_{max}] \times \Theta^k \times (-1, 1), \mathbb{R}\right), \qquad \partial_\ell \eta_{\Delta x} \geq 0, \qquad \lim_{\Delta x \to 0} \eta_{\Delta x}\left(t, \boldsymbol{z}^k, \ell\right) = \eta_0\left(t, \boldsymbol{z}^k, \ell\right) \tag{A.1b}$$

$$\eta_0 \in \mathcal{C}^1\left([0, t_{max}] \times \Theta^k \times (-1, 1), \mathbb{R}\right), \qquad \text{and} \qquad \partial_\ell \eta_0 \geq 0. \tag{A.1c}$$

in a d-dimensional domain Ω (compare Section 2.4.1). Note that due to $\lim_{\Delta x \to 0} \Gamma_{\Delta x} = \Gamma$, condition (2.70a) immediately implies

$$\lim_{\Delta x \to 0} \chi_{\Delta x} = \begin{cases} 1 & \text{for } \boldsymbol{x} \in \Omega^r \\ 0 & \text{for } \boldsymbol{x} \in \Omega^b. \end{cases}$$

First, we introduce the so-called smoothing kernel

$$\Psi_{\Delta x}(\boldsymbol{x}) = \frac{1}{\Delta x^d} \Psi_0\left(\frac{\boldsymbol{x}}{\Delta x}\right) \tag{A.2a}$$

with [**13**]

$$\Psi_0(\boldsymbol{x}) = \begin{cases} \frac{\exp\left(-1/\left(1 - \|\boldsymbol{x}\|_2^2\right)\right)}{\int_\Omega \exp\left(-1/\left(1 - \|\boldsymbol{y}\|_2^2\right)\right) d\boldsymbol{y}} & \text{for } \|\boldsymbol{x}\|_2 < 1 \\ 0 & \text{otherwise,} \end{cases} \tag{A.2b}$$

see Figure A.1 for an exemplifying illustration. Then, for any regular distribution $K \in \mathcal{D}'(\Omega)$ we define $K_{\Delta x}$ by

$$K_{\Delta x}(\boldsymbol{x}) = \left(\Psi_{\Delta x} * K\right)(\boldsymbol{x}),$$

where

$$\left(\Psi_{\Delta x} * K\right)(\boldsymbol{x}) = \int_\Omega \Psi_{\Delta x}(\boldsymbol{x} - \boldsymbol{y}) K(\boldsymbol{y})\, d\boldsymbol{y} \tag{A.3}$$

denotes the convolution of K with $\Psi_{\Delta x}$, compare Figure A.2.

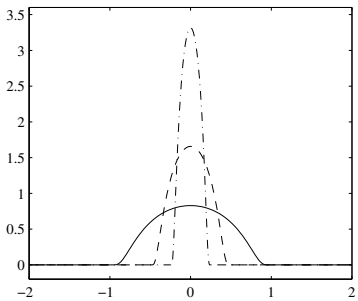


FIGURE A.1: The smoothing kernels Ψ_0 (solid), $\Psi_{1/2}$ (dashed), and $\Psi_{1/4}$ (dash-dotted) in one-dimensional space

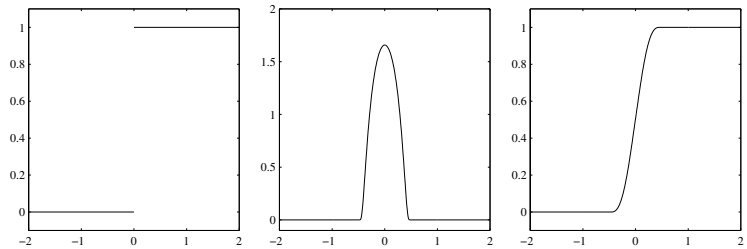


FIGURE A.2: The function χ (left), the smoothing kernel $\Psi_{1/2}$ (middle), and the convolution product $\Psi_{1/2} * \chi$ (right) in one-dimensional space

Later in this Appendix, we will show $\chi_{\Delta x} = \Psi_{\Delta x} * \chi$, where

$$\chi(t, \boldsymbol{x}) = \begin{cases} 1 & \text{for } \boldsymbol{x} \in \Omega^r(t) \\ 0 & \text{for } \boldsymbol{x} \in \Omega^b(t), \end{cases}$$

to fulfill (A.1). First, however, we want to show one basic property of convolution with $\Psi_{\Delta x}$, namely

$$\langle K_{\Delta x}, \varphi \rangle_\Omega = \langle K, \varphi \rangle_\Omega + \mathcal{O}\left(\Delta x^2\right)$$

for each regular distribution $K \in \mathcal{D}'(\Omega)$ and any test function $\varphi \in \mathcal{D}(\Omega)$. By definition, we have

$$|\langle K_{\Delta x} - K, \varphi \rangle_\Omega| = \left| \int_\Omega \int_\Omega \Psi_{\Delta x}(\boldsymbol{x} - \boldsymbol{y}) K(\boldsymbol{y}) \varphi(\boldsymbol{x}) \, d\boldsymbol{x} \, d\boldsymbol{y} - \int_\Omega K(\boldsymbol{x}) \varphi(\boldsymbol{x}) \, d\boldsymbol{x} \right| \tag{A.4}$$

and from (A.2) and the coordinate transformation $(\boldsymbol{x} - \boldsymbol{y})/\Delta x \mapsto \tilde{\boldsymbol{x}}$ we obtain

$$\int_\Omega \int_\Omega \Psi_{\Delta x}(\boldsymbol{x} - \boldsymbol{y}) K(\boldsymbol{y}) \varphi(\boldsymbol{x}) \, d\boldsymbol{x} \, d\boldsymbol{y} = \int_\Omega \int_{\mathcal{B}_1(\boldsymbol{0})} \Psi_0(\tilde{\boldsymbol{x}}) K(\boldsymbol{y}) \varphi(\boldsymbol{y} + \tilde{\boldsymbol{x}} \Delta x) \, d\tilde{\boldsymbol{x}} \, d\boldsymbol{y}, \tag{A.5}$$

where $\mathcal{B}_1(\boldsymbol{0}) = \left\{ \tilde{\boldsymbol{x}} \in \mathbb{R}^d \, : \, \|\tilde{\boldsymbol{x}}\|_2 < 1 \right\}$. By Taylor expansion, we find that

$$\varphi\left(\boldsymbol{y} + \tilde{\boldsymbol{x}} \Delta x\right) = \varphi(\boldsymbol{y}) + \Delta x \tilde{x}_\alpha \partial_\alpha \varphi(\boldsymbol{y}) + \mathcal{O}\left(\Delta x^2\right)$$

and therefore,

$$\begin{aligned} &\int_\Omega \int_{\mathcal{B}_1(\boldsymbol{0})} \Psi_0(\tilde{\boldsymbol{x}}) K(\boldsymbol{y}) \varphi(\boldsymbol{y} + \tilde{\boldsymbol{x}} \Delta x) \, d\tilde{\boldsymbol{x}} \, d\boldsymbol{y} \\ &\qquad = \int_{\mathcal{B}_1(\boldsymbol{0})} \Psi_0(\tilde{\boldsymbol{x}}) \, d\tilde{\boldsymbol{x}} \int_\Omega K(\boldsymbol{y}) \varphi(\boldsymbol{y}) \, d\boldsymbol{y} + \Delta x \int_\Omega \int_{\mathcal{B}_1(\boldsymbol{0})} \Psi_0(\tilde{\boldsymbol{x}}) K(\boldsymbol{y}) z_\alpha \partial_\alpha \varphi(\boldsymbol{y}) \, d\tilde{\boldsymbol{x}} \, d\boldsymbol{y} + \mathcal{O}\left(\Delta x^2\right). \end{aligned}$$

