Eindhoven University of Technology

MASTER

Learning the lattice Boltzmann method gas-wall interaction scattering kernel from molecular dynamics data

Beunen, J.M.P.

Award date:
2022

Link to publication

Disclaimer
This document contains a student thesis (bachelor's or master's), as authored by a student at Eindhoven University of Technology. Student theses are made available in the TU/e repository upon obtaining the required degree. The grade received is not published on the document as presented in the repository. The required complexity or quality of research of student theses may vary by program, and the required minimum study period may vary in duration.

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain

Take down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Download date: 16. Mar. 2025

Eindhoven University of Technology

Mechanical Engineering
Energy Technology & Fluid Dynamics

Applied Physics
Fluids & Flows

Learning the lattice Boltzmann method gas-wall interaction scattering kernel from molecular dynamics data

J.M.P. Beunen

Supervisors:

prof. E.H. van Brummelen
S. Mohammad Nejad M.Sc.

prof. F. Toschi
dr. A. Corbetta
dr. A. Gabbana

January 27, 2022

R-2102-A

This report was made in accordance with the TU/e Code of Scientific Conduct for the Master thesis.

Abstract

In this thesis gas-wall interaction modelling for rarefied gases is further investigated. As a starting point, gas-wall interaction data from molecular dynamics simulations of argon gas between two parallel gold walls is analyzed. The performance of standard parametric kernels as well as Gaussian mixture based scattering kernels is tested. It is found that Gaussian mixture based scattering kernels outperform the standard parametric kernels on the molecular dynamics data in several aspects. Further, a general coupling procedure between Gaussian mixture based scattering kernels and the lattice Boltzmann method is proposed for gas-wall interaction modelling. The coupling procedure is tested in the lattice Boltzmann method on rarefied Couette flow simulations with velocity slip near the walls. It is found that the performance of the coupled boundary condition correctly converges to the analytical solution. Additionally, the results show an improvement on existing rarefied gas-wall interaction boundary conditions.

Acknowledgements

First of all I would like to thank all my supervisors for the support and the useful discussions we have had during the project. Furthermore, special thanks to Shahin Mohammad Nejad for setting up and running all the molecular dynamics simulations that were used for this thesis.

Contents

1. Introduction

1.1 Rarefied gas flows

One way to quantify the type of flow regime in a fluid dynamics problem is by an indication of the mean free path of the molecules with respect to the size of the system. To do this, generally the Knudsen number is being used which is defined as $\mathrm{Kn} = \frac{\lambda}{L}$. Here, λ is the mean free path length of a molecule and L is the characteristic length scale of the system under consideration. The Knudsen number allows for classification of a flow problem in several flow regimes which are continuum flow ($\mathrm{Kn} < 0.01$), slip flow ($0.01 < \mathrm{Kn} < 0.1$), transitional flow ($0.1 < \mathrm{Kn} < 10$) and free-molecular flow ($\mathrm{Kn} > 10$)[1]. When the Knudsen number of a flow problem starts to become large enough for the problem to move into the transitional flow regime, well known continuum approaches start to become inaccurate [2, 3]. In these cases of increasing rarefaction one needs to resort to approaches that allow for taking into account the microscopic behaviour of the molecules.

Before looking more in depth into how to model rarefied flow problems it is interesting to discuss some situations where these phenomena occur. In terms of technological applications, especially lithography and flights high up in the atmosphere stand out. During the fabrication of chips, atomic layers are being etched at close to vacuum conditions. Given that it is nearly impossible to remove all gas molecules from the light-exposition environment, the behaviour of the present gas is important to model since it can cause contamination of the system's components [4]. High altitude flight is another distinct application of rarefied gas flows. In this case one is usually dealing with low Reynolds/high Mach number flows. While this reduces the need for turbulence modelling, accurately computing aerodynamic forces and heat generation still remains challenging [5].

1.2 Gas-wall interaction modelling

As with all fluid dynamics problems, accurately predicting the behaviour in the bulk is vital for obtaining a correct solution. Hence, there has already been much effort to model gas-gas interactions when modelling rarefied gases. This can be seen by the vast amount of literature available on the topic. However, if the gas-wall interaction is not also accurately taken care of the solution of the full flow problem quickly deteriorates. When the Knudsen number is small and in the hydrodynamic regime, gas-wall interaction can be modeled on a macroscopic scale by means of the no-slip condition. Though, when the gas becomes rarefied, its molecular behaviour becomes more expressive on the macroscopic scale and this condition can no longer be seen as accurate. In these cases, one commonly resorts to scattering kernels to describe the interaction of gas molecules with a solid surface.

The interactions of gas-molecules with solid surfaces under rarefied conditions have been investigated as much as the behaviour in the bulk [6]. For these interactions, the used

gas-wall interaction model appears to be able to significantly alter the dynamics of a flow problem [7]. Previous research has come up with several empirical gas-wall interaction models. Unfortunately, these gas-wall interaction models are generally not expressive enough to model the actual gas-wall interaction with sufficient accuracy.

1.3 Project goals

The goal of this thesis is to take a more in depth look into approaches to model gas-wall interaction throughout the entire Knudsen range. This will be done by making use of molecular dynamics (MD) simulations. While there already exist several empirical models [6, 8–13], as well as some exploratory work with machine learning to model rarefied gas-wall interaction [14–17], there are still various unexplored research directions. Given the unexplored research directions, the goal of this thesis is to take a better look at machine learning to find an accurate description for this gas-wall interaction. Such an interaction model can then be applied as a new type of boundary condition in higher level simulation methods like the lattice Boltzmann method (LBM). How such a coupling can be done will also be further investigated.

1.4 Thesis outline

This thesis is structured in nine chapters and implicitly structured into two main parts. The first part is focused on analyzing MD data and constructing scattering kernels based upon this data. This will be done in the conventional way via standard parametric scattering kernels as well as by means of machine learning to construct parametric scattering kernels from which the parameters are learned. As a machine learning model, Gaussian mixture models (GMMs) are being used. This part consists of the following four chapters:

- Chapter 2: In this chapter the molecular dynamics will be further elucidated upon. The fundamentals of the simulation technique as well as the current setup will be discussed.
- Chapter 3: The topic of Chapter 3 are the parametric scattering kernels that can be widely found in literature. Theory as well as application to the current problem will be debated.
- Chapter 4: In this chapter the Gaussian mixture based machine learning technique to construct parametric scattering kernels with learned parameters will be discussed upon.
- Chapter 5: Chapter 5 is devoted to determining the performance of the scattering kernels considered in Chapter 3 and 4. The kernels will be compared with the data provided by the molecular dynamics simulations from the setup discussed in Chapter 2.

The second part looks further into a possible coupling of the scattering kernels developed in the first part to the lattice Boltzmann method. This is done by making use of a discretization of the scattering kernel learned by the machine learning model. Herein, the following chapters can be distinguished:

- Chapter 6: In this chapter of the thesis the used LBM setup will be elaborated upon. This chapter acts as a preparation for the next chapter on the coupling algorithm.
- Chapter 7: In this chapter the coupling algorithm to couple the Gaussian mixture based scattering kernels to the lattice Boltzmann method is worked out.
- Chapter 8: The results of the hybrid simulation technique will be elaborated upon in this chapter. Special attention will be given to quantifying the performance of the proposed boundary condition for simulation of a rarefied Couette flow.

Finally, Chapter 9 presents the main conclusions of the thesis and a discussion.

2. Molecular dynamics

One way of simulating the gas-wall interactions of rarefied gases is via molecular dynamics simulations. In this chapter the underlying principles as well as the current setup will be discussed.

2.1 Molecular dynamics fundamentals

For simulations on the microscale level, like molecular dynamics simulations, all individual molecules in a system are tracked. While it would be the preferred way to describe the full dynamics of a system, such a microscopic description of the dynamics of the state of a gas is a nearly impossible task in practice. Given the vast amount of gas molecules that need to be considered for macroscopic volumes, it is inconceivable from a computational point of view. This makes such a description for macroscopic systems only useful for conceptual purposes. However, if a small amount of molecules is being considered it is still possible to compute the full dynamics of a fluid problem on a microscopic scale.

To simplify the description of the dynamics, as well as the state of a system, it is useful to only consider monatomic gases. For this type of gases the microscopic descriptions that are used in molecular dynamics simulations are relatively simple. The dynamics of these gas molecules are characterized by classical mechanics, which is nothing more than Newton's laws of motion, and an interaction potential between molecules to compute forces. Therefore, this enables us to write the equation of motion for the particles in a system as

$$m_i \frac{\mathrm{d}\mathbf{v}_i}{\mathrm{d}t} = \mathbf{F}_i. \tag{2.1}$$

Where i is an index that denotes a particular particle and m_i denotes the mass of that particle. The velocity of the particle is equal to $\mathbf{v}_i$ and the time derivative of $\mathbf{v}_i$ corresponds to the particle's acceleration. Furthermore, the force acting on this particle $\mathbf{F}_i$ can be written as

$$\mathbf{F}_i = -\sum_{j \neq i} \boldsymbol{\nabla} U(r_{ij}). \tag{2.2}$$

Here, a summation is done over all other particles to compute the force they exercise on particle i. The force is defined by the gradient of the interaction potential U which is dependent on the distance r_{ij} between the molecules i and j. While there exist several interaction potentials in literature, the most simple potential and the most commonly used when studying rarefied gases is the Lennard-Jones (LJ) potential. This potential can be written as

$$U(r_{ij}) = 4\varepsilon \left[\left(\frac{\sigma}{r_{ij}} \right)^{12} - \left(\frac{\sigma}{r_{ij}} \right)^{6} \right]. \tag{2.3}$$

This inter-particle potential consists of a steep repulsive term, $(\sigma/r_{ij})^{12}$, that becomes prevalent for short distances as well as a smoother attractive term for larger distances,

$(\sigma/r_{ij})^6$. Here, ε and σ are parameters denoting the well depth and the van der Waals radius respectively [18].

When an inter-particle potential is chosen, the state of a system at a later time can be computed by integrating the equations of motion of all particles to compute new velocities and positions. Doing this for a large amount of time steps makes it possible to compute how a particular system evolves in time. As a result, complex molecular systems can be simulated from which the dynamics cannot be predicted analytically.

2.2 Gas-wall interaction simulation techniques

To be able to thoroughly describe gas-wall interactions it is necessary to have a solid understanding of the scattering dynamics that are at play. Molecular dynamics is an excellent simulation technique to simulate gas-wall interaction on an atomic scale. However, as discussed before it is also computationally very expensive which makes it impossible to use directly as a simulation method for gas-wall interaction in macroscopic systems. Although, it is possible to use this simulation technique to compute the scattering dynamics between a gas and a wall for very small systems. The results that are obtained for these simulations can then be used as boundary conditions for other less computationally expensive simulation techniques.

As an example, one of the methods that can be used is by looking at the temperate gradient along a nanochannel filled with a gas during a simulation. Scattering interactions can then be computed by splitting the distributions moving to and from the wall. This was done in a paper by Nedea et al. [19].

Another possible approach is to look at changes in velocities after a molecule hits the wall. In this case several methods are possible like molecular beam models [20]. With these models, molecules are being shot at a wall after which changes in particle properties are being measured. A disadvantage of this approach is that gas-gas interactions are not taken into account. Therefore, this approach can only be considered valid when highly rarefied gases are studied.

A different method that looks at changes in velocities after a molecule hits the wall is based upon doing molecular dynamics simulations between two plates with periodic boundary conditions [17, 21]. A virtual border is then placed just in front of the wall, at the cut-off distance of the wall's interaction potential, and particles are being tracked whenever they cross the virtual border. Changes between the properties of the particles between two consecutive border crossings are being stored and counted as a single gas-wall interaction. With this approach also gas-gas interactions are taken into account in the gas-wall interaction. Hence, this can be regarded to give valid results for arbitrary Knudsen number.

2.3 Current simulation setup

In this thesis an equivalent setup with parallel walls is being used for data generation. A schematic of this three-dimensional setup can be seen below in Figure 2.1. This system to investigate gas-wall interactions is, apart from certain parameter settings, identical to the setup in the papers by Nejad et al. [17, 21].

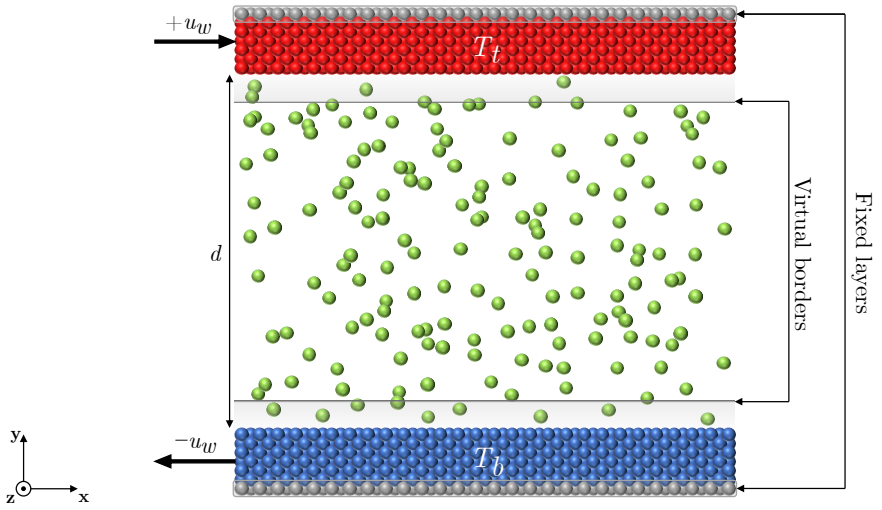


Figure 2.1: A schematic of the molecular dynamics (MD) simulation setup. Two walls with a temperature T_t and T_b kept a distance d apart from each other. The top wall is moving with a velocity $+u_w$ and the bottom wall with a velocity $-u_w$. Gas-wall interactions are recorded at the virtual borders which are placed at the cut-off distance of the wall's interaction potential.

The upper and bottom plate at which the gas-wall interactions occur, have a cross-section A of 12.5 nm by 12.5 nm and lie parallel to the x-z plane in the simulations. However, given that the simulation borders in the x and the z-direction are periodic the simulation practically extends over the entire x-z plane. Both walls are made of 9610 gold atoms arranged in a face-centred cubic (FCC) structure. The outermost layers of gold atoms in both walls that are not bordering the gas molecules have a fixed positioning. This is to make sure the wall cannot undergo any translational or rotational motion. Although, all the other layers that are close to the gas molecules can move to make sure that the gas-wall interaction can still be simulated accurately. For the dynamics of the wall molecules, the embedded atom potential (EAM) is used [22]. The wall temperature is set by making use of a Berendsen thermostat with a damping constant of 100 fs.

The gas molecule used in all the simulations is argon and the molecule used for the wall is gold. Since the gas molecule is monatomic, this avoids the need to take non-translational degrees of freedom into account for gas-wall interactions. No external forces act on the gas

molecules, their dynamics are only influenced by Lennard-Jones based interactions with other gas and wall molecules. The distance between the two walls is denoted as d and is equal to 110 nm for all the simulations in this report. The number of gas molecules is denoted as N_g and is equal to 18000. These parameters together with the cross-section of the wall were chosen to accommodate for a Knudsen number Kn = 0.01. Therefore, all the considered simulations can be classified to be in the slip flow regime according to the Knudsen number. Hence, it can be assumed that rarefaction effects only need to be taken into account near the walls while the macroscopic behaviour of the bulk can still be described by the Navier-Stokes equations.

Simulations are initialized by setting the temperature of the gas equal to the temperature of the bottom wall $T_b = 300\,\mathrm{K}$. The temperature of the top wall is a parameter that differs between simulations. After the initialization the system undergoes no changes in the number of particles, volume or total energy (NVE/microcanonical ensemble). Subsequently, an equilibration phase of 3 ns is performed after which execution continues for another 25 ns to do the data generation. The time step used for the entire simulation is 1 fs.

To simulate non-equilibrium cases, a temperature difference between the plates can be set. For this thesis, the convention that the bottom wall has a lower or equal temperature than the upper wall is applied. As stated before, the temperature of the bottom wall is denoted with T_b and the temperature of the upper wall with T_t. Furthermore, non-equilibrium cases can also be simulated by setting a velocity difference between the walls in the x-direction. Here, the convention is applied that the bottom walls moves with a velocity $-u_w$ and the top wall moves with a velocity $+u_w$. This allows for setting up a dimensionless speed ratio which can be used to quantify how fast the walls are moving. This speed ratio S_w is defined as

$$S_w = \frac{u_w}{\sqrt{2kT_b/m_g}} \tag{2.4}$$

where k is the Boltzmann constant and m_g the molecular mass of the argon gas molecules.

As can be seen in Figure 2.1, in the setup there are virtual borders present close to the walls. These borders stretch over the full cross-section and are placed at the cut-off distance of the Lennard-Jones potential between the gas and the wall molecules. Accordingly, the virtual borders are placed a distance $d_{\mathrm{vb}} = 1.2\,\mathrm{nm}$ away from the walls. Due to the cut-off of the potential, whenever a gas molecule crosses a virtual border towards one of the walls, it can be stated that the molecule is interacting with the wall. Whenever the molecule crosses this virtual border again but in the direction of the bulk it can be affirmed that a gas-wall interaction has occurred. A dataset with information on the gas-wall interaction for a particular setup is generated by storing the state of the gas molecule at the virtual border; incoming and outgoing velocity vector pairs are being stored for later analysis. It should be noted that no information is stored on what actually happens between the virtual border and the wall. Both gas-gas and gas-wall interactions can influence the velocity and the trajectory of the molecules in this region. Though, this information is only indirectly captured by the stored velocity vectors.