From (A.2b), we learn that

$$\int_{\mathcal{B}_1(\boldsymbol{0})} \Psi_0(\tilde{\boldsymbol{x}}) \, d\tilde{\boldsymbol{x}} = 1,$$

so (A.5) takes the form

$$\begin{aligned} &\int_\Omega \int_\Omega \Psi_{\Delta x}(\boldsymbol{x} - \boldsymbol{y}) K(\boldsymbol{y}) \varphi(\boldsymbol{x}) \, d\boldsymbol{x} \, d\boldsymbol{y} \\ &\qquad = \int_\Omega K(\boldsymbol{y}) \varphi(\boldsymbol{y}) \, d\boldsymbol{y} + \Delta x \int_\Omega \int_{\mathcal{B}_1(\boldsymbol{0})} \Psi_0(\tilde{\boldsymbol{x}}) K(\boldsymbol{y}) \tilde{x}_\alpha \partial_\alpha \varphi(\boldsymbol{y}) \, d\tilde{\boldsymbol{x}} \, d\boldsymbol{y} + \mathcal{O}\left(\Delta x^2\right) \end{aligned}$$

and inserting this into (A.4) we obtain

$$|\langle K_{\Delta x} - K, \varphi \rangle_\Omega| = \left| \Delta x \int_{\mathcal{B}_1(\boldsymbol{0})} \Psi_0(\tilde{\boldsymbol{x}}) \tilde{x}_\alpha \, d\tilde{\boldsymbol{x}} \int_\Omega K(\boldsymbol{y}) \partial_\alpha \varphi(\boldsymbol{y}) \, d\boldsymbol{y} + \mathcal{O}\left(\Delta x^2\right) \right|$$

which already yields

$$\langle K_{\Delta x}, \varphi \rangle_\Omega = \langle K, \varphi \rangle_\Omega + \mathcal{O}\left(\Delta x\right).$$

Note, however, that the smoothing kernel Ψ_0 is even and the function $\tilde{x}_\alpha$ is odd, so we have

$$\int_{\mathcal{B}_1(\boldsymbol{0})} \Psi_0(\tilde{\boldsymbol{x}}) \tilde{x}_\alpha \, d\tilde{\boldsymbol{x}} = 0$$

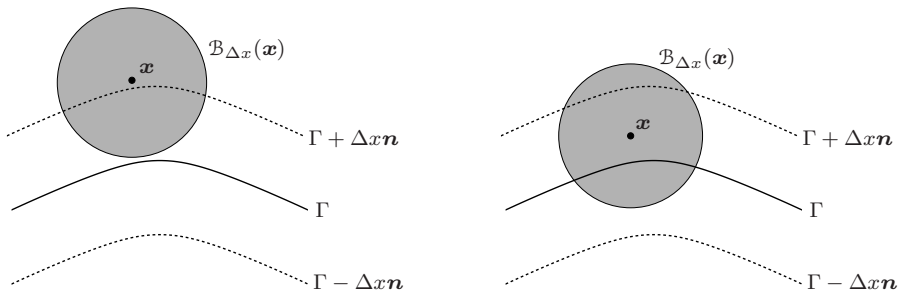


Figure A.3: $\mathcal{B}_{\Delta x}(\boldsymbol{x})$ for $\boldsymbol{x} \in \Omega \setminus \Gamma_{\Delta x}$ (left) and for $\boldsymbol{x} \in \Gamma_{\Delta x}$ (right)

and therefore,

$$\langle K_{\Delta x}, \varphi \rangle_\Omega = \langle K, \varphi \rangle_\Omega + \mathcal{O}\left(\Delta x^2\right)$$

in this special case. Altogether, we gain

Proposition A.1. *For any regular distribution $K \in \mathcal{D}'(\Omega)$ and an arbitrary test function $\varphi \in \mathcal{D}(\Omega)$*

$$\langle \Psi_{\Delta x} * K, \varphi \rangle_\Omega = \langle K, \varphi \rangle_\Omega + \mathcal{O}\left(\Delta x^2\right),$$

where the smoothing kernel $\Psi_{\Delta x}$ is given by (A.2). *If Ψ_0 was not an even function only*

$$\langle \Psi_{\Delta x} * K, \varphi \rangle_\Omega = \langle K, \varphi \rangle_\Omega + \mathcal{O}\left(\Delta x\right)$$

was fulfilled.

In the remainder of this Appendix, we want to show that $\Psi_{\Delta x} * \chi$ also fulfils conditions (A.1a) to (A.1c). Note that by construction, $\Psi_{\Delta x} * K \in \mathcal{C}^\infty(\Omega, \mathbb{R})$ for any regular distribution $K \in \mathcal{D}'(\Omega)$ [**13**] and therefore, we have $\Psi_{\Delta x} * \chi \in \mathcal{C}^1\left([0, t_{max}] \times \Omega, \mathbb{R}\right)$. Since $\Psi_{\Delta x}$ is even, i. e.

$$\Psi_{\Delta x}\left(\boldsymbol{x} - \boldsymbol{y}\right) = \Psi_{\Delta x}\left(\boldsymbol{y} - \boldsymbol{x}\right),$$

and $\operatorname{supp} \Psi_{\Delta x} = \mathcal{B}_{\Delta x}(\boldsymbol{0})$, where $\mathcal{B}_{\Delta x}(\boldsymbol{x}) = \{\boldsymbol{y} : \|\boldsymbol{y} - \boldsymbol{x}\|_2 < \Delta x\}$, we obtain from (A.3)

$$\Psi_{\Delta x} * \chi(t, \boldsymbol{x}) = \int_{\mathcal{B}_{\Delta x}(\boldsymbol{x})} \Psi_{\Delta x}\left(\boldsymbol{y} - \boldsymbol{x}\right) \chi(t, \boldsymbol{y})\, d\boldsymbol{y}.$$

For $\boldsymbol{x} \in \Omega \setminus \Gamma_{\Delta x}$, we have $\mathcal{B}_{\Delta x}(\boldsymbol{x}) \cap \Gamma = \emptyset$ and therefore, $\chi(\boldsymbol{y}) = \chi(\boldsymbol{x})$ for each $\boldsymbol{y} \in \mathcal{B}_{\Delta x}(\boldsymbol{x})$, see Figure A.3 for a sketch. Thus,

$$\int_{\mathcal{B}_{\Delta x}(\boldsymbol{x})} \Psi_{\Delta x}\left(\boldsymbol{y} - \boldsymbol{x}\right) \chi(t, \boldsymbol{y})\, d\boldsymbol{y} = 0 \qquad \text{for } \boldsymbol{x} \in \Omega^b(t) \setminus \Gamma_{\Delta x}(t)$$

and

$$\int_{\mathcal{B}_{\Delta x}(\boldsymbol{x})} \Psi_{\Delta x}\left(\boldsymbol{y} - \boldsymbol{x}\right) \chi(t, \boldsymbol{y})\, d\boldsymbol{y} = \int_{\mathcal{B}_1(\boldsymbol{z})} \Psi_0(\boldsymbol{z})\, d\boldsymbol{z} = 1 \qquad \text{for } \boldsymbol{x} \in \Omega^r(t) \setminus \Gamma_{\Delta x}(t),$$

where $\mathcal{B}_1(\boldsymbol{x}) = \{\boldsymbol{y} : \|\boldsymbol{y} - \boldsymbol{x}\|_2 < 1\}$. Therefore,

$$\Psi_{\Delta x} * \chi(t, \boldsymbol{x}) = \chi(t, \boldsymbol{x}) \qquad \text{for } \boldsymbol{x} \in \Omega \setminus \Gamma_{\Delta x}. \tag{A.6}$$

Let us now assume $\boldsymbol{x} \in \mathcal{U}^k \cap \Gamma_{\Delta x}$. In this case,

$$\Psi_{\Delta x} * \chi(t, \boldsymbol{x}) = \int_{\mathcal{B}_{\Delta x}(\boldsymbol{x})} \Psi_{\Delta x}(\boldsymbol{x} - \boldsymbol{y}) \chi(\boldsymbol{y})\, d\boldsymbol{y} = \int_{\mathcal{B}_{\Delta x}(\boldsymbol{x}) \cap \Omega^r(t)} \Psi_{\Delta x}(\boldsymbol{x} - \boldsymbol{y})\, d\boldsymbol{y}$$

and using (A.2a) together with the representation

$$\boldsymbol{x} = \boldsymbol{h}^k\left(\boldsymbol{z}^k\right) + \Delta x \ell \boldsymbol{n} \circ \boldsymbol{h}^k\left(\boldsymbol{z}^{\boldsymbol{k}}\right), \tag{A.7}$$

where $\boldsymbol{z}^k \in \Theta^k$ and $\ell \in (-1, 1)$ (compare Section 2.4.1), we obtain

$$\int_{\mathcal{B}_{\Delta x}(\boldsymbol{x}) \cap \Omega^r(t)} \Psi_{\Delta x}(\boldsymbol{x} - \boldsymbol{y})\, d\boldsymbol{y} = \int_{\mathcal{B}_{\Delta x}(\boldsymbol{x}) \cap \Omega^r(t)} \frac{1}{\Delta x^d} \Psi_0\left(\frac{\boldsymbol{h}^k\left(\boldsymbol{z}^k\right) - \boldsymbol{y}}{\Delta x} + \ell \boldsymbol{n} \circ \boldsymbol{h}^k\left(\boldsymbol{z}^k\right)\right) d\boldsymbol{y}.$$

Defining now

$$\eta_{\Delta x}\left(t,\boldsymbol{z}^k,\ell\right)=\int_{\mathcal{B}_{\Delta x}(\boldsymbol{x})\cap\Omega^r(t)}\frac{1}{\Delta x^d}\Psi_0\left(\frac{\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)-\boldsymbol{y}}{\Delta x}+\ell\boldsymbol{n}\circ\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)\right)d\boldsymbol{y} \tag{A.8}$$

and recalling (A.6) we find that $\Psi_{\Delta x}*\chi$ is in fact of the structure (A.1a).

In the following, we turn our attention to the conditions collected in (A.1b). Recall that by construction, $\Psi_{\Delta x}*\chi\in\mathcal{C}^1\left([0,t_{max}]\times\Omega,\mathbb{R}\right)$. Since Γ is by definition a $\mathcal{C}^2$-surface we also have

$$\boldsymbol{h}^k\in\mathcal{C}^2\left(\Theta^k,\Gamma\right)\qquad\text{as well as}\qquad\boldsymbol{n}\in\mathcal{C}^1\left(\Gamma,\mathbb{R}^d\right)$$

and therefore,

$$\eta_{\Delta x}\in\mathcal{C}^1\left([0,t_{max}]\times\Theta^k\times(-1,1),\mathbb{R}\right).$$

Then, we determine

$$\partial_\ell\eta_{\Delta x}=\int_{\mathcal{B}_{\Delta x}(\boldsymbol{x})\cap\Omega^r(t)}\frac{1}{\Delta x^d}\left(n_\alpha\circ\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)\right)\partial_\alpha\Psi_0\left(\frac{\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)-\boldsymbol{y}}{\Delta x}+\ell\boldsymbol{n}\circ\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)\right)d\boldsymbol{y},$$

where $\alpha=1,\dots,d$. At this stage, we apply (A.7) and (A.2a) to obtain

$$\begin{aligned}\int_{\mathcal{B}_{\Delta x}(\boldsymbol{x})\cap\Omega^r(t)}&\frac{1}{\Delta x^d}\left(n_\alpha\circ\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)\right)\partial_\alpha\Psi_0\left(\frac{\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)-\boldsymbol{y}}{\Delta x}+\ell\boldsymbol{n}\circ\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)\right)d\boldsymbol{y}.\\&=\int_{\mathcal{B}_{\Delta x}(\boldsymbol{x})\cap\Omega^r(t)}\left(n_\alpha\circ\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)\right)\partial_\alpha\Psi_{\Delta x}(\boldsymbol{x}-\boldsymbol{y})\,d\boldsymbol{y}.\end{aligned}$$

Drawing now $n_\alpha\circ\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)$ under the derivative and using the divergence theorem we write

$$\int_{\mathcal{B}_{\Delta x}(\boldsymbol{x})\cap\Omega^r(t)}\left(n_\alpha\circ\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)\right)\partial_\alpha\Psi_{\Delta x}(\boldsymbol{x}-\boldsymbol{y})\,d\boldsymbol{y}=\int_{\mathcal{B}_{\Delta x}(\boldsymbol{x})\cap\Gamma(t)}n_\alpha(\boldsymbol{y})\left(n_\alpha\circ\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)\right)\Psi_{\Delta x}(\boldsymbol{x}-\boldsymbol{y})\,d\mathscr{H}(\boldsymbol{y}),$$

where $\mathscr{H}$ denotes the surface measure on Γ. Since Γ is a $\mathcal{C}^2$-surface it is possible to choose Δx small enough such that for $\boldsymbol{y}^1,\boldsymbol{y}^2\in\mathcal{B}_{\Delta x}(\boldsymbol{x})\cap\Omega^r(t)$ we have $n_\alpha\left(\boldsymbol{y}^1\right)n_\alpha\left(\boldsymbol{y}^2\right)>0$, i. e. the angle between the inner unit normal vectors at two arbitrary points in $\mathcal{B}_{\Delta x}(\boldsymbol{x})\cap\Gamma$ is less than $\pi/2$. This yields

$$n_\alpha(\boldsymbol{y})\left(n_\alpha\circ\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)\right)>0\qquad\text{for }\Delta x\text{ small enough.}$$

Furthermore, $\Psi_{\Delta x}(\boldsymbol{x}-\boldsymbol{y})>0$ on $\mathcal{B}_{\Delta x}(\boldsymbol{x})$ by definition and for $\boldsymbol{x}\in\Gamma_{\Delta x}$, the volume measure of $\mathcal{B}_{\Delta x}(\boldsymbol{x})\cap\Gamma$ is non-zero by construction. Thus, we conclude

$$\int_{\mathcal{B}_{\Delta x}(\boldsymbol{x})\cap\Gamma(t)}n_\alpha(\boldsymbol{y})\left(n_\alpha\circ\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)\right)\Psi_{\Delta x}(\boldsymbol{x}-\boldsymbol{y})\,d\mathscr{H}(\boldsymbol{y})>0$$

and therefore, $\partial_\ell\eta_{\Delta x}\geq 0$ for Δx small enough.