2.4 Analyzed simulations

For the results in this report four molecular dynamics simulations were analyzed. These simulations generated datasets for five different gas-wall interaction velocity distributions. The simulations can be classified into three groups. First there is the equilibrium simulation were both walls are at the same temperature and the walls are not moving with respect to each other. Secondly two simulations were conducted where both walls are still at the same temperature but moving with respect to each other. Finally, there is also one simulation where the walls are not moving with respect to each other but there is a temperature difference between the walls. Given that the temperature gradient depends on the wall one is looking at, this simulation gave two datasets to analyze. The velocity distributions are dependent on whether the hot or the cold wall is being considered. Below in Table 2.1 the parameter settings of the four simulations/five generated datasets is further elucidated upon. The datasets are abbreviated with E1 for the equilibrium situation that was conducted, V1 and V2 for the two datasets with a wall velocity and T1 and T2 for the two datasets with a different wall temperature. Thorough analysis of these simulations will be postponed till Chapter 5. First, the used scattering kernels will be further explained upon in Chapter 3 and Chapter 4.

Table 2.1: The molecular dynamics simulations that were conducted for this thesis. The first column denotes the abbreviation that will be used to refer to a particular dataset. Besides, the second column displays the number of data points the dataset consists of. Furthermore, the third column indicates whether the top (t) or bottom (b) wall was used to generate the data points. In the last four columns numerical values for the wall velocity and temperatures are given.

Dataset abbr.	Data points nr.	Wall	Velocity parameters		Thermal parameters	
			S_w [-]	u_w [m/s]	T_b [K]	T_t [K]
E1	227741	b	0.00	0.00	300	300
V1	236103	b	0.25	88.35	300	300
V2	240051	b	0.50	176.69	300	300
T1	333679	b	0.00	0.00	300	600
T2	240878	t	0.00	0.00	300	600

3. Parametric scattering kernels

In this chapter standard parametric scattering kernels will be discussed. In particular, it will be discussed what kinds of standard parametric scattering kernels will be used in the rest of the thesis as well as how these can be related to the molecular dynamics data and sampled.

3.1 Scattering dynamics

A depiction of the monatomic scattering dynamics generated by the molecular dynamics setup described in Section 2.3 can be seen below in Figure 3.1. For each gas-wall interaction the incident-outgoing velocity vectors $\mathbf{v}'$-$\mathbf{v}$ at the virtual border are stored in the generated datasets. From these velocity vectors, the incident-outgoing vertical angles γ'-γ can be computed for further processing. Other angles that are important to compute are the horizontal deviation angle from the tangential vector η_t as well as the horizontal deviation angle from the x-axis η_x.

For the scattering kernels of the molecules at the virtual border several assumptions are made. First of all, it is assumed that the scattering at the virtual border happens instantaneously. As a result, a time dependency of the scattering process between the virtual border and the wall does not have to be taken into account. Second, it is assumed that the particle crosses the virtual border again at the same location. Here, two arguments are given why these assumptions could be considered reasonable. First, it should be noted that the characteristic dimensions and timescales of these effects are in the order of nanometers and nanoseconds. Therefore, when considering macroscopic systems these effects easily become negligibly small. Second, with the current setup the situation can be supposed to vary statistically only in the y-direction. Hence, in each x-z plane the situation can be considered approximately equivalent everywhere. Since this also includes the plane which contains the virtual border the previously described effects can be ruled out.

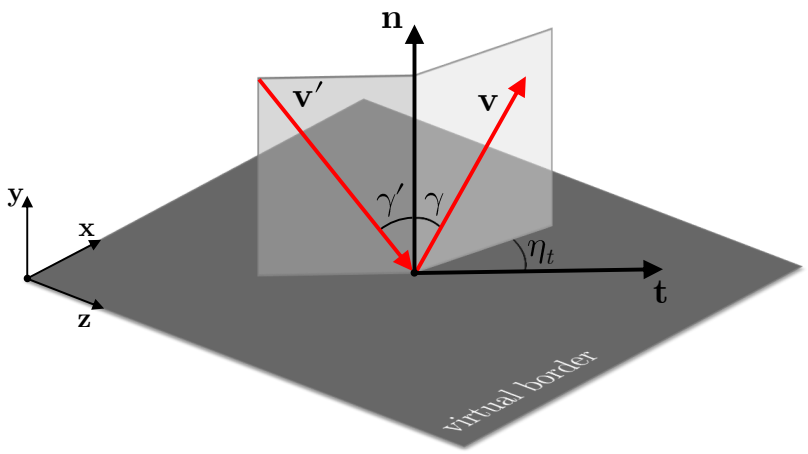


Figure 3.1: A schematic depiction of the scattering of a monatomic gas molecule from the virtual border with incident-outgoing velocity vectors, $\mathbf{v}'$-$\mathbf{v}$, incident-outgoing vertical angles, γ'-γ, and horizontal deviation angle from the tangential vector, η_t. The normal and the tangential vector are denoted with $\mathbf{n}$ and $\mathbf{t}$ respectively.

3.2 Accommodation coefficients

To describe gas-wall interactions by parametric models, usually several dimensionless parameters are computed that describe the average scattering behaviour of incoming molecules. These parameters are called accommodation coefficients (ACs) and can be seen as parameters providing a description for the gas and the wall in terms of their properties. For the stochastic interaction models that will be discussed in Section 3.3, several accommodation coefficients form part of the input commonly.

Given that the value of an accommodation coefficient is generally very sensitive to many gas and surface conditions, special care should be taken when computing them. Therefore, the method with which these ACs are being computed is very important, inconsistencies between different methods often exist [20]. There are several approaches available with which accommodation coefficients can be computed. Most of these are based on simulating the scattering behaviour of individual molecules by means of molecular dynamics simulations, as described in Section 2.2.

Independent of the method used to simulate the gas-wall interaction, the standardized form to compute ACs is

$$\alpha = \frac{\langle \varphi^{\text{in}} \rangle - \langle \varphi^{\text{out}} \rangle}{\langle \varphi^{\text{in}} \rangle - \langle \varphi^{\text{wall}} \rangle}. \tag{3.1}$$

for accommodation coefficients α related to energies. In this equation, the brackets $\langle ... \rangle$ denote averaging over the molecular quantity φ inside. Incoming populations with respect to the wall are denoted with the superscript *in* whereas outgoing populations are indicated

with *out*. When properties that hold for the thermal wall distribution are referred to, the subscript *wall* is used to indicate this. A similar formula can be used to compute accommodation coefficients related to momentum σ. While the values for in- and outgoing distributions in this equation can be computed from the molecular dynamics simulations, the values for the property denoted by *wall* need to be computed from the equilibrium distributions at the wall.

Equation (3.1) is a widely accepted approach to compute accommodation coefficients for gas-wall interactions. However, when the temperature of the gas approaches the temperature of the wall, the distributions of the incoming velocities and the thermal wall start to coincide. As a result, the denominator in (3.1) is approaching zero which invalidates the computation. To counter this issue, another method to compute the accommodation coefficients was proposed in a paper bij Spijker et al. [23]. This method makes use of the velocity correlations between the in- and outgoing particles. As stated in the paper, the goal of this approach is to have an AC close to zero when the velocity correlation is strong (reflective collisions) and an AC close to one when a velocity correlation is nonexistent (diffusive collisions). This can be done by computing the accommodation coefficients based on a line that best fits the collision data by doing least-squares regression. Thereafter, this line can be compared with the diagonal (reflective collisions) and the horizontal (diffusive collisions). Next, it should be noted that reflective and diffusive collisions are extreme cases and all other collision distributions will give a value in between these two generally. Accordingly, an energy accommodation coefficient is given as $\alpha = 1 - \beta$ where β is the slope of the least squares regression of the collision data. As such, a different computation for the accommodation coefficients is

$$\alpha = 1 - \frac{\sum_i \left(\varphi_i^{\text{in}} - \left\langle \varphi^{\text{in}} \right\rangle\right)\left(\varphi_i^{\text{out}} - \left\langle \varphi^{\text{out}} \right\rangle\right)}{\sum_i \left(\varphi_i^{\text{in}} - \left\langle \varphi^{\text{in}} \right\rangle\right)^2}. \tag{3.2}$$

In this formula, the superscripts have their usual meaning while the index i denotes a particular particle. Thereupon, accommodation coefficients are computed by doing summations over the particle properties. Given its superior performance when the ingoing distribution starts to overlap with the wall distribution, this method to compute ACs will be used in this thesis.

Finally, some words on the terminology used to denote various types of accommodation coefficients. To denote an accommodation coefficient in formula notation, the Greek letter σ is used for momentum accommodation coefficients and α for energy accommodation coefficients. The subscript denotes the direction over which this accommodation coefficient is computed. Furthermore, in this report as well as other literature accommodation coefficient in text are often abbreviated as AC. The first letter before AC denotes the physical property over which the accommodation coefficient is being computed. Here MAC stands for momentum accommodation coefficient and EAC for energy accommodation coefficient. Furthermore, usually there is also a distinction in the direction with respect to the wall

for which an accommodation coefficient is being computed. In this case, normal and tangential directions are being distinguished. Hence, NMAC stands for normal momentum accommodation coefficient and TMAC for tangential momentum accommodation coefficient. One thing should be noted when considering tangential accommodation coefficients: with these ACs it is possible to distinguish several types. First, there are general tangential ACs which are computed for both tangential directions simultaneously; the vector norm of the projection of the velocity vector on the tangential plane is used. Second, there are also tangential ACs that are computed for one tangential direction in particular. In the rest of the report it will be ensured that it is clear from the context which tangential ACs are meant when they are used.

3.3 Interaction models

In this section, the most common description for stochastic gas-wall interaction models will be further elaborated on. The goal of a gas-wall interaction model is to accurately map the properties of gas molecules before collision to the wall with the properties of gas molecules after collision. This is generally done by predicting the velocity distribution of gas molecules bouncing from the wall based on their incoming velocity distribution. An important parameter herein is the scattering kernel $P(\mathbf{v}' \to \mathbf{v})$ which is the conditional probability of an incoming particle with velocity $\mathbf{v}'$ being bounced back with an outgoing velocity $\mathbf{v}$. The full velocity distribution functions of the in- and outgoing particles can be computed by means of the scattering kernel by

$$|v_n|\, f^{\text{out}}(\mathbf{v}) = \int_{\mathbf{v}'\cdot\mathbf{n}<0} |v_n'|\, P\left(\mathbf{v}' \to \mathbf{v}\right) f^{\text{in}}\left(\mathbf{v}'\right) \mathrm{d}\mathbf{v}'. \tag{3.3}$$

In this equation the terms $f^{\text{in}}(\mathbf{v}')$ and $f^{\text{out}}(\mathbf{v})$ are the velocity distribution functions of the incoming and the outgoing molecules respectively. The subscript n is used to denote a normal component and the letter $\mathbf{n}$ to denote the normal vector of the wall pointing towards the gas. Furthermore, v_n and v_n' denote the scalar normal projection of the velocity vectors.

3.4 Scattering kernels

Several scattering kernels have been developed to describe gas-wall interaction by means of (3.3). In this section the most common scattering kernels will be discussed for usage in this equation. These scattering kernels take into account different accommodation coefficients as well as scattering dynamics. Therefore, the resulting scattering behaviour of molecules differs per scattering kernel.

3.4.1 Purely reflective kernel

The first kind of scattering kernel that will be considered here is the purely reflective kernel. This kernel can be displayed as

$$P\left(\mathbf{v}' \to \mathbf{v}\right) = \delta\left(\mathbf{v}' - \mathbf{v} + 2\mathbf{n}v_n\right) \tag{3.4}$$

and was first introduced as a gas-wall interaction model by Maxwell in 1879 [8]. A collision is purely reflective when a particle collides with a wall that is perfectly smooth and there exists an infinite interaction potential with the wall. In simple physics, this means that the wall is hard and does not deform when it is hit by the particle, neither is there energy exchange between the wall and the particle. In this case the particle is bounced from the wall with the normal velocity component inverted and the parallel velocity components unchanged. As can be seen in (3.4) this is denoted by a Dirac delta function from which the argument is zero only when the reflection is purely specular. Consequently, also the outgoing velocity distribution function f^{out} is the ingoing velocity distribution f^{in} with the normal component inverted when this scattering kernel is used in (3.3).

3.4.2 Thermal wall kernel

Another wall model proposed by Maxwell is the thermal wall model [8]. This wall model can be seen as the opposite of a purely reflective wall as discussed above. In the thermal wall model, the wall is assumed to be so rough that when a molecule collides with the wall it undergoes a whole series of collisions with the wall. Given that the wall is rough, it is assumed that after the series of collisions the velocity of the particle is fully randomized and therefore uncorrelated to the initial velocity. Hence, it is assumed that when the particle leaves the wall, f^{out} is equal to the equilibrium distribution of a gas which has the same temperature as the wall T_w. For an ideal gas this results in a Maxwell-Boltzmann distribution wherefore the scattering kernel is equal to

$$P\left(\mathbf{v}' \to \mathbf{v}\right) = \frac{m^2 v_n}{2\pi\left(kT_w\right)^2} \exp\left(-\frac{mv^2}{2kT_w}\right). \tag{3.5}$$

In this equation, the symbols have their usual meaning. Besides, m denotes the mass of an impinging particle and v the vector norm of the velocity vector $\mathbf{v}$.

3.4.3 Maxwell kernel

The two kernels discussed above can be combined by means of linear superposition to form the Maxwell scattering kernel which describes combined specular and diffusive reflection of incoming molecules. This scattering kernel can be written as

$$P\left(\mathbf{v}' \to \mathbf{v}\right) = (1-\alpha)\delta\left(\mathbf{v}' - \mathbf{v} + 2\mathbf{n}v_n\right) + \alpha\frac{m^2 v_n}{2\pi\left(kT_w\right)^2} \exp\left(-\frac{mv^2}{2kT_w}\right) \tag{3.6}$$

and is a combination of the two kernels discussed above with the exemption of a scaling factor. As holds for the individual components, this linear combination has also been described by Maxwell in [8]. To scale the two parts of the kernel an accommodation coefficient is used which can be the TMAC or the EAC depending on the flow situation [9]. In this case the EAC is chosen as an accommodation coefficient to scale the kernels; it can be seen as the factor denoting the ratio between the two scattering regimes. The EAC scales the weight of the diffusion present in the gas-wall interaction and is between zero and one by definition. In the extreme cases, the purely reflective and thermal wall kernel are re-obtained.

3.4.4 Extended Maxwell model/Maxwell-Yamamoto kernel

The Maxwell model as described in the section above has been extended by Yamamoto et al. to allow for different accommodation coefficients for the tangential and normal velocity components [9]. This scattering kernel decomposes the dependence between both velocity components as

$$P\left(v_t' \to v_t\right) = \left(1-\alpha_t\right)\delta\left(v_t' - v_t\right) + \alpha_t \sqrt{\frac{m}{2\pi k T_w}} \exp\left(-\frac{m v_t^2}{2 k T_w}\right) \tag{3.7}$$

for the two tangential components, and

$$P\left(v_n' \to v_n\right) = \left(1-\alpha_n\right)\delta\left(v_n' + v_n\right) + \alpha_n \frac{m v_n}{k T_w} \exp\left(-\frac{m v_n^2}{2 k T_w}\right) \tag{3.8}$$

for the normal component. In these two equations, α_t denotes the TEAC for the x and z directions which should be computed separately. Further, α_n denotes the NEAC in the direction normal to the wall. As can be seen, these two scattering kernels look very similar to (3.6). It is notable that with this kernel there is no direct correlation between the normal and the tangential velocity components. There is only an implicit relation between their accommodation coefficients, which can be seen as a shortcoming of the model.

3.4.5 Cercignani-Lampis-Lord kernel

Another scattering kernel that is often used to describe gas-wall interactions is the Cercignani-Lampis-Lord (CLL) kernel [6, 10] which originates from the kernels developed by Cercignani and Lampis [11–13]. The kernel can be written as

$$P\left(v_t' \to v_t\right) = \frac{1}{\sqrt{\pi \sigma_t \left(2-\sigma_t\right)}} \exp\left(\frac{-\left(v_t - \left(1-\sigma_t\right) v_t'\right)^2}{\sigma_t \left(2-\sigma_t\right)}\right) \tag{3.9}$$

for the two tangential components and as

$$P\left(v_n' \to v_n\right) = \frac{2u}{\alpha_n} I_0\left(\frac{2\sqrt{\left(1-\alpha_n\right)} v_n v_n'}{\alpha_n}\right) \exp\left(\frac{-\left(v_n^2 + \left(1-\alpha_n\right) v_n'^2\right)}{\alpha_n}\right) \tag{3.10}$$

for the normal component. The Cercignani-Lampis kernel has three accommodation coefficients that are directly related to physical properties of the system. These accommodation coefficients, denoted here by σ_t and α_n are two TMACs and one NEAC respectively. I_0 denotes the zeroth order Bessel function of the first kind. This kernel imposes an empirical relation between the TMACs and TEACs, that is, $\alpha_t = \sigma_t (2 - \sigma_t)$. Therefore, as can also be seen in (3.9), this kernel produces a reflected distribution of molecules that is a drifting Maxwell-Boltzmann distribution with a temperature of $\alpha_t T_w$ where α_t is the empirically computed AC. Furthermore, using this empirical relation the mean velocity of the reflected molecules is equal to $\sqrt{1 - \alpha_t}$ times the impinging velocity for this kernel. This model allows a direct relation between the normal and tangential velocity components. Although, as stated, several accommodation coefficients are only taken into account empirically.

3.5 Sampling Algorithms

Before most of the scattering kernels discussed in Section 3.4 can be used it is necessary to first compute the correct accommodation coefficients from the molecular dynamics data. Given that the walls can be moving it is necessary to first transform the velocity data from the simulations to the moving inertial system of the wall being considered. This transformation only has a small influence on the values of the accommodation coefficients that have a dependency on the direction in which the wall is moving. However, it has to be taken into account to ensure that the scattering results produced by the parametric kernels are correct.

To sample the kernels, an equivalent transformation of the velocity data is being done as with the accommodation coefficients. Since the walls can move, the velocities in the dataset that form the ingoing distribution have to be transformed towards the inertial system of the wall. Hereafter, the velocity data can be fed to the parametric scattering kernels from which outgoing velocities can be sampled. To do this sampling, two cases are distinguished: sampling of kernels that are closely related to the Maxwell kernel and sampling of the CLL kernel.

3.5.1 Maxwellian kernel sampling

For kernels that are related to the Maxwell kernel, kernels that consist of a specular reflective part, a diffusive reflective part or both are considered. This includes every kernel in Section 3.4 except the CLL kernel. Given that a specular reflective part can be implemented by doing nothing more than an inversion of the normal velocities, the difficulty lies in the diffusive reflections. Fortunately it is possible to separate a three-dimensional diffusive reflection into three equations that denote a diffusive reflection along all coordinate directions. This reduces the problem towards sampling a one dimensional probability distribution. Sampling these kinds of distributions can be done quickly as long as cumulative distributions functions and percent point functions can be readily computed. This is the case for the distributions being dealt with.

3.5.2 Cercignani-Lampis-Lord kernel sampling

To sample the Cercignani-Lampis-Lord kernel the distributions are not sampled directly. Instead, use is made of an algorithm developed by Peddakotla et al. [24]. This algorithm can be seen below in Table 3.1 where some symbols were changed to accommodate for the notation used in this thesis.

Table 3.1: The algorithm used to compute the scattered velocities for the CLL kernel. $x_{1,\dots,6}$ are random numbers sampled uniformly between 0 and 1 and $v'_{mp,s}$ is the most probable speed of the ingoing velocities. The accommodation coefficients for the tangential directions x and z are computed by the empirical formulas $\alpha_x = \sigma_x (2 - \sigma_x)$ and $\alpha_z = \sigma_z (2 - \sigma_z)$ respectively.

x-component	y-component	z-component
$r_x = \sqrt{-\alpha_x \ln x_1}$	$r_y = \sqrt{-\alpha_y \ln x_2}$	$r_z = \sqrt{-\alpha_z \ln x_3}$
$\theta_x = 2\pi x_4$	$\theta_y = 2\pi x_5$	$\theta_z = 2\pi x_6$
$v_{x,m} = \frac{v'_x\sqrt{1-\alpha_x}}{v'_{mp,s}}$	$v_{y,m} = \frac{v'_y\sqrt{1-\alpha_y}}{v'_{mp,s}}$	$v_{z,m} = \frac{v'_z\sqrt{1-\alpha_z}}{v'_{mp,s}}$
$v_x = v'_{mp,s}\left(v_{x,m} + r_x \cos\theta_x\right)$	$v_y = v'_{mp,s}\sqrt{r_1^2 + v_{y,m}^2 + 2 r_y v_{y,m} \cos\theta_y}$	$v_z = v'_{mp,s}\left(v_{z,m} + r_z \sin\theta_z\right)$

Finally, the scattered velocities of all kernels are transformed back to the stationary inertial system. This terminates the sampling procedures.

4. Gaussian mixture based scattering kernels

In this chapter, the Gaussian mixture based scattering kernels that were used are further elucidated upon. These kernels are based upon Gaussian mixture models as a machine learning technique for the molecular dynamics data. The basis for these models will be discussed first. Hereafter, data preprocessing, fitting procedures, sampling and hyperparameter optimization will be debated.

4.1 Gaussian mixture models

In this thesis, Gaussian mixture models will be used as a basis for scattering kernels. Setting up a scattering kernel is in principle the same as finding a suitable conditional density estimation model to describe the scattering behaviour of the molecules. To do this several options are available from which the most notable are the standard parametric models from which several have been described in Chapter 3. As will be shown in Chapter 5, the performance of these parametric models provides a relatively accurate description of the scattering but can be improved. Therefore, it becomes viable to also look into other approaches for doing conditional density estimation. In previous years several new techniques have been developed and tested to describe the scattering of molecules. Among these are non-parametric models based on kernel density estimation or distributed element trees [14, 15] as well as parametric models based on Gaussian mixtures [16, 17]. Although Gaussian mixture models are classified as parametric models, their performance does not have to be inferior to non-parametric models given that the number of parameters can be set arbitrarily. This also gives them arbitrary flexibility. Other approaches that would be possible for probability density estimation to be used in scattering kernels are for instance histograms and neural density estimation [25]. However, all of these methods have their drawbacks which could make them less preferable than Gaussian mixture models for the current problem. The computational complexity of non-parametric models scales with the number of data points, histograms have unwanted smoothing problems and neural density estimation is more suited for high-dimensional data [25]. With GMMs, the computational complexity can be tuned together with the models predictive accuracy. Furthermore, with GMMs there is a direct correlation between the velocity profiles in all directions. This is not trivial for several other methods. Therefore, the focus will be on Gaussian mixture models for estimating the probability densities in this thesis.

4.2 Data preprocessing

Before the molecular dynamics data can be fitted with a Gaussian mixture model, it is necessary to apply several transformations on the dataset. In principle, it is not necessary

to transform the velocities to the inertial system of the wall. Given that this is only a linear transposition of the data points, the Gaussian mixture model has enough flexibility to capture this. Only adjusting the means of the internal Gaussian functions is enough to obtain the same performance. However, given that a transformation to the inertial system of the wall offers considerable advantages when coupling Gaussian mixture model based scattering kernels to the lattice Boltzmann method, this transformation is still applied.

Furthermore, it is necessary to make sure that the support constraints of the Gaussian distributions used by the model as well as the molecular dynamics data correspond. The support constraints of the normal components, $v_n' < 0$ and $v_n > 0$, are not compatible with Gaussian distributions. To ensure compatibility, the normal components of the molecular dynamics data will be preprocessed by transforming these approximately Rayleigh distributions to approximately Gaussian distributions. This is based on an equivalent procedure by Liao et al. [16]. First, the probability density function of the Rayleigh distribution

$$f(u) = \frac{u}{\theta^2} e^{-u^2/(2\theta^2)}, \quad u \geq 0 \tag{4.1}$$

is fitted through the absolute values of the velocities in the normal directions to obtain a scale parameter θ for the ingoing velocities v_n' as well as a scale parameter for the outgoing velocities v_n. This scale parameter can then be used to transform the normal components to an estimated Gaussian distribution via the transformation

$$\mathcal{T}(u) = \sqrt{2}\theta \, \mathrm{erf}^{-1}\left[1 - 2\exp\left(-\frac{u^2}{2\theta^2}\right)\right]. \tag{4.2}$$

Applying this transformation on the normal data will ensure that the data points obey approximately Gaussian distributions in all directions. Moreover, the support constraints of the Gaussian functions are satisfied after which a Gaussian mixture model can be fitted through the data. When the fitting of a Gaussian mixture model is completed, samples can be transformed backwards by the transformation

$$\mathcal{T}^{-1}(u) = \sqrt{-2\theta^2 \ln\left[\frac{1}{2} - \frac{1}{2}\,\mathrm{erf}\left(\frac{u}{\sqrt{2}\theta}\right)\right]} \tag{4.3}$$

while using the same scale parameters as obtained before.

4.3 Fitting procedure

After the data preprocessing, a Gaussian mixture can be fitted on the scattering data. This mixture model based scattering kernel can be written as

$$P(\mathbf{v}' \to \mathbf{v}) = P(\mathbf{v} \mid \mathbf{v}') = P(\boldsymbol{X}) = \sum_j \phi_j P_j(\boldsymbol{X}) = \sum_j \phi_j \mathcal{N}\left(\boldsymbol{X} \mid \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j\right) \tag{4.4}$$

to produce the desired density estimate. In this equation, the index j runs over the K multivariate Gaussian distributions $\mathcal{N}$ that are in the mixture model. Each multivariate Gaussian distribution j is weighted by mixing coefficient ϕ_j, has a mean vector $\boldsymbol{\mu}_j$ and covariance matrix $\boldsymbol{\Sigma}_j$. In addition, each of the multivariate Gaussian distributions in (4.4) can generally be written as

$$\mathcal{N}\left(\boldsymbol{X} \mid \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j\right) = \frac{1}{(2\pi)^{d/2}} \frac{1}{|\boldsymbol{\Sigma}_j|^{1/2}} \exp\left[-\frac{1}{2}\left(\boldsymbol{X} - \boldsymbol{\mu}_j\right)^T \boldsymbol{\Sigma}_j^{-1}\left(\boldsymbol{X} - \boldsymbol{\mu}_j\right)\right] \tag{4.5}$$

with d the length of the feature vector. Given that the vectors for both the incoming as well as the outgoing velocities have three components d equals six. Furthermore, the other parameters of the two equations above can be further written out as

$$\boldsymbol{X} = \begin{bmatrix} \mathbf{v} \\ \mathbf{v}' \end{bmatrix}, \qquad \boldsymbol{\mu}_j = \begin{bmatrix} \boldsymbol{\mu}_{j\mathbf{v}} \\ \boldsymbol{\mu}_{j\mathbf{v}'} \end{bmatrix}, \qquad \boldsymbol{\Sigma}_j = \begin{bmatrix} \boldsymbol{\Sigma}_{j\mathbf{v}\mathbf{v}} & \boldsymbol{\Sigma}_{j\mathbf{v}\mathbf{v}'} \\ \boldsymbol{\Sigma}_{j\mathbf{v}'\mathbf{v}} & \boldsymbol{\Sigma}_{j\mathbf{v}'\mathbf{v}'} \end{bmatrix}, \tag{4.6}$$

where $\mathbf{v}'$ and $\mathbf{v}$ in the subscripts act as indices to denote to which velocities the particular part of the matrices is associated.

4.3.1 Expectation–maximization algorithm

Assuming a suitable parameter initialization, optimization of the parameters ϕ_j, $\boldsymbol{\mu}_j$ and $\boldsymbol{\Sigma}_j$ can be done for all K multivariate Gaussians on different datasets by means of the expectation-maximization (EM) algorithm [26]. This algorithm is an iterative procedure which consist of an expectation and a maximization step. During the expectation step the likelihood $\mathcal{L}$ of the dataset on a particular set of parameters is computed. This can be done as

$$\mathcal{L} = \prod_i P\left(\boldsymbol{X}_i\right) = \prod_i \sum_j \phi_j \mathcal{N}\left(\boldsymbol{X}_i \mid \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j\right) \tag{4.7}$$

where the index i runs over all N samples in the dataset. Equation (4.7) only holds exactly when all samples are generated independently. Also, it is possible to compute the possibility of a sample i being drawn from cluster j by

$$P_{ij} = \frac{\phi_j \mathcal{N}\left(\boldsymbol{X}_i \mid \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j\right)}{P\left(\boldsymbol{X}_i\right)}. \tag{4.8}$$

With this information on the likelihood of a dataset and the cluster sample probabilities P_{ij} for all values i and j the algorithm can continue with the maximization step. During this step, the parameters of the Gaussian mixture model are optimized to yield the largest likelihood on the full dataset. This can be done in the following way for the parameters ϕ_j, $\boldsymbol{\mu}_j$ and $\boldsymbol{\Sigma}_j$ that define the model:

$$\widehat{\phi}_j = \frac{1}{N} \sum_i P_{ij}, \tag{4.9}$$

$$\widehat{\boldsymbol{\mu}}_j = \frac{\sum_i P_{ij} \boldsymbol{X}_i}{\sum_i P_{ij}}, \tag{4.10}$$

$$\widehat{\boldsymbol{\Sigma}}_j = \frac{\sum_i \left(\boldsymbol{X}_i - \widehat{\boldsymbol{\mu}}_j\right) \otimes \left(\boldsymbol{X}_i - \widehat{\boldsymbol{\mu}}_j\right)}{\sum_i P_{ij}}. \tag{4.11}$$

Where the hat denotes the new parameters to be used for the next EM-iteration. The algorithm can now continue with a new iteration. The iterations should continue till the parameters converge or until another stopping criterion is met.

4.4 Sampling algorithm

There are two sampling techniques used in this thesis for the kernels based on Gaussian mixture models. First, there is the full distribution sampling which can be used to compare the probability density functions of the fitted Gaussian mixture with the probability density functions of the molecular dynamics data as well as other scattering kernels based on sampled data. Second, there is the conditional sampling of the Gaussian mixture model. This allows for the implementation of traditional scattering procedures were the ingoing velocity of a particle is known and the outgoing velocity should be sampled. Furthermore, the conditional sampling procedure makes it possible for the GMM based scattering kernel to be implemented in MD simulations as well as Direct Simulation Monte Carlo (DSMC).

4.4.1 Full distribution sampling

Sampling from the full distribution can be done by making use of (4.4). For each sample $\boldsymbol{X}$ to be drawn from the fitted Gaussian mixture, first all K individual multivariate Gaussian distributions with means $\boldsymbol{\mu}_j$ and covariance matrices $\boldsymbol{\Sigma}_j$ need to be sampled [27]. Thereafter, one of these samples needs to be chosen by sampling a multinomial distribution. The probability for returning the sample taken from the ith Gaussian is equal to the value of the mixing coefficient ϕ_i.

4.4.2 Conditional distribution sampling

When the Gaussian mixture needs to be sampled conditionally, the ingoing velocity $\mathbf{v}'$ is known and the outgoing velocity $\mathbf{v}$ should be sampled from the distribution. In this case, the marginal distribution for each multivariate Gaussian can be written as [27, 28]

$$P_j\left(\mathbf{v}'\right) = \int P_j\left(\mathbf{v}, \mathbf{v}'\right) \mathrm{d}\mathbf{v} = \mathcal{N}\left(\mathbf{v}' \mid \boldsymbol{\mu}_{j\mathbf{v}'}, \boldsymbol{\Sigma}_{j\mathbf{v}'\mathbf{v}'}\right). \tag{4.12}$$

This enables the conditional probability distribution for each Gaussian multivariate to be written as

$$P_j\left(\mathbf{v} \mid \mathbf{v}'\right) = \frac{P_j\left(\mathbf{v}, \mathbf{v}'\right)}{P_j\left(\mathbf{v}'\right)} = \mathcal{N}\left(\mathbf{v} \mid \boldsymbol{\mu}_{j\mathbf{v}|\mathbf{v}'}, \boldsymbol{\Sigma}_{j\mathbf{v}|\mathbf{v}'}\right) \tag{4.13}$$

with means

$$\boldsymbol{\mu}_{j\mathbf{v}|\mathbf{v}'} = \boldsymbol{\mu}_{j\mathbf{v}} + \boldsymbol{\Sigma}_{j\mathbf{v}\mathbf{v}'}\boldsymbol{\Sigma}^{-1}_{j\mathbf{v}'\mathbf{v}'}\left(\mathbf{v}' - \boldsymbol{\mu}_{j\mathbf{v}'}\right) \tag{4.14}$$

and covariance matrices

$$\boldsymbol{\Sigma}_{j\mathbf{v}|\mathbf{v}'} = \boldsymbol{\Sigma}_{j\mathbf{v}\mathbf{v}} - \boldsymbol{\Sigma}_{j\mathbf{v}\mathbf{v}'}\boldsymbol{\Sigma}^{-1}_{j\mathbf{v}'\mathbf{v}'}\boldsymbol{\Sigma}_{j\mathbf{v}'\mathbf{v}}. \tag{4.15}$$

Given the equation for the conditional probability distribution of a single Gaussian multivariate distribution, the conditional probability distribution for the full GMM can be formed and is equal to

$$P\left(\mathbf{v} \mid \mathbf{v}'\right) = \sum_j \tilde{\phi}_j P_j\left(\mathbf{v} \mid \mathbf{v}'\right) \tag{4.16}$$

with mixing coefficients

$$\tilde{\phi}_j = \frac{\phi_j P_j\left(\mathbf{v}'\right)}{\sum_j \phi_j P_j\left(\mathbf{v}'\right)}. \tag{4.17}$$

4.5 Hyperparameter optimization

The Gaussian mixture model scattering kernels used in this report are built upon the Gaussian mixture model implementation available in `scikit-learn` [29]. Therefore, several hyperparameter optimization options are available out-of-the-box. For GMMs, the most important hyperparameter to optimize is the number of Gaussian distributions, also called the number of components, used for the mixture. For the results in this report the number of components was varied from 1 to 256 for which the results will be discussed in Chapter 5. The number of components was not increased to values higher than 256 due to computational limitations. Furthermore, several covariance types were tested. While it is possible to couple the covariance matrices of different Gaussian distributions in (4.4) this was not done. For coupled matrices, the computation times were approximately equal although the performance was worse. Therefore, full covariance matrices that are not coupled were used everywhere. The initialization state of the weights as well as the means were left at their defaults. The results for these settings appeared to be good already. Likewise, the regularization settings were left at their very small defaults. Given that the number of data points was higher than 200K for all datasets and the number of Gaussian distributions not more than 256, the chances of overfitting are very small. Hence, applying extra regularization does not make a lot of sense.