In the following, we shall consider properties (A.1c). For any $\boldsymbol{x}\in\Gamma_{\Delta x}\cap\mathcal{U}^k$ we have $\mathcal{B}_{\Delta x}(\boldsymbol{x})\subset\Gamma_{2\Delta x}\cap\mathcal{U}^k$, so we can use the representation

$$\boldsymbol{y}=\boldsymbol{h}^k\left(\hat{\boldsymbol{z}}^k\right)+\Delta x\hat{\ell}\boldsymbol{n}\circ\boldsymbol{h}^k\left(\hat{\boldsymbol{z}}^k\right)$$

with $\hat{\boldsymbol{z}}^k\in\Theta^k$ and $\hat{\ell}\in(-2,2)$ to rewrite the right hand side of definition (A.8) in the form

$$\begin{aligned}&\int_{\mathcal{B}_{\Delta x}(\boldsymbol{x})\cap\Omega^r(t)}\frac{1}{\Delta x^d}\Psi_0\left(\frac{\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)-\boldsymbol{y}}{\Delta x}+\ell\boldsymbol{n}\circ\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)\right)d\boldsymbol{y}\\&\quad=\int_{\Theta^k(t)}\int_0^2\frac{1}{\Delta x^d}\Psi_0\left(\frac{\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)-\boldsymbol{h}^k\left(\hat{\boldsymbol{z}}^k\right)}{\Delta x}+\ell\boldsymbol{n}\circ\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)-\hat{\ell}\boldsymbol{n}\circ\boldsymbol{h}^k\left(\hat{\boldsymbol{z}}^k\right)\right)\widehat{\mathcal{J}}\left(\hat{\boldsymbol{z}}^k,\hat{\ell}\right)d\hat{\ell}\,d\hat{\boldsymbol{z}}.\end{aligned} \tag{A.9}$$

The Jacobian $\widehat{\mathcal{J}}\left(\hat{\boldsymbol{z}}^k,\hat{\ell}\right)$ is given by

$$\widehat{\mathcal{J}}\left(\hat{\boldsymbol{z}}^k,\hat{\ell}\right)=\left|\det\begin{pmatrix}\partial_1y_1&\cdots&\partial_{d-1}y_1&\partial_{\hat{\ell}}y_1\\\vdots&&\vdots&\vdots\\\partial_1y_d&\cdots&\partial_{d-1}y_d&\partial_{\hat{\ell}}y_d\end{pmatrix}\right| \tag{A.10a}$$

and thus,

$$\widehat{\mathcal{J}}\left(\hat{\boldsymbol{z}}^k,\hat{\ell}\right)=\left|\det\begin{pmatrix}\partial_1\hat{h}_1^k+\Delta x\hat{\ell}\partial_1\hat{n}_1^k & \cdots & \partial_{d-1}\hat{h}_1^k+\Delta x\hat{\ell}\partial_{d-1}\hat{n}_1^k & \Delta x\hat{n}_1^k\\ \vdots & & \vdots & \vdots\\ \partial_1\hat{h}_d^k+\Delta x\hat{\ell}\partial_1\hat{n}_d^k & \cdots & \partial_{d-1}\hat{h}_d^k+\Delta x\hat{\ell}\partial_{d-1}\hat{n}_d^k & \Delta x\hat{n}_d^k\end{pmatrix}\right|, \tag{A.10b}$$

where $\hat{\boldsymbol{h}}^k=\boldsymbol{h}^k\left(\hat{\boldsymbol{z}}^k\right)$ and $\hat{\boldsymbol{n}}^k=\boldsymbol{n}\circ\hat{\boldsymbol{h}}^k$. This yields

$$\widehat{\mathcal{J}}\left(\hat{\boldsymbol{z}}^k,\hat{\ell}\right)=\Delta x\widetilde{\mathcal{J}}(\hat{\boldsymbol{z}},\hat{\ell})=\Delta x\left|\det\begin{pmatrix}\partial_1\hat{h}_1^k+\Delta x\hat{\ell}\partial_1\hat{n}_1^k & \cdots & \partial_{d-1}\hat{h}_1^k+\Delta x\hat{\ell}\partial_{d-1}\hat{n}_1^k & \hat{n}_1^k\\ \vdots & & \vdots & \vdots\\ \partial_1\hat{h}_d^k+\Delta x\hat{\ell}\partial_1\hat{n}_d^k & \cdots & \partial_{d-1}\hat{h}_d^k+\Delta x\hat{\ell}\partial_{d-1}\hat{n}_d^k & \hat{n}_d^k\end{pmatrix}\right|. \tag{A.11}$$

By Taylor expansion, we obtain

$$\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)=\boldsymbol{h}^k\left(\hat{\boldsymbol{z}}^k\right)+\left(z_\beta^k-\hat{z}_\beta^k\right)\partial_\beta\boldsymbol{h}^k\left(\hat{\boldsymbol{z}}^k\right)+\mathcal{O}\left(\left(z_\beta^k-\hat{z}_\beta^k\right)^2\right),$$

where $\beta=1,\ldots,d-1$, and therefore,

$$\frac{\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)-\boldsymbol{h}^k\left(\hat{\boldsymbol{z}}^k\right)}{\Delta x}=\frac{z_\beta^k-\hat{z}_\beta^k}{\Delta x}\partial_\beta\boldsymbol{h}^k\left(\hat{\boldsymbol{z}}^k\right)+\mathcal{O}\left(\Delta x\left(\frac{z_\beta^k-\hat{z}_\beta^k}{\Delta x}\right)^2\right).$$

Inserting now the definition

$$\boldsymbol{w}^k=\frac{1}{\Delta x}\left(\hat{\boldsymbol{z}}^k-\boldsymbol{z}^k\right) \tag{A.12}$$

into the right hand side yields

$$\frac{\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)-\boldsymbol{h}^k\left(\hat{\boldsymbol{z}}^k\right)}{\Delta x}=-w_\beta^k\partial_\beta\boldsymbol{h}^k\left(\boldsymbol{z}^k+\boldsymbol{w}^k\Delta x\right)+\mathcal{O}\left(\Delta x\right).$$

Expanding $\boldsymbol{n}\circ\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)$ in exactly the same way and using again (A.12) we obtain

$$\boldsymbol{n}\circ\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)=\boldsymbol{n}\circ\boldsymbol{h}^k\left(\boldsymbol{z}^k+\boldsymbol{w}^k\Delta x\right)+\mathcal{O}\left(\Delta x\right)$$

and substituting those expansions into Ψ_0 we find that

$$\begin{aligned}&\Psi_0\left(\frac{\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)-\boldsymbol{h}^k\left(\hat{\boldsymbol{z}}^k\right)}{\Delta x}+\ell\boldsymbol{n}\circ\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)-\hat{\ell}\boldsymbol{n}\circ\boldsymbol{h}^k\left(\hat{\boldsymbol{z}}^k\right)\right)\\ &\qquad=\Psi_0\Bigg(-w_\beta^k\partial_\beta\boldsymbol{h}^k\left(\boldsymbol{z}^k+\boldsymbol{w}^k\Delta x\right)+\left(\ell-\hat{\ell}\right)\left(\boldsymbol{n}\circ\boldsymbol{h}^k\left(\boldsymbol{z}^k+\boldsymbol{w}^k\Delta x\right)\right)+\mathcal{O}\left(\Delta x\right)\Bigg).\end{aligned}$$