5. Scattering kernel performance comparison

In this chapter the performance of the scattering kernels will be compared with the molecular dynamics data. First, the accommodation coefficients between the generated datasets are discussed. Hereafter, Kullback-Leibler divergences along with outgoing velocity distributions will be further elaborated upon.

5.1 Data generation

This chapter is devoted to investigating how well the scattering kernels discussed in Chapter 3 and 4 perform with respect to the results from the molecular dynamics simulations in Chapter 2. Therefore, the ingoing velocity data from the MD datasets will be fed to the scattering kernels to make them generate outgoing velocities. To check how well the kernels perform, the distribution of the generated outgoing velocities will be compared with the outgoing velocity distribution of the MD data. This makes it possible to see how well the scattering kernels can reproduce the velocity distribution of the MD data. To keep this chapter somewhat concise only the kernel performance on a fraction of the molecular dynamics datasets will be discussed in this chapter. To treat the results for an equilibrium simulation, a simulation with moving walls as well as a simulation with a temperature gradient, the three datasets E1, V1 and T1 were chosen to be discussed more extensively.

To check the performance of the parametric scattering kernels, the accommodation coefficients for the parametric kernels are computed from the molecular dynamics data as described in Section 3.5. Hereafter the incoming velocity distribution from the molecular dynamics data is fed to these kernels and outgoing velocities are generated as described in the same section. To make the generated velocity distributions displayed in this chapter more trustworthy each incoming velocity is scattered 16 times before generating the distributions. Furthermore, to keep this chapter to the point only the most advanced parametric kernels from Chapter 3 will be discussed. The analysis will be limited to the Maxwell, Maxwell-Yamamoto and CLL kernel.

Before the properties of the Gaussian mixture model based scattering kernels are inspected, they are first fitted on the datasets as described in Section 4.4. Hereafter, to analyze the performance of the Gaussian mixture kernels conditional distribution sampling can be chosen to generate outgoing velocities. However, this kind of sampling is relatively expensive with respect to full distribution sampling. Therefore, full distribution sampling will be used for the Gaussian mixture models. Since we are only interested in the outgoing velocity distributions it can be stated that due to the law of large numbers, sampling a large number of velocity sets makes the differences between the two methods negligible. In this case, the outgoing velocity distribution produced by the conditional sampling will be

equivalent to the velocity distribution that is generated by sampling the full distribution and discarding the incoming velocities.

5.2 Accommodation coefficients

One way of checking how well a scattering kernel performs is by looking at the accommodation coefficients that can be computed from the generated outgoing distributions. Furthermore, while this allows for checking the performance of the standard parametric kernels with respect to the molecular dynamics data in the first place, it can also be used as an optimization procedure for the GMM kernels. Below, in Figure 5.1 the EAC and TMAC for the molecular dynamics data as well as the standard parametric kernels discussed in this chapter are shown. Also, the performance on these accommodation coefficients of the GMM kernel is plotted for an increasing number of components. The EAC and TMAC were chosen here given that they are often used as accommodation coefficients to quantify gas-wall interaction in the case of a Couette or a Poisseuille flow.

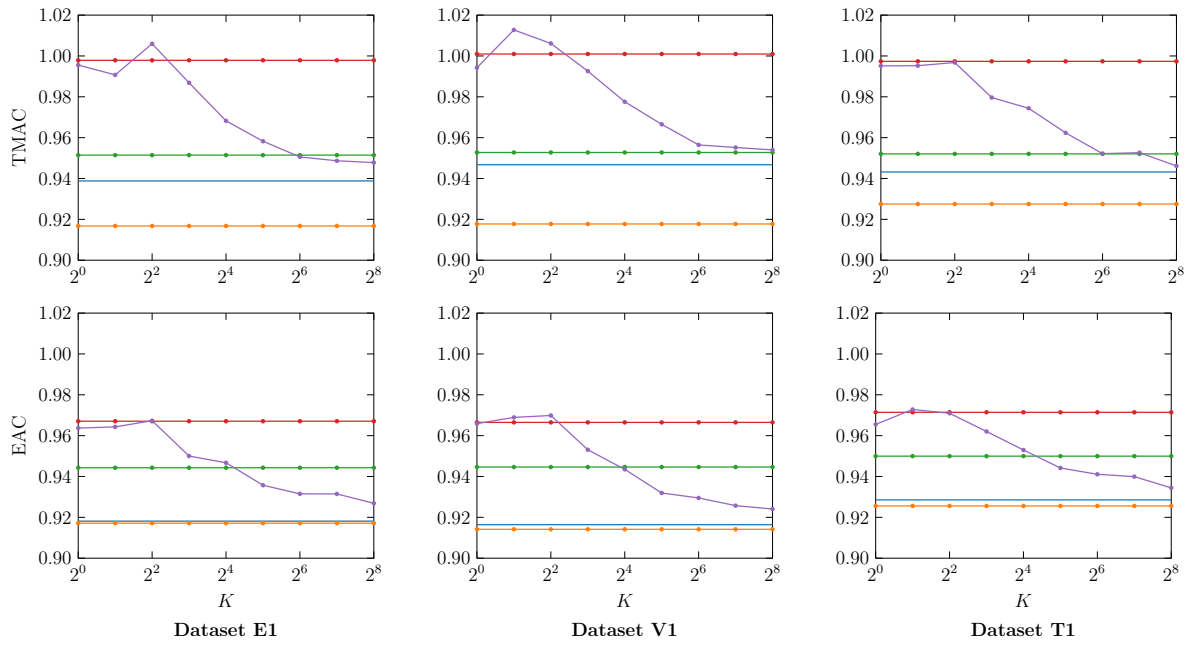


Figure 5.1: The tangential momentum accommodation coefficient (TMAC) and the energy accommodation coefficient (EAC) plotted against the number of Gaussian distributions K in the Gaussian mixture model (GMM) based scattering kernels. The displayed plots are for the datasets E1, V1 and T1 in Table 2.1. The accommodation coefficients obtained from the MD data are denoted with (—). Further, the accommodation coefficients for the velocity data generated by kernels is denoted as: Maxwell kernel (—), Maxwell-Yamamoto kernel (—), CLL kernel (—) and the GMM kernel (—).

As can be seen from Figure 5.1 the EACs and TMACs computed for the standard paramet-

ric kernels are not excellent; they generally deviate significantly from the accommodation coefficients computed from the molecular dynamics data. Note that although the Maxwell kernel appears to perform very well for the EACs this might be a bit of a deceiving figure since this accommodation coefficient is given directly as an input parameter. As expected, the performance of the GMM kernel improves when the number of components is increased. The value of the accommodation coefficients for this kernel moves closer towards the value of the molecular dynamics data it is fitting. As can be seen, for a large number of components the performance of the GMM kernel on the ACs gets better than the standard parametric kernels in general.

5.3 Kullback-Leibler divergence

Another possibility to quantify how well the generated outgoing velocity distributions match is by making use of some kind of divergence measure. Since the velocity distributions considered are continuous it is possible to use approximations for the generally applied Kullback-Leibler (KL) divergence [30]. The KL divergence between the continuous probability distributions P and Q that exist on $\mathbb{R}^d$ is defined as

$$D_{\mathrm{KL}}(P\|Q) \equiv \int_{\mathbb{R}^d} \mathrm{d}P \log \frac{\mathrm{d}P}{\mathrm{d}Q}. \tag{5.1}$$

In this case, since P and Q are reflective velocity distributions with three components, $d = 3$. To compute values for this divergence several approaches are possible. Since the outgoing velocity distributions produced in this report cannot be described exactly by an analytical formula for as far as known, working with probability densities based on samples from these distributions is a natural choice. The probability densities of the underlying distributions can be approximated in several ways which will be discussed below in Section 5.3.1 for estimates based on binning and in Section 5.3.3 for estimates based on k-Nearest Neighbors (k-NN).

5.3.1 Binning based estimates

The first method used is based on dividing the domain in which the distributions are defined in several bins. The probability density for distribution P in bin i can then be estimated as

$$\hat{p}_i = \frac{N_i}{\sum_i N_i}. \tag{5.2}$$

Hereafter, the computed probability density estimates $\hat{p}_i$ and $\hat{q}_i$ for all bins can be used to approximate the KL divergence as

$$\hat{D}_{\mathrm{KL}}(P\|Q) = \sum_i \hat{p}_i \log \frac{\hat{p}_i}{\hat{q}_i}. \tag{5.3}$$

In this report this method will be used to estimate the KL divergences between full outgoing probability distributions as well as projections of these distributions on different axis.

5.3.2 Component optimization

Computing approximations of KL divergences can also be used to optimize the number of components of the Gaussian mixture based scattering kernels. To compare the performance improvement of the Gaussian mixture based scattering kernels with the parametric kernels, Kullback-Leibler divergence ratios will be used. In the plots and tables below, the KL divergences of the parametric scattering kernels with the MD data will be divided by the KL divergence of the GMM based scattering kernel with the MD data. Hence, ratios are obtained were a value of zero indicates the performance of a kernel is as good as it can be and perfectly in line with the predictions from the MD data. A ratio between zero and one means better than the GMM kernel and a ratio above one means worse than the GMM kernel. The KL divergence ratio of the GMM kernel is one by definition. This makes it possible to easily compare the performance of the GMM kernels with the standard parametric kernels.

Below in Figure 5.2 the Kullback-Leibler divergence ratios computed by the binning procedure described above are shown for the full multidimensional outgoing velocity distributions and plotted for GMM scattering kernels with a varying number of components. The binning interval for the molecular velocities was chosen to be between $-1000\,\mathrm{m/s}$ and $1000\,\mathrm{m/s}$ for the tangential directions. For the normal direction, the binning interval was chosen to be between $0\,\mathrm{m/s}$ and $1000\,\mathrm{m/s}$. For each of the three directions, use was made of 3 bins which gives a total of $3^3 = 27$ three-dimensional bins for the full velocity distributions. The number of bins in each direction was kept low to ensure that even for the bins at the ends of the domain samples are available. Hence, there will be no divisions by zero in the KL divergence computation and the results do not have to be invalidated immediately. However, it should be noted that it is hard to exactly quantify how well the estimates for this method based on binning converge to the true value for these distributions. Although, one can still use them as a measure for making comparisons between distributions.

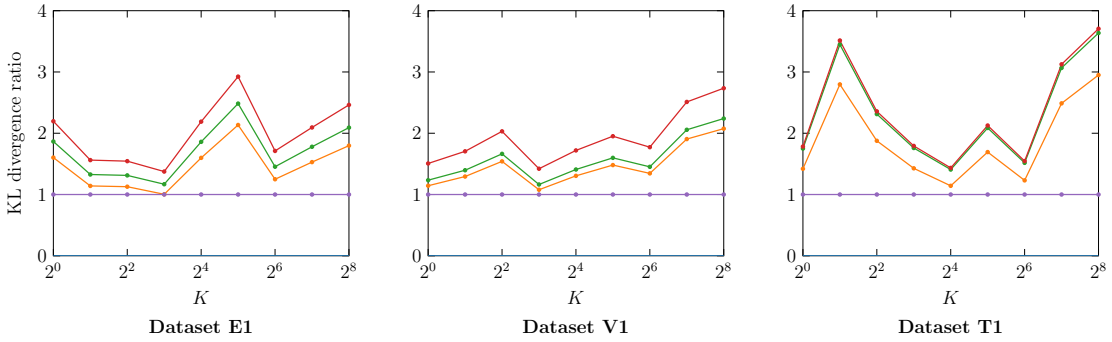


Figure 5.2: The Kullback-Leibler (KL) divergence ratios plotted against a varying number of components of the GMM kernel. The KL divergence of the MD data (—) is by definition zero and the KL divergence of the GMM based kernel (—) is by definition one. The parametric kernels are denoted with (—) for the Maxwell kernel, (—) for the Maxwell-Yamamoto kernel and (—) for the CLL kernel.

In Figure 5.2 it can be seen that although the KL divergence ratio estimates slightly increase for a larger number of components for the full outgoing velocity distributions, they do not vary a lot when the number of components of the Gaussian mixture models is increased. Taking this into account together with changes in the accommodation coefficients displayed in Figure 5.1 as well as computational requirements, 64 components were chosen for the Gaussian mixture models in the scattering kernels for the rest of the thesis. For this number of components the accommodation coefficients give results that are close to the molecular dynamics data as well as not being unworkable computationally expensive. The KL divergences shown in Figure 5.2 do not give a strong hint which number of components should be preferred.

5.3.3 k-NN based estimates

In addition, next to binning samples to get probability density estimates also knowledge of the nearest neighbors of the samples can be used for this purpose. The distance of sample X_i to its nearest neighbor $\{X_j\}_{j\neq i}$ in a set of samples $\{X_1, \ldots, X_n\}$ from distribution P can be computed as

$$\rho_n(i) = \min_{j=1,\ldots,n, j\neq i} \|X_i - X_j\| \tag{5.4}$$

with $\|\cdot\|$ the L^2 norm in $\mathbb{R}^d$. Furthermore, the distance from sample X_i to a sample Y_j in a set of samples $\{Y_1, \ldots, Y_m\}$ drawn from Q can be computed as

$$\nu_m(i) = \min_{j=1,\ldots,m} \|X_i - Y_j\| . \tag{5.5}$$

As proposed by [31], the latter two equations can then be used to approximate the KL divergence between probability distributions P and Q via

$$\hat{D}_{\mathrm{KL}}(P\|Q) = \frac{d}{n}\sum_{i=1}^{n} \log \frac{\nu_m(i)}{\rho_n(i)} + \log \frac{m}{n-1}. \tag{5.6}$$

This can be extended from only using nearest neighbors to using k-nearest neighbors which results in better convergence properties. Furthermore, using k-NN evades problems with identical samples from the outgoing velocity distributions. For the current datasets a number of neighbours $k = 128$ was chosen which was found to be sufficient to avoid problems with identical samples. Also, it proved to be relatively stable while not being computationally too expensive. Though, this divergence measure was only used on the full outgoing velocity distributions due to the computational complexity. It should be noted that also for this method it is hard to exactly quantify how well the estimates converge to the true value for these distributions.

5.3.4 Performance evaluation

In the tables below the KL divergence ratios for the standard parametric scattering kernels are computed via the same method as discussed in Section 5.3.2. As is also discussed

in Section 5.3.2 the number of components of the GMM scattering kernels will be fixed to 64 for these tables as well as the remainder of the thesis. In the tables below, binning estimates for the KL divergence are computed for the full outgoing velocity distributions as well as for the projections of these distributions on the Cartesian axis. For the projections on the Cartesian axis 30 bins are used for the density estimates. These bins lie between $-1000\,\mathrm{m/s}$ and $1000\,\mathrm{m/s}$ for the tangential directions. For the normal direction the binning interval was chosen to be between $0\,\mathrm{m/s}$ and $1000\,\mathrm{m/s}$. For the full velocity distributions, the same binning procedures as described in Section 5.3.2 are used. k-NN is only used on the full distribution due to the computational requirements.

Table 5.1: Rounded KL divergence ratios for the parametric scattering kernels on dataset E1.

Procedure →	Binning				k-NN
Kernel ↓ Data →	v_x	v_y	v_z	$\mathbf{v}$	$\mathbf{v}$
Maxwell kernel	2.46	5.92	1.41	1.25	1.79
Maxwell-Yamamoto kernel	1.32	7.58	1.11	1.45	2.59
CLL kernel	1.13	8.17	1.06	1.71	2.99

Table 5.2: Rounded KL divergence ratios for the parametric scattering kernels on dataset T1.

Procedure →	Binning				k-NN
Kernel ↓ Data →	v_x	v_y	v_z	$\mathbf{v}$	$\mathbf{v}$
Maxwell kernel	0.77	8.27	2.26	1.23	2.16
Maxwell-Yamamoto kernel	1.36	9.34	2.09	1.52	2.63
CLL kernel	1.31	10.53	2.94	1.55	2.70

Table 5.3: Rounded KL divergence ratios for the parametric scattering kernels on dataset V1.

Procedure →	Binning				k-NN
Kernel ↓ Data →	v_x	v_y	v_z	$\mathbf{v}$	$\mathbf{v}$
Maxwell kernel	1.00	7.43	0.90	1.33	1.18
Maxwell-Yamamoto kernel	0.92	8.25	0.89	1.44	1.78
CLL kernel	1.01	9.33	0.81	1.76	1.37

As can be seen from the tables above, the KL divergence indicates that the GMM based scattering kernels mainly improve modelling the distribution in the direction normal to the wall. For this direction the KL divergence ratios are sometimes even more than a factor ten. Interestingly, according to the KL divergences, the performance for the GMM kernels appears to be worse for the two directions tangential to the wall for several cases. Here, the computed divergence ratios are smaller than one. However, when looking at the values for the KL divergence estimates computed by binning as well as k-NN for the full velocity distributions it appears that GMM based scattering kernels still show a significant improvement upon the standard parametric ones.

5.4 Cartesian velocity distributions

In Figure 5.3 the projections of the outgoing velocity distributions on the Cartesian axes are plotted. The probability densities plotted are generated in the same fashion as for the probability densities used for the binning based KL divergence computations in the previous section. Here, 30 bins are used with the same cut-off velocities dependent on the projection axis. Hence, it is possible to directly compare these distribution visualizations with the KL divergences in the three tables of Section 5.3.4. As can be seen in Figure 5.3, for all the kernels the distributions appear to coincide with the data from the molecular dynamics simulations. Although, there are some subtle deviations. For instance, looking at the leftmost data point of the distributions of the normal directions it can be seen that the Maxwell, Maxwell-Yamamoto and CLL kernel persistently underpredict the probability density at this point. This can be seen as an explanation why the KL divergences for the normal directions are so much larger than the KL divergences of the tangential directions. Furthermore, also interesting to note is the deviation that can be seen between the tangential directions for the figures of dataset V1. As can be seen, there is a shift away from zero velocity in the peak of the distribution for the x-direction. This is due to the fact that the wall is moving in this direction.

5.5 Angular velocity distributions

In Figure 5.4 the angular velocity distributions for the outgoing velocities can be seen for the vertical angle γ and the horizontal deviation angle from the x-axis η_x. Also, the probability densities are approximated by means of binning samples here. For the vertical angle, 30 bins or one bin per three degrees were used to approximate the local probability density and therefrom construct the plots. For the horizontal deviation angle 36 bins or one bin per ten degrees were used. As can be seen from the plots of the vertical angle, deficiencies between the standard parametric scattering kernels and the molecular dynamics data are the biggest for an angle of approximately 45°. As can also be seen, the performance of the GMM kernels for the vertical angles is better than the performance of the standard scattering kernels. What is especially noticeable from the plots of the horizontal deviation angles is the shift of the distribution for the dataset V1 with the moving wall. It can clearly be seen that the velocities are biased towards the negative x-direction or a 180° horizontal deviation angle.

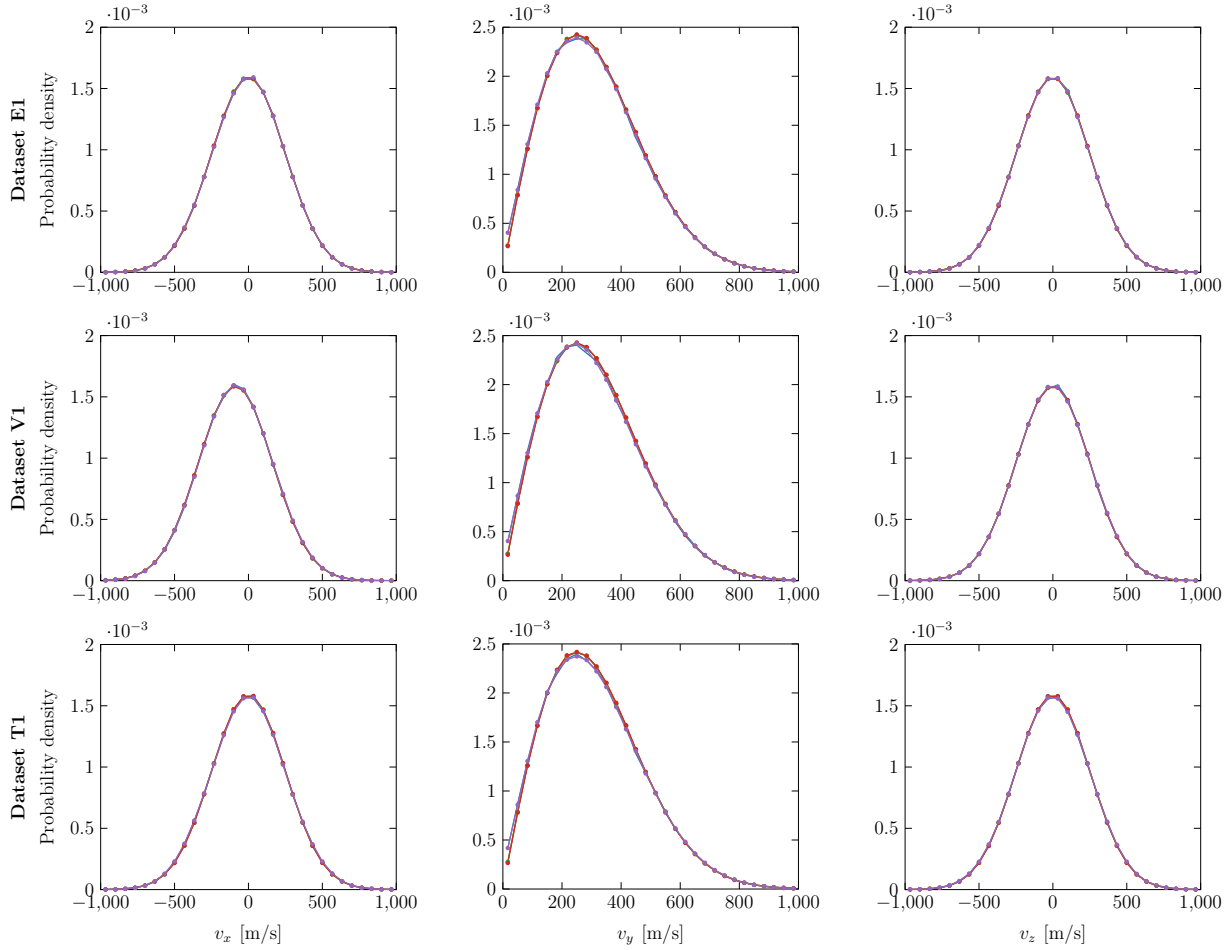
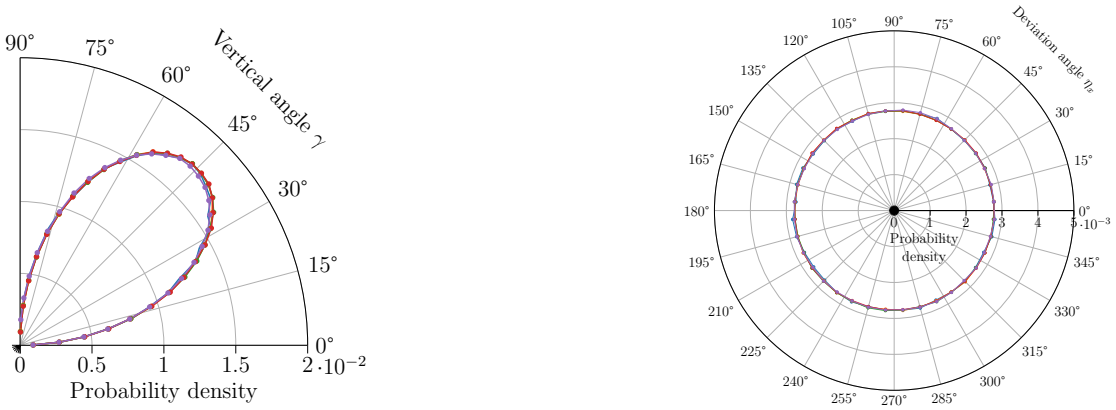
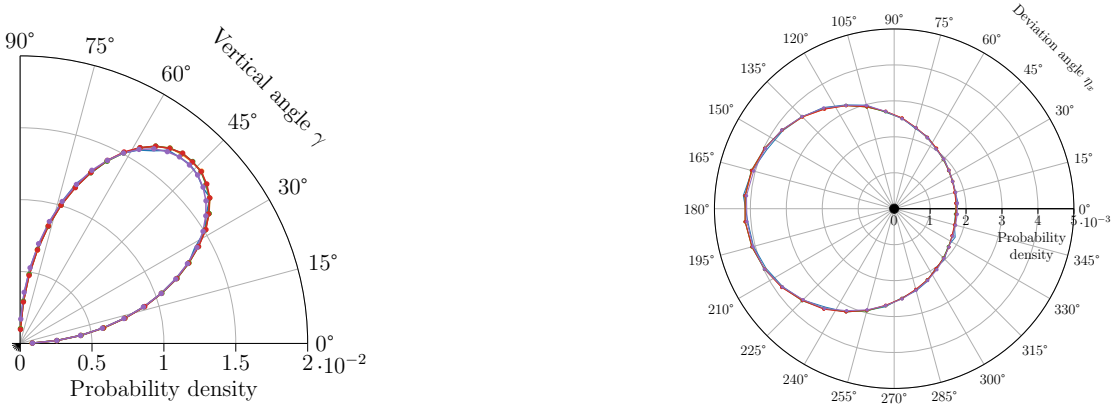


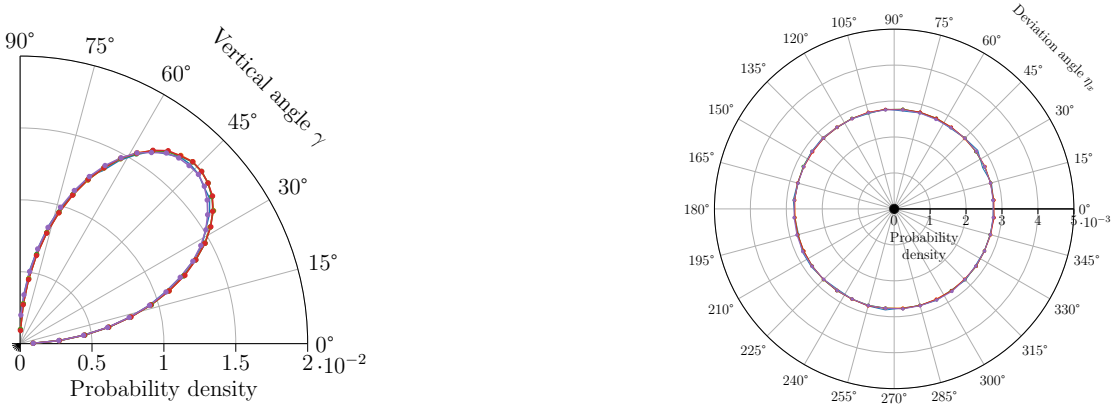
Figure 5.3: The velocity distributions projected on the Cartesian axis for dataset E1, V1 and T1 as well as the velocity distributions generated by the kernels. The lines denote the following data or kernel: MD data (—), Maxwell kernel (—), Maxwell-Yamamoto kernel (—), Cercignani-Lampis-Lord kernel (—) and the GMM kernel (—).



Dataset E1



Dataset V1



Dataset T1

Figure 5.4: The velocity distributions for the vertical angle γ as well as the horizontal deviation angle with the x-axis η_x for dataset E1, V1 and T1 with both the MD data and the scattering kernels. The lines denote the following data or kernel: MD data (—), Maxwell kernel (—), Maxwell-Yamamoto kernel (—), Cercignani-Lampis-Lord kernel (—) and the GMM kernel (—).

6. Lattice Boltzmann method

In this chapter the lattice Boltzmann method simulation technique will be further discussed. First, the fundamentals of the simulation technique along with its relation to the fundamental equations of fluid dynamics will be treated. Hereafter, the setup that will be used for tests in this thesis will be reviewed. This includes a discussion of the simulation of gas-wall interactions with the lattice Boltzmann method.

6.1 Underlying theory and equations

6.1.1 Navier-Stokes equations

When considering fluid dynamics problems that could be modelled by continuum mechanics, the first and foremost mathematical framework used to describe the underlying dynamics of a problem is the Navier-Stokes equations. The general vector form of these equations can be written as

$$\rho \frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} = \nabla \cdot \mathbb{P} + \mathbf{F}. \tag{6.1}$$

Here ρ denotes the local density of a fluid element, $\mathbf{u}$ the velocity vector of a fluid element and $\mathbf{F}$ an external force like gravity that might be acting on the fluid element that is currently being considered. $\mathbb{P}$ denotes a general stress tensor that normally contains pressure and viscosity related stress components. Furthermore, $\frac{\mathrm{d}}{\mathrm{d}t}$ denotes the total derivative. There are several simulation techniques available that are based on direct discretization of this equation. These simulation techniques can be classified as conventional computational fluid dynamics (CFD) and are most notably finite volume, element and difference methods. The Navier-Stokes equations unavoidably break down when the particulate nature of a fluid becomes more expressive. Since the equation used to describe the dynamics breaks down also the simulation techniques that are directly based on it start to show deficiencies. Therefore, these simulation techniques are not suitable to simulate rarefied gases, at least not up to a highly rarefied level.

In previous work it was found that in the slip flow regime of the Knudsen number, the resulting flow phenomena can still be modelled correctly by the Navier-Stokes equations as long as velocity slip at the boundaries is taken care of [32]. However, when the degree of rarefaction increases further and the flow enters the transition regime other techniques need to be used. The first possibility is applying corrections on the Navier-Stokes equations that include higher order approximations on the stress tensor and heat flux [33]. The second and most generic approach is by discarding the Navier-Stokes equations and resorting to kinetic theory to describe the underlying physics. This is the approach that will be taken in this thesis to make the developed boundary conditions as generally applicable as possible.

6.1.2 Kinetic theory

Between full macroscopic approaches based on the Navier-Stokes equations and full microscopic approaches that model the particulate nature of a flow directly, also mesoscopic approaches are available to model fluid dynamics problems. These approaches are particularly suitable for mildly rarefied conditions given that the focus is on descriptions for the distribution of particles in a gas rather than continuum descriptions or tracking individual molecules. The behaviour of distributions of particles is described by the framework of kinetic theory which is at the basis of all mesoscopic simulation techniques.

While kinetic theory can in principle be formulated for every kind of fluid it is usually applied on rarefied gases. To simplify the discussion, as with the rest of the thesis, only monatomic gases will be considered. This avoids the need of having to take vibrational and rotational degrees of freedom into account for the molecules. All changes of state can be written as a translational change in the position of the molecules. Therefore, it can be stated that all molecules in the monatomic gas that we consider collide elastically because no energy transfer is possible to other degrees of freedom. Kinetic theory is based on capturing this information on the state of a gas in a so-called particle distribution function which will here be denoted as $f(\mathbf{x}, \mathbf{u}, t)$. In this expression, $\mathbf{x}$ is a position vector, $\mathbf{u}$ a velocity vector and t the time. To express changes in the distribution of particles, the Boltzmann equation can be used which is an essential part of describing gases by means of mesoscopic descriptions.

6.1.3 Boltzmann equation

The Boltzmann equation in its general form can be written as

$$\frac{\partial f}{\partial t} + \mathbf{u} \cdot \frac{\partial f}{\partial \mathbf{x}} + \frac{\mathbf{F}}{\rho} \cdot \frac{\partial f}{\partial \mathbf{u}} = \Omega(f) \tag{6.2}$$

and can be seen as an advection equation for particle distributions [34]. In this equation, the symbols have their usual meaning. It is important to note, however, that the molecular velocity $\mathbf{u}$ in (6.2) is an independent variable. The first two terms on the left-hand side denote the material derivative of the particle distribution function. Therefore, these terms illustrate the time rate of change. The third term on the left-hand side describes the influence of external forces. On the right-hand side, $\Omega(f)$ is used to denote the total differential of the particle distributions. It is the source term which represents the local redistribution of f due to collisions. Therefore, the source term $\Omega(f)$ is often called the collision operator [35]. The Boltzmann equation forms the backbone of several simulation techniques from which the most notable are the Lattice Boltzmann Method (LBM), Direct Simulation Monte Carlo (DSMC) and the Method of Moments (MoM). How this equation relates to the lattice Boltzmann method will be discussed in the next section.

6.1.4 Lattice Boltzmann equation

The Boltzmann equation, (6.2), can be discretized in physical space, velocity space and time to find the lattice Boltzmann equation

$$f_i\left(\mathbf{x} + \mathbf{c}_i\Delta t, t + \Delta t\right) = f_i(\mathbf{x}, t) + \Omega_i(\mathbf{x}, t). \tag{6.3}$$

This equation expresses that particles $f_i(\mathbf{x}, t)$ move with velocity $\mathbf{c}_i$ to a neighbouring point $\mathbf{x} + \mathbf{c}_i\Delta t$ at the next time step $t + \Delta t$ [35]. Furthermore, at the same time particles are affected by a collision operator $\Omega_i(\mathbf{x}, t)$. The index i runs over all velocities in a velocity set $\mathbf{c}_i$ that form the velocity discretization. Given that position, velocity and time are discretized, this equation can be implemented on a lattice and so-called moments can be computed for every lattice node. The first two and most important moments are

$$\rho = \sum_i f_i \tag{6.4}$$

and

$$\rho\mathbf{u} = \sum_i \mathbf{c}_i f_i \tag{6.5}$$

which denote the mass and momentum density [36]. The space and time dependencies of the parameters are omitted for brevity. Furthermore, in case it is assumed that thermal fluctuations in a system are negligible it is possible to find a direct relation between the pressure, speed of sound and the density. This relation is

$$p = c_s^2\rho \tag{6.6}$$

with p the pressure in the system and c_s the speed of sound. For all general non multi-speed velocity sets $\mathbf{c}_i$ the speed of sound can be further defined as

$$c_s^2 = \frac{1}{3}\frac{\Delta x^2}{\Delta t^2}. \tag{6.7}$$

In this equation Δx is the spacing of the lattice nodes and Δt is the time step.