Inserting definition (A.12) into $\widetilde{\mathcal{J}}$ yields

$$\widetilde{\mathcal{J}}\left(\boldsymbol{z}^k,\boldsymbol{w}^k,\tilde{\ell}\right)=\left|\det\begin{pmatrix}\partial_1\tilde{h}_1^k+\Delta x\tilde{\ell}\partial_1\tilde{n}_1^k & \cdots & \partial_{d-1}\tilde{h}_1^k+\Delta x\tilde{\ell}\partial_{d-1}\tilde{n}_1^k & \tilde{n}_1^k\\ \vdots & & \vdots & \vdots\\ \partial_1\tilde{h}_d^k+\Delta x\tilde{\ell}\partial_1\tilde{n}_d^k & \cdots & \partial_{d-1}\tilde{h}_d^k+\Delta x\tilde{\ell}\partial_{d-1}\tilde{n}_d^k & \tilde{n}_d^k\end{pmatrix}\right|,$$

where $\tilde{\boldsymbol{h}}^k=\boldsymbol{h}^k\left(\boldsymbol{z}^k+\boldsymbol{w}^k\Delta x\right)$ and $\tilde{\boldsymbol{n}}^k=\boldsymbol{n}\circ\tilde{\boldsymbol{h}}^k$, and recalling (A.11) we find the coordinate transformation $\Theta^k\ni\hat{\boldsymbol{z}}^k\mapsto\boldsymbol{w}^k\in\widetilde{\Theta}^k$ to turn the right hand side of equation (A.9) into

$$\begin{aligned}&\int_{\Theta^k(t)}\int_0^2\frac{1}{\Delta x^d}\Psi_0\left(\frac{\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)-\boldsymbol{h}^k\left(\hat{\boldsymbol{z}}^k\right)}{\Delta x}+\ell\boldsymbol{n}\circ\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)-\hat{\ell}\boldsymbol{n}\circ\boldsymbol{h}^k\left(\hat{\boldsymbol{z}}^k\right)\right)\widehat{\mathcal{J}}\left(\hat{\boldsymbol{z}},\hat{\ell}\right)\,d\hat{\ell}\,d\hat{\boldsymbol{z}}\\ &\qquad=\int_{\widetilde{\Theta}^k(t)}\int_0^2\Psi_0\left(-w_\beta^k\partial_\beta\boldsymbol{h}^k\left(\boldsymbol{z}^k+\boldsymbol{w}^k\Delta x\right)+\left(\ell-\hat{\ell}\right)\boldsymbol{n}\circ\boldsymbol{h}^k\left(\boldsymbol{z}^k+\boldsymbol{w}^k\Delta x\right)+\mathcal{O}\left(\Delta x\right)\right)\\ &\qquad\qquad\cdot\widetilde{\mathcal{J}}\left(\boldsymbol{z}^k,\boldsymbol{w}^k,\hat{\ell}\right)\,d\hat{\ell}\,d\boldsymbol{w}^k.\end{aligned}$$

Note here that the Jacobian of the transformation $\Theta^k\ni\hat{\boldsymbol{z}}^k\mapsto\boldsymbol{w}^k\in\widetilde{\Theta}^k$ equals Δx^{d-1}.

At this stage, we determine

$$\lim_{\Delta x\to 0}\int_{\widetilde{\Theta}^k(t)}\int_0^2 \Psi_0\left(-w_\beta^k\partial_\beta \boldsymbol{h}^k\left(\boldsymbol{z}^k+\boldsymbol{w}^k\Delta x\right)+\left(\ell-\hat{\ell}\right)\boldsymbol{n}\circ\boldsymbol{h}^k\left(\boldsymbol{z}^k+\boldsymbol{w}^k\Delta x\right)+\mathcal{O}\left(\Delta x\right)\right)$$
$$\cdot\,\widetilde{\mathcal{J}}\left(\boldsymbol{z}^k,\boldsymbol{w}^k,\hat{\ell}\right)\,d\hat{\ell}\,d\boldsymbol{w}^k=\int_{\widetilde{\Theta}^k(t)}\int_0^2\Psi_0\left(-w_\beta^k\partial_\beta\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)+\left(\ell-\hat{\ell}\right)\boldsymbol{n}\circ\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)\right)\mathcal{J}\left(\boldsymbol{z}^k\right)\,d\hat{\ell}\,d\boldsymbol{w}^k$$

with

$$\mathcal{J}(\boldsymbol{z}^k)=\left|\det\begin{pmatrix}\partial_1 h_1^k\left(\boldsymbol{z}^k\right) & \cdots & \partial_{d-1}h_1^k\left(\boldsymbol{z}^k\right) & n_1\circ\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)\\ \vdots & & \vdots & \vdots\\ \partial_1 h_d^k\left(\boldsymbol{z}^k\right) & \cdots & \partial_{d-1}h_d^k\left(\boldsymbol{z}^k\right) & n_d\circ\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)\end{pmatrix}\right|,$$

so we define

$$\eta_0\left(t,\boldsymbol{z}^k,\ell\right)=\int_{\widetilde{\Theta}^k(t)}\int_0^2\Psi_0\left(-w_\beta^k\partial_\beta\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)+\left(\ell-\hat{\ell}\right)\boldsymbol{n}\circ\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)\right)\mathcal{J}\left(\boldsymbol{z}^k\right)\,d\hat{\ell}\,d\boldsymbol{w}^k \tag{A.13}$$

to obtain $\lim_{\Delta x\to 0}\eta_{\Delta x}=\eta_0$. Recalling

$$\Psi_0\in\mathcal{C}^\infty(\Omega,\mathbb{R}),\qquad \boldsymbol{h}^k\in\mathcal{C}^2\left(\Theta^k,\Gamma\right)\qquad\text{as well as}\qquad \boldsymbol{n}\in\mathcal{C}^1\left(\Gamma,\mathbb{R}^d\right)$$

and observing that $\mathcal{J}\left(\boldsymbol{z}^k\right)$ cannot become zero we obtain

$$\eta_0\in\mathcal{C}^1\left([0,t_{max}]\times\Theta^k\times(-1,1),\mathbb{R}\right).$$

Now, it only remains to show $\partial_\ell\eta_0\geq 0$. Recalling definition (A.13) we find that

$$\partial_\ell\eta_0=\int_{\widetilde{\Theta}^k(t)}\int_0^2\left(n_\alpha\circ\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)\right)\partial_\alpha\Psi_0\left(-w_\beta^k\partial_\beta\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)+\left(\ell-\hat{\ell}\right)\boldsymbol{n}\circ\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)\right)\mathcal{J}\left(\boldsymbol{z}^k\right)\,d\hat{\ell}\,d\boldsymbol{w}^k$$

and the coordinate transformation $\widetilde{\Theta}^k\times[0,2]\ni\left(\boldsymbol{w}^k,\hat{\ell}\right)\mapsto\boldsymbol{\xi}\in\Omega^r\cap\Gamma_2^k$, where

$$\Gamma_2^k=\left\{\boldsymbol{\xi}\in\mathbb{R}^d\,:\,\boldsymbol{\xi}=\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)+w_\beta^k\partial_\beta\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)+\hat{\ell}\boldsymbol{n}\circ\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)\ \text{with}\ \hat{\ell}\in(-2,2)\right\},$$

yields

$$\int_{\widetilde{\Theta}^k(t)}\int_0^2\left(n_\alpha\circ\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)\right)\partial_\alpha\Psi_0\left(-w_\beta^k\partial_\beta\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)+\left(\ell-\hat{\ell}\right)\boldsymbol{n}\circ\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)\right)\mathcal{J}\left(\boldsymbol{z}^k\right)\,d\hat{\ell}\,d\boldsymbol{w}^k$$
$$=\int_{\Omega^r(t)\cap\Gamma_2^k}\left(n_\alpha\circ\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)\right)\partial_\alpha\Psi_0\left(\boldsymbol{\zeta}-\boldsymbol{\xi}\right)\,d\boldsymbol{\xi},$$

where

$$\boldsymbol{\zeta}=\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)+\ell\boldsymbol{n}\circ\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)\qquad\text{with}\ \ell\in(-1,1).$$