6.2 Lattice parameters

For the LBM simulation setup to be studied in this thesis, a Couette flow was chosen. Additionally, given that the goal of this thesis is to couple both simulation techniques, isothermal simulations were chosen as a proof of concept. This makes MD simulations E1, V1 and V2 suitable for usage and makes it possible to ignore temperature gradients in the lattice Boltzmann method. Hence, it is viable to apply temperature related simplifications to LBM. Given that the Knudsen number of the molecular dynamics simulations in this thesis is 0.01, they are in the slip flow regime. In this regime of the Knudsen number, rarefaction effects are mainly noticed at the boundaries were a no-slip condition can no

longer be seen as sufficient. However, in the bulk the continuum hypothesis imposed by the Navier-Stokes equations can still be considered to hold. Hence, it can be assumed that for accurately simulating a rarefied Couette flow rarefaction effects only need to be taken into account in the gas-wall interaction. Also, this enables various simplifications in the used LBM setup. For the bulk, only the moments of the weights w_i up to the fifth order need to be isotropic on the used lattice [37]. Therefore, if we would like to simulate a Couette flow in two dimensions, using a D2Q9 velocity set is sufficient. This velocity set can be displayed as

$$\mathbf{c}_i = \begin{cases} \mathbf{c}_0 & = (0,0)c \\ \mathbf{c}_1, \mathbf{c}_2, \mathbf{c}_3, \mathbf{c}_4 & = (1,0)c, (0,1)c, (-1,0)c, (0,-1)c \\ \mathbf{c}_5, \mathbf{c}_6, \mathbf{c}_7, \mathbf{c}_8 & = (1,1)c, (-1,1)c, (-1,-1)c, (1,-1)c \end{cases} \tag{6.8}$$

with c the lattice constant equal to $\Delta x/\Delta t$. As a result, the velocities of a single lattice Boltzmann method node can be graphically displayed as is shown in Figure 6.1.

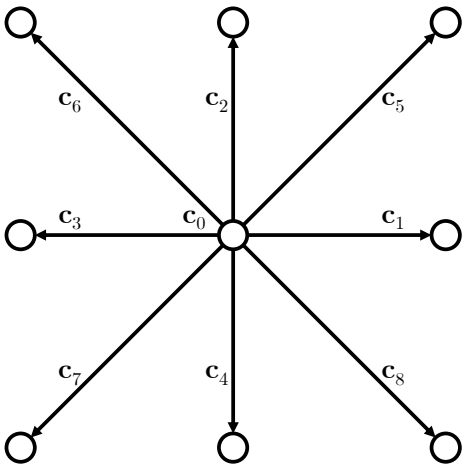


Figure 6.1: The discrete lattice velocity vectors of a D2Q9 lattice.

Furthermore, the weights of this velocity set can be written as

$$w_i = \begin{cases} w_0 & = 4/9 \\ w_1, w_2, w_3, w_4 & = 1/9 \\ w_5, w_6, w_7, w_8 & = 1/36 \end{cases} \tag{6.9}$$

for the discrete lattice velocities. For a Couette flow simulation, pressure fluctuations in the system should be almost non-existent. Consequently, it is possible to make use of the simplest collision operator that still allows for the reconstruction of the solution of the Navier-Stokes equations in the bulk. This is the Bhatnagar-Gross-Krook (BGK) collision operator

$$\Omega_i = -\frac{\Delta t}{\tau}\left(f_i - f_i^{\mathrm{eq}}\right) \tag{6.10}$$

with τ the relaxation time. For the slightly compressible case, the second order equilibrium populations can be computed as

$$f_i^{\mathrm{eq}} = w_i \rho \left(1 + \frac{3}{c^2} \left(\mathbf{u} \cdot \mathbf{c}_i \right) + \frac{9}{2c^4} \left(\mathbf{u} \cdot \mathbf{c}_i \right)^2 - \frac{3}{2c^2} \left(\mathbf{u} \cdot \mathbf{u} \right) \right) \tag{6.11}$$

for this collision operator [38].

6.3 Simulation matching

To simulate Couette flows with the lattice Boltzmann method, several parameters that are used internally by the lattice Boltzmann simulations need to be matched to ensure correct behaviour. To do this, first it is useful to define dimensionless parameters to be used in the simulations. These parameters can then be mapped on the correct physical units by means of conversion factors. In this thesis, these dimensionless parameters will be denoted with a star ($\star$). Therefore, the generally applicable lattice units for the dimensionless length scale, the dimensionless timescale as well as the initial dimensionless density of the system can be defined as

$$\Delta x^\star = 1, \quad \Delta t^\star = 1, \quad \rho_0^\star = 1. \tag{6.12}$$

Next, it can be stated that the lattice velocity c should be set equal to the root-mean-square velocity of an ideal gas

$$v_{\mathrm{rms}} = \sqrt{\frac{3kT}{m}} \tag{6.13}$$

for a D2Q9 lattice with a BGK collision operator [38]. This makes it possible to directly relate the Knudsen number to the dimensionless relaxation time of the lattice as

$$\tau^\star = \frac{\tau}{\Delta t} = \sqrt{\frac{3\pi}{8}} \mathrm{Kn} N_{\Delta x^\star} + \frac{1}{2} \tag{6.14}$$

with $N_{\Delta x^\star}$ the number of dimensionless length scales in the characteristic length direction between the walls [39, 40]. This parameter for the number of length scales, $N_{\Delta x^\star}$, is also often called the lattice number. With this information, a complete set of conversion factors between the LBM simulations and physical unit systems can be generated as

$$C_x = \frac{H}{N_{\Delta x^\star}}, \tag{6.15}$$

$$C_t = \frac{C_x}{v_{\mathrm{rms}}}, \tag{6.16}$$

$$C_\rho = \rho_0. \tag{6.17}$$

In this equation, H is the channel size in the lateral direction in the Couette flow simulations and ρ_0 is a reference density. These conversion factors for the length, time and density should be multiplied by the corresponding quantities in the lattice to obtain the result in the desired physical unit system.

6.4 Boundary conditions

To apply a lattice Boltzmann boundary condition, more populations need to be set than the available macroscopic variables. Since there are more parameters to be set than the amount of equations available, this gives rise to a non-uniqueness problem which is evidenced by the large amount of lattice Boltzmann boundary schemes available [35]. As stated above, a Couette flow will be studied. First, to simplify the discussion it is convenient to make use of straight boundaries aligned with the lattice nodes. This makes it obvious to solve this problem in the direction parallel to the walls by making use of periodic boundaries. However, this still leaves the boundary condition problem at the location of the walls open. Given that the boundary conditions at these walls should be able to cope with rarefied conditions, and therefore with velocity slip, further complications arise.

Before digging deeper into the available boundary conditions, first, it should be noted that two big families can be distinguished in the large amount of lattice Boltzmann boundary conditions available. These are *link-wise* and *wet-node* boundary conditions. As the name suggests, link-wise boundary conditions apply the boundary on a lattice link between two nodes. Wet-node boundary conditions work with nodes that are partially in the fluid domain and partially part of a boundary. Generally speaking, wet-node boundary conditions allow for more variation in the procedure applied since they have more degrees of freedom than link-wise boundary conditions. Furthermore, this also means that they can be more accurate than link-wise boundary conditions. Hence, all the boundary conditions considered below will be wet-node boundary conditions.

Second, it should be noted that the boundary conditions discussed below can be written in a kernel formulation. Using such a scattering kernel notation for a boundary condition will make a comparison with the coupling algorithm discussed in Chapter 7 easier. As proposed by Sbragaglia and Succi [41], it is possible to write a LBM boundary condition as a discrete scattering kernel on the populations as

$$f_j^{\text{out}} = \sum_i \mathcal{K}_{j,i} f_i^{\text{in}}. \tag{6.18}$$

In this equation, f_j^{out} is a vector of the outgoing populations from a wall node that are being injected back into the fluid domain based on the incoming populations f_i^{in}. The indexes i and j stand for the particular in- and outgoing populations respectively. $\mathcal{K}_{j,i}$ is the discrete analogue of a boundary scattering kernel that captures the interaction of a gas with a wall. To ensure mass and normal momentum conservation, all mass of an ingoing population should be redistributed over all outgoing populations. Therefore, the following should hold:

$$\sum_j \mathcal{K}_{j,i} = 1. \tag{6.19}$$

To simplify the discussion somewhat, the focus will be on stationary boundary conditions for the south wall on the used D2Q9 lattice. For a wall node on this wall, there are

three populations incoming and three populations outgoing. As depicted in Figure 6.2 the incoming populations are $f_7(x+\Delta x, y+\Delta y)$, $f_4(x, y+\Delta y)$ and $f_8(x-\Delta x, y+\Delta y)$. Here, the argument between brackets denotes the location of the lattice node from which the population originates. These populations should be mapped on the outgoing populations $f_5(x,y)$, $f_2(x,y)$ and $f_6(x,y)$. It is possible to write this mapping in a discrete scattering kernel formulation as

$$\begin{pmatrix} f_5(x,y) \\ f_2(x,y) \\ f_6(x,y) \end{pmatrix} = \mathcal{K} \begin{pmatrix} f_7(x+\Delta x, y+\Delta y) \\ f_4(x, y+\Delta y) \\ f_8(x-\Delta x, y+\Delta y) \end{pmatrix} \tag{6.20}$$

for the depicted node. Here, the scattering kernel $\mathcal{K}$ is a 3×3 matrix. This scattering kernel should be applied on the node's populations after the streaming step as an alternative collision operation for the wall node.

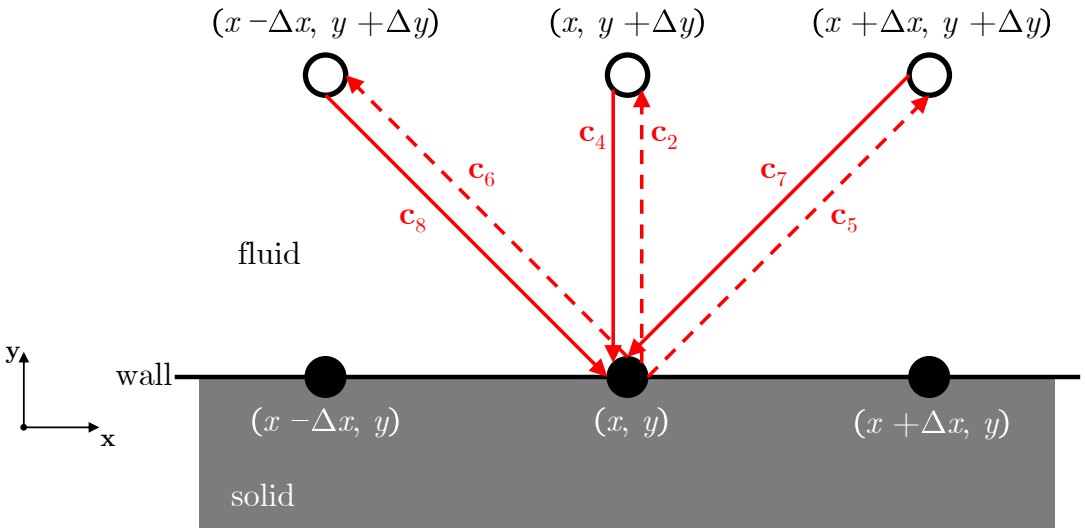


Figure 6.2: A depiction of the velocity vectors entering and exiting a boundary node on the south side of the domain.

Finally, it should be stated that the boundary conditions formulated here only work for stationary walls in the current form. This is of course not in agreement with the moving walls considered in a Couette flow. To resolve this issue, population transformations to ensure correct scattering can be used. The transformations discussed here can be seen as a general transformation to move the populations from the stationary inertial system of the lattice to the moving inertial system of the wall. Furthermore, while they are applied here on the south wall of a D2Q9 lattice the concepts are generally applicable. First, for the particular case considered, it can be shown that the mass and tangential momentum equations for the fluid at the walls are

$$f_8 + f_4 + f_7 = \rho, \qquad f_8 - f_7 = \rho u_x^{\text{fluid}}, \tag{6.21}$$

as in accordance with (6.4) and (6.5) for mass and momentum conservation respectively. The superscript *fluid* denotes the populations of the wall node that are in the fluid domain. The locational arguments of the populations as shown in (6.20) are omitted for brevity. Only the populations that are adding tangential momentum to the wall need to be transformed. For the particular case of the wall and the lattice being considered, this means that population f_7 and f_8 need to be changed as shown in Figure 6.2. Besides, the following momentum equation in the inertial system of the moving wall should hold:

$$f_8^{\mathrm{t}} - f_7^{\mathrm{t}} = \rho \left(u_x^{\mathrm{fluid}} - u_x^{\mathrm{wall}} \right). \tag{6.22}$$

Here, the superscript t denotes the transformed populations that are in the inertial system of the wall. Since f_4 does not influence the velocity of the fluid at the wall in the tangential direction, it is not changed when doing the transformation. Therefore, after combining (6.21) and (6.22) the following expressions for the transformed populations can be obtained:

$$f_7^{\mathrm{t}} = \frac{1}{2}\rho \left(u_x^{\mathrm{wall}} - u_x^{\mathrm{fluid}} + 1 \right) - \frac{1}{2} f_4, \tag{6.23}$$

$$f_4^{\mathrm{t}} = f_4, \tag{6.24}$$

$$f_8^{\mathrm{t}} = \frac{1}{2}\rho \left(u_x^{\mathrm{fluid}} - u_x^{\mathrm{wall}} + 1 \right) - \frac{1}{2} f_4. \tag{6.25}$$

After these transformations, a scattering kernel can be applied to obtain the transformed post-scattered populations. To transform these populations back to the inertial system of the lattice, the mass movement in the transformation step is undone again. For the current situation this can be shown as

$$f_5 = f_5^{\mathrm{t}} + (f_7^{\mathrm{t}} - f_7), \tag{6.26}$$

$$f_2 = f_2^{\mathrm{t}}, \tag{6.27}$$

$$f_6 = f_6^{\mathrm{t}} + (f_8^{\mathrm{t}} - f_8), \tag{6.28}$$

for the populations being injected back into the fluid domain. In this equation f_5^{t}, f_2^{t} and f_6^{t} are populations computed from scattering the incoming transformed populations with a scattering kernel. Naturally, care needs to be taken that the populations considered never become negative during any of these steps.

In the sections below, the LBM boundary conditions that will be used in this report will be considered. While there exists a large amount of LBM boundary conditions, focus is in particular on boundary conditions that allow for rarefaction effects. Furthermore, also the possibility of writing a kernel formulation for a particular boundary condition was taken into account.

6.4.1 Slip-reflection kernel

The first class of rarefied boundary conditions that can be distinguished are models that make use of a combination of bounce-back and specular reflection. As was shown analytically by [42], using a bounce-back condition to model a wall results in zero velocity slip.

Furthermore, specular reflected populations give rise to a free-slip boundary condition since there is no momentum exchange with a wall. Therefore, it is possible to combine bounce-back with specular reflection to get a partial slip boundary condition as was proposed by Succi [43]. This has been further related to the accommodation coefficients by Sbragaglia and Succi [41] and can be written for the considered south wall on a D2Q9 lattice as

$$f_5 = \frac{1}{2}\alpha f_7 + \left(1 - \frac{1}{2}\alpha\right) f_8, \tag{6.29}$$

$$f_2 = f_4, \tag{6.30}$$

$$f_6 = \frac{1}{2}\alpha f_8 + \left(1 - \frac{1}{2}\alpha\right) f_7, \tag{6.31}$$

where the node locations are omitted for brevity and usage is made of the EAC as an accommodation coefficient. Additionally, this can be written in terms of a slip-reflection kernel formulation as

$$\mathcal{K} = \begin{pmatrix} \frac{1}{2}\alpha & 0 & 1 - \frac{1}{2}\alpha \\ 0 & 1 & 0 \\ 1 - \frac{1}{2}\alpha & 0 & \frac{1}{2}\alpha \end{pmatrix} \tag{6.32}$$

for usage in (6.20).

6.4.2 Maxwell kernel

The second type of boundary condition that will be considered is a boundary condition which is directly based on the well known Maxwell kernel from continuous kinetic theory. The Maxwell kernel can be written as a boundary condition in the lattice Boltzmann method as

$$f_j^{\text{out}} = (1-\alpha) f_i^{\text{in}} \left|\mathbf{c}_i \cdot \mathbf{n}\right| + \alpha \frac{\sum_{\mathbf{c}_k \cdot \mathbf{n} < 0} \left|\mathbf{c}_k \cdot \mathbf{n}\right| f_k}{\sum_{\mathbf{c}_k \cdot \mathbf{n} > 0} \left|\mathbf{c}_k \cdot \mathbf{n}\right| f_k^{\text{eq}}} \times f_j^{\text{eq}} \tag{6.33}$$

for a stationary wall [39, 44]. As with the slip-reflection kernel, (6.20), also here the EAC is used as an accommodation coefficient. Equation (6.33) for the Maxwell kernel can be written as a discrete scattering kernel by making use of an adjustment to the slip-reflection-accommodation model proposed by Sbragaglia and Succi [41]. Since bounce-back is not incorporated in the model discussed here, the kernel for the south wall can be written as

$$\mathcal{K} = \begin{pmatrix} \alpha W_2 & \alpha W_2 & 1 - \alpha + \alpha W_2) \\ \alpha W_1 & 1 - \alpha(1 - W_1) & \alpha W_1 \\ 1 - \alpha(1 - W_2) & \alpha W_2 & \alpha W_2 \end{pmatrix} \tag{6.34}$$

with $W_1 = 2/3$ and $W_2 = 1/6$. The presence of these weights is due to the fact that this kernel is based on the discrete analog of the perfect accommodation kernel. Here, perfectly accommodated means that the scattered particles have an uniform Maxwell distribution with the temperature of the wall.

7. Coupling algorithm

In Chapter 7 a coupling algorithm for the molecular dynamics data and the lattice Boltzmann method will be proposed. First, a discrete scattering kernel for the lattice Boltzmann method based on trained Gaussian mixture based scattering kernels will be further elaborated upon. Second, matching this kernel to the state of a wall node in lattice Boltzmann method simulations for interpolation purposes will be discussed. This chapter is concluded by a discussion on manipulation of kernels and interpolation options.

7.1 Coupling procedure

To improve upon the performance of the boundary conditions debated upon in Section 6.4, one possibility is to make use of more information than a single accommodation coefficient. This is what will be discussed in this section, transferring information on the scattering from molecular dynamics data directly into a scattering kernel that can be used by the lattice Boltzmann method. The coupling procedure which is described here could be trivially extended to higher dimensions, different velocity sets and multi-speed lattices in full generality. However, for simplicity the same case of a south wall on a D2Q9 lattice, as discussed in Section 6.4, will be considered to further illustrate the method.

The coupling procedure can be split in two steps from which the first one is to construct a continuous scattering kernel that can be used directly in particle based simulations. This is necessary since finite probability densities need to be computed for the scattering dynamics. As has been extensively discussed in Chapter 4 scattering kernels based on Gaussian mixture models are an excellent candidate for this purpose. Their large number of tunable parameters allows for taking into account the coarse scattering dynamics as well as more subtle scattering behaviour. This has been further illustrated in Chapter 5 where it was shown that the performance of these kernels generally improves over several well known standard parametric kernels. Furthermore, values for the learned probability density function can be computed with ease.

The second step is to cast this continuous scattering kernel into a discrete kernel that can be used in a procedure as described in Section 6.4. The entries of this discrete scattering kernel $\mathcal{K}$ denote the probabilities of a particle of population i to be bounced in the direction and velocity of population j. The Gaussian mixture based scattering kernel $P(\mathbf{v}' \to \mathbf{v})$ denotes the probability density function of particles to be scattered in a particular direction and velocity. If it can be assumed that $P(\mathbf{v}' \to \mathbf{v})$ is continuous, the ratio of probabilities of a molecule to be scattered in one or another discrete direction can be written as the ratio of the values of the probability density function. For the scattering kernel based on a Gaussian mixture we have a guarantee that the probability density function is continuous since it is a superposition of multivariate Gaussian distributions. Hence, for the Gaussian mixture based scattering kernel it is possible to use the values of its probability density

function directly as matrix entries for $\mathcal{K}$ as long as mass conservation given by (6.19) is ensured.

This can also be shown by directly discretizing (3.3). First, it needs to be ensured that the integral on the right-hand side runs over a continuous function. Since this is the case with GMM based kernels it is possible to discretize this integral as a summation over different values of the incoming velocity distribution function and associated probability density given by the scattering kernel. Therefore, this equation reduces to

$$f_j^{\text{out}} = \sum_{\mathbf{c}_i \cdot \mathbf{n} < 0} \lambda_i P\left(\mathbf{c}_i \to \mathbf{c}_j\right) f_i^{\text{in}} \tag{7.1}$$

where the normalization factors from the normal velocities that should ensure mass conservation are replaced by λ_i. For computing multiple outgoing populations (7.1) can be casted in a scattering kernel formulation as long as mass conservation parameters are appropriately scaled.

For the case of a south wall on a D2Q9 lattice this enables writing the discrete scattering kernel as

$$\mathcal{K} = \begin{pmatrix} \lambda_7 P\left(\mathbf{c}_7 \to \mathbf{c}_5\right) & \lambda_4 P\left(\mathbf{c}_4 \to \mathbf{c}_5\right) & \lambda_8 P\left(\mathbf{c}_8 \to \mathbf{c}_5\right) \\ \lambda_7 P\left(\mathbf{c}_7 \to \mathbf{c}_2\right) & \lambda_4 P\left(\mathbf{c}_4 \to \mathbf{c}_2\right) & \lambda_8 P\left(\mathbf{c}_8 \to \mathbf{c}_2\right) \\ \lambda_7 P\left(\mathbf{c}_7 \to \mathbf{c}_6\right) & \lambda_4 P\left(\mathbf{c}_4 \to \mathbf{c}_6\right) & \lambda_8 P\left(\mathbf{c}_8 \to \mathbf{c}_6\right) \end{pmatrix} \tag{7.2}$$

for usage in (6.20). In this discrete scattering kernel, $\lambda_{7,4,8}$ are constants that ensure mass conservation and are equal to one divided by the sum of the probability densities in a column. This safeguards that the columns sum to one as enforced by condition (6.19). Therefore, mass conservation is guaranteed given that the mass of all incoming populations is redistributed over all outgoing ones. It should be noted that the probability densities in (7.2) should be computed with the parameters in non-lattice units. The lattice constant c, to be used in the probability density function computations, should be set equal to the root-mean-square velocity in (6.13) for this D2Q9 lattice.

7.2 Kernel matching

As is shown in (7.2), a discrete scattering kernel can be computed directly from a fitted Gaussian mixture for application in (6.20). For the results in Chapter 8 this will be done for the kernel from equilibrium simulation E1. This discrete scattering kernel will be fed directly into LBM to be used as a boundary condition for the Couette flow simulations. However, this approach does not allow for adaption of a kernel based on a particular flow situation at a boundary. For this, other approaches need to be followed.

A kernel can be matched to a specific situation at a wall only for a very limited number of situations where there is an equivalent molecular dynamics simulation available. For the lattice Boltzmann method simulations in this thesis, temperature gradients will not be considered. Of course thermal effects can be considered as well, but for a proof of concept

this will likely not give added value. Therefore, only velocity gradients between the fluid and the wall will be taken into account where use can be made of MD simulations E1, V1 and V2. This implies that missing kernels for a particular flow situation at a wall should be computed by means of interpolation which will be discussed in Section 7.4. Though, before this can be further quantified, velocity gradients in LBM first need to be linked to the molecular dynamics simulations.

For the molecular dynamics datasets in this thesis the velocity of the top wall is always set equal to the inverse of the velocity of the bottom wall. Furthermore, for the molecular dynamics simulations in which there is no temperature difference between the walls it can trivially be stated that both walls are equivalent. Accordingly, this implies that no matter how large the slip velocity of the molecules at the walls is, statistically speaking, there is no drift of the particles in the bulk. Therefore, it can be stated that the velocity difference between the bulk of the fluid and both walls in the MD simulations is equal to the wall velocity u_w when looking at the fluid from a macroscopic level. Furthermore, when looking at the system from a macroscopic level it can be assumed that the distance between the wall and the virtual border as well as the bulk of the fluid is negligibly small. Therefore, for macroscopic length scales the behaviour of the gas at the wall can be captured in a single LBM wet-node. As an extension of this, for the interpolation based kernels the following is assumed: when the velocity difference between the fluid and a wall is known at an LBM wet-node boundary condition, it can be matched with a scattering kernel for a molecular dynamics simulation with the same underlying wall velocity.

7.3 Kernel manipulation

As stated above, the molecular dynamics datasets E1, V1 and V2 are suitable for the LBM simulations in this thesis. Given that dataset E1 comes from a simulation that is in full equilibrium it can be concluded that the symmetry condition

$$P^{\mathrm{eq}}\left(\mathbf{v}' \rightarrow \mathbf{v}\right) = P^{\mathrm{eq}}\left(\mathbf{v} \rightarrow \mathbf{v}'\right) \tag{7.3}$$

should hold on the probability distribution function. Although, it cannot be guaranteed that this also holds for the scattering kernel learned by the Gaussian mixture. Consequently, for the discrete scattering kernel that is obtained from this simulation it is useful to explicitly enforce this by applying

$$\mathcal{K}^{\mathrm{eq}}_{j,i} = \frac{\mathcal{K}_{j,i} + \mathcal{K}_{m-j+1,m-i+1}}{2} \tag{7.4}$$

as a symmetrization procedure for the discrete scattering kernel. In this equation, the subscripts denote the matrix entries. Further, $\mathcal{K}_{j,i}$ can be a general kernel of size $m \times m$ where $m = 3$ for the D2Q9 case. Equation (7.4) ensures kernel point symmetry around the entry $\left(\frac{m+1}{2}, \frac{m+1}{2}\right)$. After this procedure proper mass conservation should be imposed again by linearly scaling the entries of all columns to make sure (6.19) holds.

Second, by negating the value of the wall velocity an equivalent discrete scattering kernel can be obtained for the datasets V1 and V2. This can be done by manipulating the discrete scattering kernel by applying

$$\mathcal{K}_{j,i} = \mathcal{K}_{m-j+1,m-i+1} \tag{7.5}$$

on the full kernel matrix. This equation inverts the scattering kernel in terms of the velocity, entries are swapped as such that the kernel holds for the inverted velocity. Thus, two extra discrete scattering kernels with corresponding wall velocities are acquired for usage in the LBM simulations.

7.4 Kernel interpolation

With five discrete kernels, for five discrete velocities, interpolation between the acquired kernels can finally be discussed. For the results in this thesis, two types of interpolation are used which are step wise linear and quadratic interpolation. Both types work separately for every individual entry of the matrices. For the step wise linear interpolation a velocity difference between the fluid and the wall which is between the obtained discrete scattering kernels is interpolated linearly. To this aim, the entries of the two kernels that are the closest are used. The quadratic interpolation works in an equivalent fashion. Nevertheless, for quadratic interpolation three kernels are used to compute the missing entries. Naturally, mass conservation as given by (6.19) is enforced everywhere.

8. Hybrid algorithm results

In this chapter the performance of the discrete scattering kernels coupled with the molecular dynamics data will be compared with two traditional kernels to model rarefied gases in lattice Boltzmann method simulations. First, the computed velocity profiles will be investigated. Hereafter, the velocity slip between the gas and the walls will be examined.

8.1 Velocity profiles

In this chapter the performance of the discrete scattering kernels discussed in Section 6.4 along with the kernels constructed from the coupled kernels in Chapter 7 will be further investigated. Given that isothermal LBM simulations are being examined, results need to be compared by means of velocity differences between the simulations. Here, special attention is given to the slip velocities at the wall. Nonetheless, to get a general idea of the performance of the developed MD kernels first the full generated velocity profiles are delved into. In Figure 8.1, the normalized velocity profiles for the Couette flow simulations for these kernels are shown. Note that the Couette flow studied here is different from the Couette flow studied for the molecular dynamics simulations. Here, one of the walls has zero velocity while the other wall has a velocity u_w. For the constructed velocity profiles, a total of 32 LBM nodes is used in the lateral direction. Given that the dimensionless relaxation time for every simulation is computed by means of (6.14) this gives $\tau^\star \approx 0.84$. The wall velocity was set to a low value of $u_w = 0.001$ in lattice units to get maximum accuracy as well as a guarantee on stability. However, given that both axis in Figure 8.1 are normalized the result is not dependent on the wall velocity and the LBM simulations will produce the same plots as long as stability criteria are not violated. In these plots, the vertical axis denotes the fluid velocity divided by the wall velocity. Besides, the horizontal axis denotes the normalized location in the lateral direction. The LBM simulations were stopped after convergence was reached. Here, the simulation is considered converged when the following inequality holds,

$$|\mathbf{u}(\mathbf{x}, t) - \mathbf{u}(\mathbf{x}, t-1)| < 10^{-12}, \tag{8.1}$$

for every lattice node in lattice units. Further, the outputs of the LBM simulations are plotted next to the analytical solution of the Navier-Stokes equations for a Couette flow with velocity slip at the boundaries. This analytical solution is given as

$$\frac{u}{u_w} = \frac{\frac{x}{H} + \mathrm{Kn}\frac{2-\alpha}{\alpha}}{1 + 2\mathrm{Kn}\frac{2-\alpha}{\alpha}} \tag{8.2}$$

with both the fluid velocity u/u_w on the left-hand side and the channel location in the lateral direction x/H on the right-hand side made non-dimensional and normalized [45]. The EAC that is used in this equation as well as the traditional LBM boundary conditions is computed from MD dataset E1. While strict equilibrium is not the case with moving

walls, the LBM simulations are close to equilibrium which makes it viable to assume that the EAC does not change a lot. This can be further substantiated by comparing with the EACs computed from dataset V1 and V2. For these MD simulations with increasing wall velocities the EACs remain close to the EAC computed from E1 with differences being in the order of 10^{-3}.

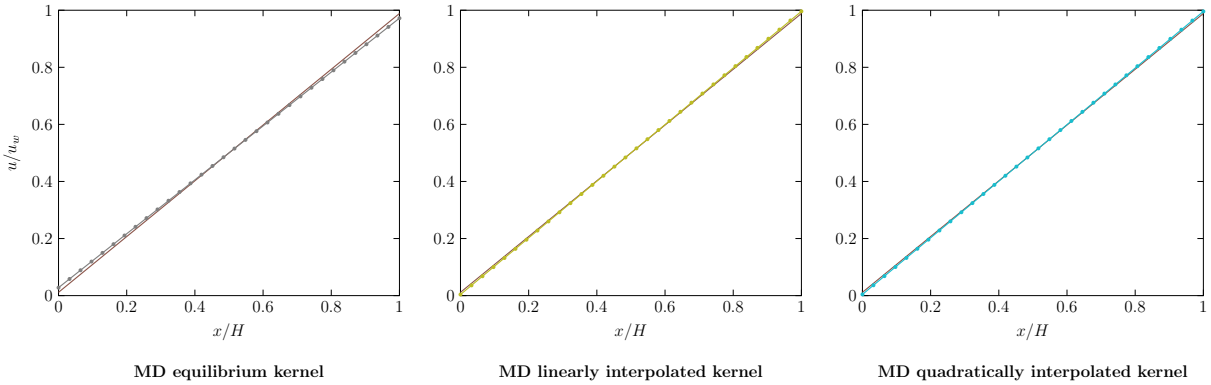


Figure 8.1: The dimensionless velocity profiles generated by the lattice Boltzmann method (LBM) simulations. The analytical solution is plotted with (——). The other lines are the following kernels based on the molecular dynamics data: the equilibrium kernel (—●—), the linearly interpolated kernel (—●—) and the quadratically interpolated kernel (—●—). The dots denote the lateral location of the lattice nodes.

As can be seen from Figure 8.1, the velocity profiles for all kernels used in the LBM simulations show a reasonable overlay with the analytical velocity profile. The velocity profile in the bulk should follow a straight line given that the behaviour of the gas in the bulk can be described by the Navier-Stokes equations. Therefore, the deviation between the profiles can be fully quantified by looking at the velocity slip near the walls. An illustration of this velocity slip can be seen in Figure 8.2. Here, the velocity profile at the wall is zoomed in on.

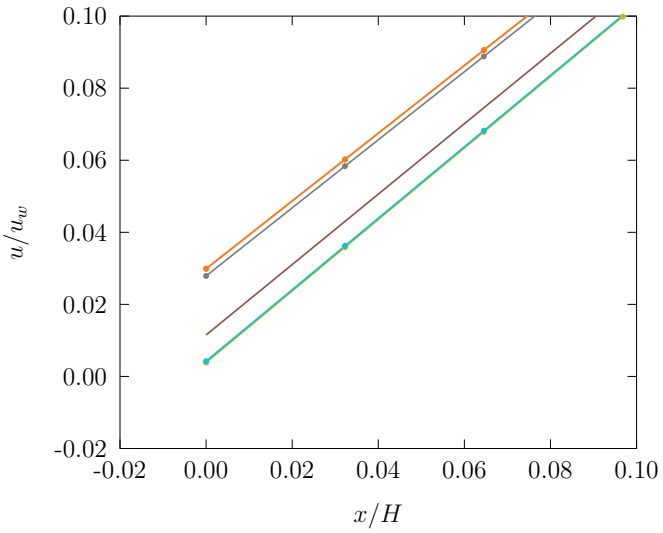


Figure 8.2: An inset zooming of the difference in the velocity profiles near the wall displayed in Figure 8.1. In addition to the kernels based on the MD data, also the velocity profiles for the traditional kernels are displayed. The lines denote the following LBM gas-wall interaction kernels: (—●—) is the slip-reflection kernel, (—●—) is the Maxwell kernel, (—●—) is the kernel based on MD simulation E1, (—●—) is the kernel based on MD simulations E1, V1 and V2 with linear interpolation and (—●—) is the kernel based on MD simulations E1, V1 and V2 with quadratic interpolation. Note that the lines for the slip-reflection and the Maxwell kernel lie on top of each other.

It can clearly be seen from Figure 8.2 that the kernels do not exactly predict the analytical solution for this number of LBM nodes in the lateral direction. How the solutions converge when the number of LBM nodes is increased will be studied in the next section where the convergence behaviour will be inspected.

8.2 Convergence behaviour

To compare the slip velocities between the LBM simulations of a Couette flow, use is made of the fluid velocity at the nodes closest to the walls. Given that all considered boundary conditions are wet-node boundary conditions the last nodes should lie exactly on the wall. Therefore, the difference between the fluid velocities at these nodes and the velocity of the wall that lies on these nodes is the slip velocity Δu in the LBM simulations. For the analytical solution in (8.2) an equivalent procedure can be followed. For an x value of 0 or H the fluid velocity at the walls can be obtained. Hereafter, this value can be compared with the wall velocity at both walls to get the slip velocity. In Figure 8.3 the slip velocities are compared against the number of lattice nodes. The rounded values for the dimensionless relaxation time that are associated with the number of lattice nodes are shown in Table 8.1.