Drawing $\left(n_\alpha\circ\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)\right)$ under the derivative and using the divergence theorem we obtain

$$\int_{\Omega^r(t)\cap\Gamma_2^k}\left(n_\alpha\circ\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)\right)\partial_\alpha\Psi_0\left(\boldsymbol{\zeta}-\boldsymbol{\xi}\right)\,d\boldsymbol{\xi}=\int_{\partial\left\{\Omega^r(t)\cap\Gamma_2^k\right\}}n_\alpha(\xi)\left(n_\alpha\circ\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)\right)\Psi_0\left(\boldsymbol{\zeta}-\boldsymbol{\xi}\right)\,d\hat{\sigma}(\boldsymbol{\xi}).$$

Due to (A.2b), we have for $\boldsymbol{\zeta}$ fixed

$$\operatorname{supp}\Psi_0\left(\boldsymbol{\zeta}-\boldsymbol{\xi}\right)=\left\{\boldsymbol{\xi}\in\mathbb{R}^d\,:\,\|\boldsymbol{\zeta}-\boldsymbol{\xi}\|<1\right\}$$

and by definition of ζ and Γ_2^k, we know that

$$\left\{\boldsymbol{\xi}\in\mathbb{R}^d\,:\,\|\boldsymbol{\zeta}-\boldsymbol{\xi}\|<1\right\}\cap\partial\left\{\Omega^r(t)\cap\Gamma_2^k\right\}$$

is not of measure zero. This implies $\Psi_0>0$ on a set of non-zero measure in the domain of integration. Since for $\Delta x\to 0$ also $n_\alpha(\xi)\left(n_\alpha\circ\boldsymbol{h}^k\left(\boldsymbol{z}^k\right)\right)>0$ we have $\partial_\ell\eta_0>0$ which yields that, in fact, $\partial_\ell\eta_0\geq 0$. Therefore, we finally gain

THEOREM A.1. *If the assumptions stated in Section 2.1.1 hold and Δx is chosen small enough then the function $\Psi_{\Delta x}*\chi$ with $\Psi_{\Delta x}$ defined by* (A.2) *fulfils the conditions* (A.1).

Bibliography

[1] T. Abe, *Derivation of the lattice Boltzmann method by means of the discrete ordinate method for the Boltzmann equation*, J. Comput. Phys. **131** (1997), pp. 241–246. cited on page(s) 9

[2] E. Bänsch, *Finite element discretization of the Navier-Stokes equations with a free capillary surface*, Numer. Math. **88** (2001), pp. 203–235. cited on page(s) 28

[3] R. Benzi, S. Succi, and M. Vergassola, *The lattice Boltzmann equation: Theory and applications*, Phys. Rep. **222** (1992), pp. 145–197. cited on page(s) 9

[4] 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** (1954), pp. 511–524. cited on page(s) 9

[5] J. U. Brackbill, D. B. Kothe, and C. Zemach, *A continuum method for modeling surface tension*, J. Comput. Phys. **100** (1992), pp. 335–354. cited on page(s) 22

[6] R. P. Brent, *Algorithms for Minimization without Derivatives*, Prentice-Hall, Englewood Cliffs, 1973. cited on page(s) 92

[7] M. do Carmo, *Differentialgeometrie von Kurven und Flächen*, Vieweg, Braunschweig, 2nd ed., 1992, in German. cited on page(s) 25, 26, 42

[8] Y. C. Chang, T. Y. Hou, B. Merriman, and S. Osher, *A level set formulation of Eulerian interface capturing methods for incompressible flow*, J. Comput. Phys. **124** (1996), pp. 449–464. cited on page(s) 22

[9] H. Chen, S. Chen, and W. H. Matthaeus, *Recovery of the Navier-Stokes equations using a lattice-gas Boltzmann approach*, Phys. Rev. A **45** (1992), pp. R5339–R5342. cited on page(s) 9, 29

[10] S. Chen and G. D. Doolen, *Lattice Boltzmann method for fluid flows*, Annu. Rev. Fluid Mech. **30** (1998), pp. 329–364. cited on page(s) 9

[11] R. M. Corless, G. H. Gonnet, D. E. G. Hare, J. D. Jeffrey, and D. E. Knuth, *On the Lambert W function*, Adv. Comput. Math. **5** (1996), pp. 329–359. cited on page(s) 60

[12] R. Cornubert, D. d'Humières, and D. Levermore, *A Knudsen layer theory for lattice gases*, Physica D **41** (1991), pp. 241–259. cited on page(s) 19

[13] R. Dautray and J.-L. Lions, *Mathematical Analysis and Numerical Methods for Science and Technology*, vol. 2, Springer, Berlin, 1988. cited on page(s) 22, 107, 109

[14] U. D'Ortona, D. Salin, M. Cieplak, R. B. Rybka, and J. R. Banavar, *Two-color nonlinear Boltzmann cellular automata: surface tension and wetting*, Phys. Rev. E **51** (1995), pp. 3718–3728. cited on page(s) 55

[15] M. Feistauer, *Mathematical Methods in Fluid Dynamics*, Longman, Harlow, 1993. cited on page(s) 11, 23

[16] W. H. Fleming, *Functions of Several Variables*, Addison-Wesley, Reading, 1965. cited on page(s) 23, 26, 33

[17] U. Frisch, B. Hasslacher, and Y. Pomeau, *Lattice-gas automata for the Navier-Stokes equation*, Phys. Rev. Lett. **56** (1986), pp. 1505–1508. cited on page(s) 9

[18] I. M. Gel'fand and varying co-authors, *Generalized Functions*, Academic Press, New York, 1964-1968, 5 volumes. cited on page(s) 22

[19] I. Ginzbourg and P. M. Adler, *Boundary flow condition analysis for the three-dimensional lattice Boltzmann model*, J. Phys. II France **4** (1994), pp. 191–214. cited on page(s) 14

[20] I. Ginzbourg and D. d'Humières, *Local second order boundary methods for lattice Boltzmann models*, J. Stat. Phys. **84** (1996), pp. 927–971. cited on page(s) 14

[21] I. Ginzburg, private communication, 2001. cited on page(s) 55

[22] I. Ginzburg and K. Steiner, *A free-surface lattice Boltzmann method for modelling the filling of expanding cavities by Bingham fluids*, Phil. Trans. R. Soc. Lond. A **360** (2002), pp. 453–466. cited on page(s) 31, 105

[23] I. Ginzburg and K. Steiner, *Lattice Boltzmann model for free-surface and its application to filling process in casting*, accepted for publication in J. Comput. Phys., 2002. cited on page(s) 31, 105

[24] I. Ginzburg and G. Wittum, *Two-phase flows on unstructured grids with spline volume tracking*, J. Comput. Phys. **166** (2001), pp. 302–335. cited on page(s) 49, 91

[25] T. Grahs, A. Meister, and T. Sonar, *Image processing for numerical approximations of conservation laws: nonlinear anisotropic artificial dissipation*, SIAM J. Sci. Comp. **23** (2002), pp. 1439–1455. cited on page(s) 57

[26] D. Grunau, S. Chen, and K. Eggert, *A lattice Boltzmann model for multiphase fluid flows*, Phys. Fluids A **5** (1993), pp. 2557–2562. cited on page(s) 29, 30

[27] A. K. Gunstensen and D. H. Rothman, *Microscopic modeling of immiscible fluids in three dimensions by a lattice Boltzmann method*, Europhys. Lett. **18** (1992), pp. 157–161. cited on page(s) 29

[28] A. K. Gunstensen, D. H. Rothman, S. Zaleski, and G. Zanetti, *Lattice Boltzmann model of immiscible fluids*, Phys. Rev. A **43** (1991), pp. 4320–4327. cited on page(s) 29

[29] W. Hackbusch, *Theorie und Numerik elliptischer Differentialgleichungen*, Teubner, Stuttgart, 1986, in German. cited on page(s) 80

[30] X. He and L.-S. Luo, *Lattice Boltzmann model for the incompressible Navier-Stokes equation*, J. Stat. Phys. **88** (1997), pp. 927–944. cited on page(s) 14

[31] X. He and L.-S. Luo, *Theory of the lattice Boltzmann method: From the Boltzmann equation to the lattice Boltzmann equation*, Phys. Rev. E **56** (1997), pp. 6811–6817. cited on page(s) 9

[32] X. He, Q. Zou, L.-S. Luo, and M. Dembo, *Analytic solutions of simple flow and analysis of noslip boundary conditions for the lattice Boltzmann BGK model*, J. Stat. Phys. **87** (1997), pp. 115–136. cited on page(s) 14

[33] F. J. Higuera and J. Jiménez, *Boltzmann approach to lattice gas simulations*, Europhys. Lett. **9** (1989), pp. 663–668. cited on page(s) 9

[34] D. J. Holdych, D. Rovas, J. G. Georgiadis, and R. O. Buckis, *An inproved hydrodynamics formulation for multiphase flow lattice-Boltzmann models*, Int. J. Mod. Phys. C **9** (1998), pp. 1393–1304. cited on page(s) 29

[35] S. Hou, X. Shan, Q. Zou, G. D. Doolen, and W. E. Soll, *Evaluation of two lattice Boltzmann models for multiphase flow*, J. Comput. Phys. **138** (1997), pp. 695–713. cited on page(s) 29

[36] D. d'Humières, *Generalized lattice Boltzmann equations*, in: Rarified Gas Dynamics: Theory and Simulations (B. D. Shizgal and D. P. Weaver, eds.), Progress in Astronautics and Aeronautics, vol. 159, AIAA, Washington, 1992. cited on page(s) 12

[37] D. d'Humières, M. Bouzidi, and P. Lallemand, *Thirteen-velocity three-dimensional lattice Boltzmann model*, Phys. Rev. E **63** (2001), pp. 066702, 1–7. cited on page(s) 12

[38] D. d'Humières, P. Lallemand, and U. Frisch, *Lattice gas models for 3d hydrodynamics*, Europhys. Lett. **2** (1986), pp. 291–297. cited on page(s) 9

[39] B. Jawerth, P. Lin, and E. Sinzinger, *Lattice Boltzmann models for anisotropic diffusion*, J. Math. Imaging Vis. **11** (1999), pp. 231–237. cited on page(s) 57

[40] M. Junk, *Kinetic schemes in the case of low Mach numbers*, J. Comput. Phys. **151** (1999), pp. 947–968. cited on page(s) 9

[41] M. Junk, *On the construction of discrete equilibrium distributions for kinetic schemes*, ITWM Report 14, Institut für Techno- und Wirtschaftsmathematik, Kaiserslautern, 1999. cited on page(s) 77

[42] M. Junk, *A finite difference interpretation of the lattice Boltzmann method*, Numer. Methods Partial Differ. Equations **17** (2001), pp. 383–402. cited on page(s) 11, 13

[43] M. Junk, *LBM – discrete dynamics and finite difference method*, in: Proceedings of the GAMM Workshop "Discrete Modelling and Discrete Algorithms in Continuum Mechanics" (T. Sonar and I. Thomas, eds.), Logos, Berlin, 2001. cited on page(s) 10, 11, 13

[44] M. Junk, A. Klar, and L.-S. Luo, *Theory of the lattice Boltzmann method: Mathematical analysis of the lattice Boltzmann equation*, submitted to J. Comput. Phys., 2002. cited on page(s) 14

[45] A. Klar, *Relaxation scheme for a lattice-Boltzmann-type discrete velocity model and numerical Navier-Stokes limit*, J. Comput. Phys. **148** (1999), pp. 416–432. cited on page(s) 12

[46] J. Köhnlein, *Volume-of-Fluid-Methoden zur Frontenverfolgung*, Diplomarbeit, Universität Hamburg, Hamburg, 1999, in German. cited on page(s) 91

[47] B. Lafaurie, C. Nardone, R. Scardovelli, S. Zaleski, and G. Zanetti, *Modelling merging and fragmentation in multiphase flows with SURFER*, J. Comput. Phys. **113** (1994), pp. 134–147. cited on page(s) 25, 49

[48] P. Lallemand and L.-S. Luo, *Theory of the lattice Boltzmann method: dispersion, dissipation, isotropy, Galilean invariance, and stability*, Phys. Rev. E **61** (2000), pp. 6546–6562. cited on page(s) 12

[49] L. D. Landau and E. M. Lifschitz, *Lehrbuch der Theoretischen Physik*, vol. 6, Akademie-Verlag, Berlin, 3rd ed., 1978, in German. cited on page(s) 11, 22, 49

[50] P.-L. Lions, *Mathematical Topics in Fluid Mechanics*, vol. 1, Clarendon Press, Oxford, 1996. cited on page(s) 11

[51] L.-S. Luo, *Unified theory of lattice Boltzmann models for nonideal gases*, Phys. Rev. Lett. **81** (1998), pp. 1618–1621. cited on page(s) 9, 29

[52] L.-S. Luo, *Theory of the lattice Boltzmann method: lattice Boltzmann models for nonideal gases*, Phys. Rev. E **62** (2000), pp. 4982–4996. cited on page(s) 9, 29

[53] G. R. MacNamara and G. Zanetti, *Use of the Boltzmann equation to simulate lattice-gas automata*, Phys. Rev. Lett. **61** (1988), pp. 2332–2335. cited on page(s) 9

[54] R. Mei, L.-S. Luo, and W. Shyy, *An accurate curved boundary treatment in the lattice Boltzmann method*, J. Comput. Phys. **155** (1999), pp. 307–330. cited on page(s) 14

[55] R. Mei, W. Shyy, D. Yu, and L.-S. Luo, *Lattice Boltzmann method for 3-d flows with curved boundaries*, J. Comput. Phys. **161** (2000), pp. 680–699. cited on page(s) 14

[56] B. J. Miller, *Improvements in Computing Multiple Phase Flows*, doctoral thesis, University of California, Los Angeles, 1997. cited on page(s) 22

[57] X. Nie, Y.-H. Qian, G. D. Doolen, and S. Chen, *Lattice Boltzmann simulation of the two-dimensional Rayleigh-Taylor instability*, Phys. Rev. E **58** (1998), pp. 6861–6864. cited on page(s) 30, 48

[58] E. Orlandini, M. R. Swift, and J. M. Yeomans, *A lattice Boltzmann model of binary-fluid mixtures*, Europhys. Lett. **32** (1995), pp. 463–468. cited on page(s) 29

[59] W. R. Osborn, E. Orlandini, M. R. Swift, J. M. Yeomans, and J. R. Banavar, *Lattice Boltzmann study of hydrodynamic spinodal decomposition*, Phys. Rev. Lett. **75** (2000), pp. 4031–4034. cited on page(s) 29

[60] R. Peyret and T. D. Taylor, *Computational Methods for Fluid Flow*, Springer, New York, 1983. cited on page(s) 81

[61] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C*, Cambridge University Press, Cambridge, 2nd ed., 1992. cited on page(s) 92

[62] E. G. Puckett, A. S. Almgren, J. B. Bell, D. L. Marcus, and W. T. Rider, *A high-order projection method for tracking fluid interfaces in variable density incompressible flow*, J. Comput. Phys. **130** (1997), pp. 269–282. cited on page(s) 91, 93

[63] Y. H. Qian, D. d'Humières, and P. Lallemand, *Lattice BGK models for Navier-Stokes equation*, Europhys. Lett. **17** (1992), pp. 479–484. cited on page(s) 9, 29

[64] M. B. Reider and J. D. Sterling, *Accuracy of discrete-velocity BGK models for the simulation of the incompressible Navier-Stokes equations*, Computers & Fluids **24** (1995), pp. 459–467. cited on page(s) 11

[65] W. J. Rider and D. B. Kothe, *Reconstructing volume tracking*, J. Comput. Phys. **141** (1998), pp. 112–152. cited on page(s) 91, 93

[66] D. H. Rothman and J. M. Keller, *Immiscible cellular-automaton fluids*, J. Stat. Phys. **52** (1988), pp. 1119–1127. cited on page(s) 29

[67] D. H. Rothman and S. Zaleski, *Lattice-Gas Cellular Automata*, Cambridge University Press, Cambridge, 1997. cited on page(s) 9

[68] R. Scardovelli and S. Zaleski, *Direct numerical simulation of free-surface and interfacial flow*, Annu. Rev. Fluid Mech. **31** (1999), pp. 567–603. cited on page(s) 22

[69] L. Schwartz, *Théorie des distributions*, Hermann, Paris, rev. and suppl. ed., 1973, in French. cited on page(s) 22

[70] X. Shan and H. Chen, *Lattice Boltzmann model for simulating flows with multiple phases and components*, Phys. Rev. E **47** (1993), pp. 1815–1819. cited on page(s) 29

[71] X. Shan and H. Chen, *Simulation of nonideal gases and liquid-gas phase transitions by the lattice Boltzmann equation*, Phys. Rev. E **49** (1994), pp. 2941–2948. cited on page(s) 29

[72] X. Shan and G. D. Doolen, *Multicomponent lattice-Boltzmann model with interparticle interaction*, J. Stat. Phys. **81** (1995), pp. 379–393. cited on page(s) 29

[73] X. Shan and G. D. Doolen, *Diffusion in a multicomponent lattice Boltzmann equation model*, Phys. Rev. E **54** (1996), pp. 3614–3620. cited on page(s) 29

[74] M. Spivak, *A Comprehensive Introduction to Differential Geometry*, Publish or Perish, Boston, 1970-1975, 5 volumes. cited on page(s) 25, 26, 42

[75] G. Strang, *On the construction and comparison of difference schemes*, SIAM J. Numer. Anal. **5** (1968), pp. 506–517. cited on page(s) 95

[76] M. R. Swift, E. Orlandini, W. R. Osborn, and J. M. Yeomans, *Lattice Boltzmann simulations of liquid-gas and binary fluid systems*, Phys. Rev. E **54** (1996), pp. 5041–5052. cited on page(s) 29

[77] M. R. Swift, W. R. Osborn, and J. M. Yeomans, *Lattice Boltzmann simulations of nonideal fluids*, Phys. Rev. Lett. **75** (1995), pp. 830–833. cited on page(s) 29

[78] J. Tölke, *Gitter-Boltzmann-Verfahren zur Simulation von Zweiphasenströmungen*, Shaker, Aachen, 2001, in German. cited on page(s) 55

[79] J. Weickert, *Anisotropic Diffusion in Image Processing*, Teubner, Stuttgart, 1998. cited on page(s) 57, 59, 60, 61, 63

[80] F. M. White, *Viscous Fluid Flow*, McGraw-Hill, Singapore, 2nd ed., 1991. cited on page(s) 11

[81] D. Wolf-Gladrow, *Lattice-Gas Cellular Automata and Lattice Boltzmann Models*, Springer, Berlin, 2000. cited on page(s) 9

[82] M. M. Zdravkovich, *Flow around Circular Cylinders*, vol. 1, Oxford University Press, Oxford, 1997. cited on page(s) 14

On the author

Personal data

Name:	Dirk Kehrwald
Date of birth:	27th October 1972
Place of birth:	Annweiler am Trifels, Germany

School education

August 1979 to June 1983	Schillerschule, Münchweiler an der Rodalb, Germany (primary school)
August 1983 to June 1992	Immanuel-Kant-Gymnasium, Pirmasens (secondary school)
June 1992	Abitur (final examination)

National service

July 1992 to June 1993	National service in the German army

University studies

October 1993 to March 1999	Student of technomathematics at the University of Kaiserslautern, Germany (physics and computer science as secondary subjects)
March 1996	Intermediate Diploma in Technomathematics
October 1996 to March 1999	Participant of the ECMI Industrial Mathematics Educational Programme
March 1999	Diploma in Technomathematics
Since April 1999	Doctoral candidate at the Mathematical Department of the University of Kaiserslautern

Professional career

April 1999 to April 2000	Holder of a scholarship from the Deutsche Forschungsgemeinschaft (German Research Council)
May 2000 to October 2000	Scientific employee of the Mathematical Department of University of Kaiserslautern
November 2000 to September 2001	Holder of a scholarship from the Deutsche Forschungsgemeinschaft
October 2001 to June 2002	Collaborator in the project "Analysis of lattice Boltzmann methods" of the Deutsche Forschungsgemeinschaft
Since July 2002	Scientific employee of the Fraunhofer Institute for Industrial Mathematics (ITWM), Kaiserslautern

Über den Autor

Persönliche Daten

Name:	Dirk Kehrwald
Geburtsdatum:	27. Oktober 1972
Geburtsort:	Annweiler am Trifels

Schulbildung

August 1979 bis Juni 1983	Schillerschule, Münchweiler an der Rodalb (Grundschule)
August 1983 bis Juni 1992	Immanuel-Kant-Gymnasium, Pirmasens
Juni 1992	Abitur

Wehrdienst

Juli 1992 bis Juni 1993	Wehrdienst in Veitshöchheim und Zweibrücken

Studium

Oktober 1993 bis März 1999	Student der Technomathematik an der Universität Kaiserslautern (Nebenfächer: Physik und Datenverarbeitung)
März 1996	Vordiplom in Technomathematik
Oktober 1996 bis März 1999	Teilnehmer am ECMI-Studienprogramm „Industrial Mathematics“
März 1999	Diplom in Technomathematik
Seit April 1999	Doktorand am Fachbereich Mathematik der Universität Kaiserslautern

Beruflicher Werdegang

April 1999 bis April 2000	DFG-Stipendiat im Graduiertenkolleg Technomathematik an der Universität Kaiserslautern
Mai 2000 bis Oktober 2000	Wissenschaftlicher Mitarbeiter am Fachbereich Mathematik der Universität Kaiserslautern
November 2000 bis September 2001	DFG-Stipendiat im Graduiertenkolleg Technomathematik an der Universität Kaiserslautern
Oktober 2001 bis Juni 2002	Wissenschaftlicher Mitarbeiter im DFG-Projekt „Analyse von Lattice-Boltzmann-Methoden“ an der Universität Kaiserslautern
Seit Juli 2002	Wissenschaftlicher Mitarbeiter am Fraunhofer-Institut für Techno- und Wirtschaftsmathematik (ITWM) in Kaiserslautern