Table 8.1: The rounded $\tau^\star$ values computed from the number of LBM nodes in the characteristic length direction used in the simulations. The number of LBM nodes is displayed in terms of the lattice number as $N_{\Delta x^\star} + 1$. The values are computed according to (6.14).

$N_{\Delta x^\star} + 1$	2^3	2^4	2^5	2^6	2^7	2^8	2^9	2^{10}	2^{11}	2^{12}
$\tau^\star$	0.58	0.66	0.84	1.18	1.88	3.27	6.05	11.60	22.72	44.95

Hereafter, the slip velocities can be computed from the LBM simulations. These velocities are plotted in Figure 8.3 to check convergence with the analytical solution. On the vertical axis the dimensionless slip velocity is shown. The horizontal axis denotes the number of lattice nodes in the lateral direction. This number of LBM nodes will be shown in terms of the lattice number $N_{\Delta x^\star} + 1$.

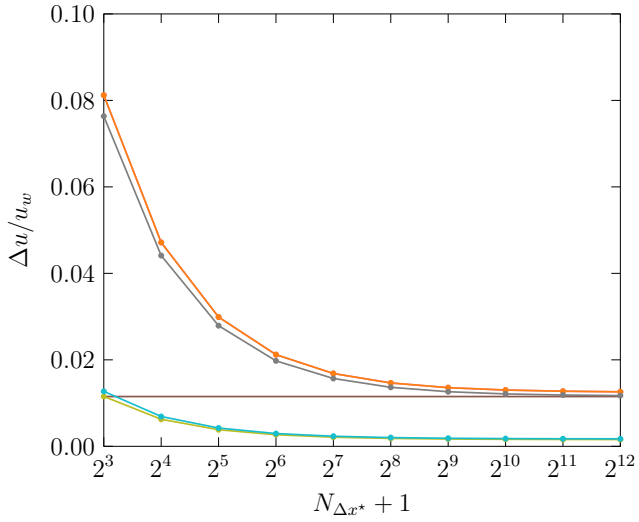


Figure 8.3: The normalized slip velocities generated by the LBM simulations plotted against the number of lattice nodes in the lateral direction. The analytical solution is displayed with (——). The gas-wall interaction kernels in the LBM simulations are denoted with: (—•—) for the slip-reflection kernel, (—•—) for the Maxwell kernel, (—•—) for the kernel learned from MD simulation E1, (—•—) for the kernels learned from simulation E1, V1 and V2 with linear interpolation and (—•—) for the kernels learned from simulation E1, V1 and V2 with quadratic interpolation.

As can be seen from Figure 8.3, when the number of LBM nodes in the lateral direction is increased the values for the velocity slip appear to converge. When comparing this with the analytical solution, it appears that the values for the interpolated MD kernels converge towards a too low value. This cannot be said from the Maxwell kernel, the slip-reflection kernel as well as the non-interpolated kernel learned from MD simulation E1. It appears that these kernels converge to the exact value provided by the analytical solution. Here, the slip-reflection kernel and the Maxwell kernel produce exactly the same solution for these simulation setups. Hence, their lines in Figure 8.3 lie exactly on top of each other. What

was found in Chapter 5 is also indicated in this plot, kernels based on Gaussian mixtures improve upon the performance of the Maxwell kernel. This is suggested by Figure 8.3 given that the value for the kernel learned from MD simulation E1 gives a slip velocity that is closer to the analytical solution than the tested traditional kernels. Furthermore, this result also gives an indication that the coupling procedure works.

To quantify the performance improvement of the kernel learned from MD simulation E1 on the slip-reflection and Maxwell kernel, Figure 8.4 was made. In this Figure, the L^2 norm for the error between the computed and the analytical solution is plotted as a function of the number of lattice nodes in the lateral direction.

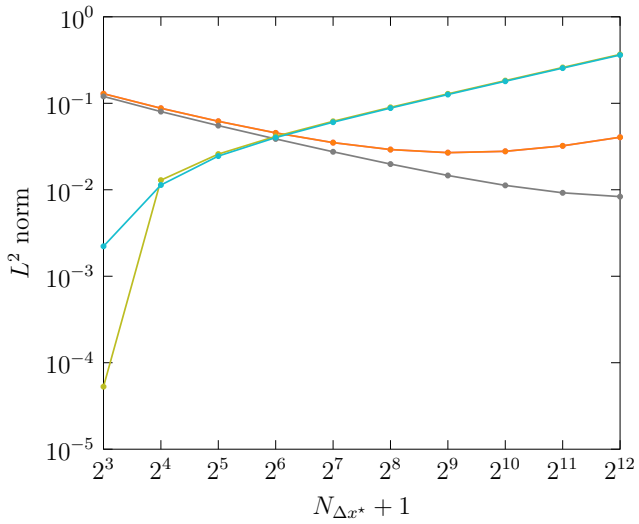


Figure 8.4: The L^2 norm for the error between the computed and the analytical solution plotted as a function of the number of lattice nodes in the lateral direction. The lines denote the following LBM gas-wall interaction kernels: (—●—) is the slip-reflection kernel, (—●—) is the Maxwell kernel, (—●—) is the kernel based on MD simulation E1, (—●—) is the kernel based on MD simulations E1, V1 and V2 with linear interpolation and (—●—) is the kernel based on MD simulations E1, V1 and V2 with quadratic interpolation.

As can be seen from Figure 8.4 the L^2 norm for the error of the LBM solutions shows decreasing behaviour for the slip-reflection kernel, the Maxwell kernel and the kernel learned from MD simulation E1 when the number of lattice nodes in the lateral direction is increased. What can also be seen from Figure 8.4 is that the L^2 norm of the error shows better convergence for the kernel learned from MD simulation E1. While it appears that the error for the slip-reflection and the Maxwell kernel reaches some kind of plateau for 2^9 LBM nodes, the kernel learned from MD simulation E1 keeps on reducing the error in the displayed window. For 2^{12} lattice nodes in the lateral direction a reduction of more than 79% in terms of the L^2 norm of the error is obtained when comparing the MD based kernel with the slip-reflection and the Maxwell kernel.

What should also be noted from Figure 8.3 and Figure 8.4 is that the interpolated kernels do not work flawlessly. The discrepancy with the analytical solution can be due to several causes. Hereof, the two most possible causes are likely the following: too crude assumptions for the interpolation and not enough MD datasets. As stated in Section 7.2 several assumptions were made before interpolation could be applied on the learned discrete scattering kernels. It cannot be ruled out whether these assumptions on the state of the flow and its relation to the wall velocities are too coarse. Besides, not enough MD datasets could also be a viable cause for the discrepancies. Interesting to note here is that the quadratic interpolation gives better results than the linear interpolation. This indicates that more datasets to be used in the interpolation could improve the results. Moreover, the wall velocities differ a lot between consecutive MD simulations in the interpolation. Between the learned discrete scattering kernels a speed ratio difference of 0.25 or a wall velocity difference of more than 88 m/s holds for the MD simulations. For the interpolated discrete kernels, this could lead to too much influence by the kernels learned for moving walls when the flow state at a wall is close to equilibrium. Hence, the velocity difference between fluid and wall could be reduced to a too small slip velocity.

9. Conclusions

In this thesis gas-wall interaction modelling for rarefied gases was further investigated. To this end, gas-wall interaction data from MD simulations of argon gas between two parallel gold walls was analyzed. The performance of standard parametric scattering kernels was tested on the gas-wall collision data as well as the performance of non-standard Gaussian mixture based scattering kernels. The GMM based scattering kernels showed improved performance with respect to the standard parametric kernels due to the higher number of parameters that could be used to describe the corresponding distributions.

To quantify this performance improvement several approaches were followed. The accommodation coefficients as well as the Kullback-Leibler divergences that were computed for the velocity distributions show better values for the GMM based scattering kernels. Also, for the plotted velocity distributions several differences can be observed between the kernels types. Likewise, it can be noted that the Gaussian mixture based scattering kernels outperform the standard parametric ones.

Hereafter, a method to directly couple the GMM based scattering kernels to boundary conditions in the lattice Boltzmann method was proposed. The proposed method does not rely on an empirical relation to model the gas-wall interaction but instead sets all the required parameters directly from the learned Gaussian mixture based scattering kernels. Furthermore, the proposed method can be trivially extended in full generality to higher dimensions as well as multi-speed lattices of arbitrary size. Besides, given that non-equilibrium molecular dynamics simulations are possible, the coupling can also be applied to simulate non-equilibrium gas-wall interaction in the lattice Boltzmann method.

As a proof of concept, the coupling algorithm was tested in the lattice Boltzmann method for rarefied Couette flow simulations in the slip flow regime. The scattering kernel constructed by the coupling algorithm from a GMM based scattering kernel trained on the data from the equilibrium MD simulation shows satisfactory convergence with the analytical solution. The L^2 norm of the error is almost one fifth of the error norm of the discrete Maxwell kernel for 2^{12} lattice nodes. Given that a performance improvement also holds for their continuous variants it can be seen as an indication that the algorithm produces correct results.

Additionally, interpolated versions of the discrete scattering kernels produced by the coupling algorithm from different MD simulations were tested. This interpolation provides relatively low overhead given that only the computed discrete kernels are interpolated. The constructed interpolated kernels did not deliver adequate convergence for the rarefied Couette flow simulations. There are two probable reasons for this performance gap. First, the assumptions underlying the matching between MD simulations and the state of the fluid near the wall in LBM might be incorrect. Second, more MD simulations might be needed so the discrete scatterings kernels can be more accurately interpolated.

Bibliography

[1] K. Huang, *Statistical mechanics*, 2nd ed. (Wiley, New York, 1987), 493 pp.

[2] M. N. Kogan, *Rarefied gas dynamics* (Springer US, Boston, 1969), 515 pp.

[3] C. Cercignani, *Rarefied gas dynamics: from basic concepts to actual calculations* (Cambridge University Press, New York, 2000), 320 pp.

[4] C. Wagner and N. Harned, "Lithography gets extreme", Nature Photonics **4**, 24–26 (2010).

[5] M. S. Ivanov and S. F. Gimelshein, "Computational hypersonic rarefied flows", Annual Review of Fluid Mechanics **30**, 469–505 (1998).

[6] R. G. Lord, "Some extensions to the Cercignani–Lampis gas–surface scattering kernel", Physics of Fluids A: Fluid Dynamics **3**, 706–710 (1991).

[7] T. Liang, Q. Li, and W. Ye, "Performance evaluation of Maxwell and Cercignani-Lampis gas-wall interaction models in the modeling of thermally driven rarefied gas transport", Physical Review E **88**, 013009 (2013).

[8] J. C. Maxwell, "On stresses in rarified gases arising from inequalities of temperature", Philosophical Transactions of the Royal Society of London **170**, 231–256 (1879).

[9] K. Yamamoto, H. Takeuchi, and T. Hyakutake, "Characteristics of reflected gas molecules at a solid surface", Physics of Fluids **18**, 046103 (2006).

[10] R. G. Lord, "Some further extensions of the Cercignani–Lampis gas–surface interaction model", Physics of Fluids **7**, 1159–1161 (1995).

[11] C. Cercignani and M. Lampis, "Kinetic models for gas-surface interactions", Transport Theory and Statistical Physics **1**, 101–114 (1971).

[12] C. Cercignani, "Scattering kernels for gas-surface interactions", Transport Theory and Statistical Physics **2**, 27–53 (1972).

[13] C. Cercignani and M. Lampis, "Free molecular flow past a flat plate in the presence of a nontrivial gas-surface interaction", Zeitschrift für angewandte Mathematik und Physik ZAMP **23**, 713–728 (1972).

[14] M. Liao, Q.-D. To, C. Léonard, and V. Monchiet, "Non-parametric wall model and methods of identifying boundary conditions for moments in gas flow equations", Physics of Fluids **30**, 032008 (2018).

[15] N. Andric, D. W. Meyer, and P. Jenny, "Data-based modeling of gas-surface interaction in rarefied gas flow simulations", Physics of Fluids **31**, 067109 (2019).

[16] M. Liao, Q.-D. To, C. Léonard, and W. Yang, "Prediction of thermal conductance and friction coefficients at a solid-gas interface from statistical learning of collisions", Physical Review E **98**, 042104 (2018).

[17] S. Mohammad Nejad, E. Iype, S. Nedea, A. Frijns, and D. Smeulders, "Modeling rarefied gas-solid surface interactions for Couette flow with different wall temperatures using an unsupervised machine learning technique", Physical Review E **104**, 015309 (2021).

[18] P. W. Atkins and J. De Paula, *Physical chemistry for the life sciences* (Oxford University Press ; W.H. Freeman, Oxford ; New York, 2006), 699 pp.

[19] S. V. Nedea, A. J. Markvoort, P. Spijker, and A. A. van Steenhoven, "Heat transfer predictions using accommodation coefficients for a dense gas in a micro/nanochannel", in ASME 2008 6th international conference on nanochannels, microchannels, and minichannels (2008), pp. 929–936.

[20] E. Davis, "A history and state-of-the-art of accommodation coefficients", Atmospheric Research **82**, 561–578 (2006).

[21] S. Mohammad Nejad, S. Nedea, A. Frijns, and D. Smeulders, "The influence of gas–wall and gas–gas interactions on the accommodation coefficients for rarefied gases: a molecular dynamics study", Micromachines **11**, 319 (2020).

[22] H. W. Sheng, M. J. Kramer, A. Cadien, T. Fujita, and M. W. Chen, "Highly optimized embedded-atom-method potentials for fourteen FCC metals", Physical Review B **83**, 134118 (2011).

[23] P. Spijker, A. J. Markvoort, S. V. Nedea, and P. A. J. Hilbers, "Computation of accommodation coefficients and the use of velocity correlation profiles in molecular dynamics simulations", Physical Review E **81**, 011203 (2010).

[24] S. A. Peddakotla, K. K. Kammara, and R. Kumar, "Molecular dynamics simulation of particle trajectory for the evaluation of surface accommodation coefficients", Microfluidics and Nanofluidics **23**, 79 (2019).

[25] G. Papamakarios, "Neural density estimation and likelihood-free inference", (2019).

[26] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the *EM* algorithm", Journal of the Royal Statistical Society: Series B (Methodological) **39**, 1–22 (1977).

[27] C. E. Rasmussen and C. K. I. Williams, *Gaussian processes for machine learning*, Adaptive computation and machine learning (MIT Press, Cambridge, 2006), 248 pp.

[28] H. G. Sung, "Gaussian mixture regression and classification", 117 (2004).

[29] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, and D. Cournapeau, "Scikit-learn: machine learning in Python", **12**, 6 (2011).

[30] S. Kullback and R. A. Leibler, "On information and sufficiency", The Annals of Mathematical Statistics **22**, 79–86 (1951).

[31] Q. Wang, S. Kulkarni, and S. Verdu, "A nearest-neighbor approach to estimating divergence between continuous random vectors", in 2006 IEEE international symposium on information theory (July 2006), pp. 242–246.

[32] P. A. Chambre and S. A. Schaaf, *Flow of rarefied gases* (Princeton University Press, 1961).

[33] S. Chapman and T. G. Cowling, *The mathematical theory of non-uniform gases: an account of the kinetic theory of viscosity, thermal conduction, and diffusion in gases*, 3rd ed., Cambridge mathematical library (Cambridge University Press, Cambridge ; New York, 1990), 422 pp.

[34] C. Cercignani, *The Boltzmann equation and its applications*, red. by F. John, J. E. Marsden, and L. Sirovich, Vol. 67, Applied Mathematical Sciences (Springer New York, New York, 1988).

[35] T. Krüger, H. Kusumaatmaja, A. Kuzmin, O. Shardt, G. Silva, and E. M. Viggen, *The lattice Boltzmann method: principles and practice*, Graduate Texts in Physics (Springer International Publishing, Cham, 2017).

[36] S. Chen and G. D. Doolen, "Lattice Boltzmann method for fluid flows", Annual Review of Fluid Mechanics **30**, 329–364 (1998).

[37] J. Latt, "Hydrodynamic limit of lattice Boltzmann equations", PhD thesis (University of Geneva, 2017).

[38] Y. H. Qian, D. D'Humières, and P. Lallemand, "Lattice BGK models for navier-stokes equation", Europhysics Letters (EPL) **17**, 479–484 (1992).

[39] G. H. Tang, W. Q. Tao, and Y. L. He, "Lattice Boltzmann method for gaseous microflows using kinetic theory boundary conditions", Physics of Fluids **17**, 058101 (2005).

[40] Y. Zhang, R. Qin, and D. R. Emerson, "Lattice Boltzmann simulation of rarefied gas flows in microchannels", Physical Review E **71**, 047702 (2005).

[41] M. Sbragaglia and S. Succi, "Analytical calculation of slip flow in lattice Boltzmann models with kinetic boundary conditions", Physics of Fluids **17**, 093602 (2005).

[42] X. He, Q. Zou, L.-S. Luo, and M. Dembo, "Analytic solutions of simple flows and analysis of nonslip boundary conditions for the lattice Boltzmann BGK model", Journal of Statistical Physics **87**, 115–136 (1997).

[43] S. Succi, "Mesoscopic modeling of slip motion at fluid-solid interfaces with heterogeneous catalysis", Physical Review Letters **89**, 064502 (2002).

[44] S. Ansumali and I. V. Karlin, "Kinetic boundary conditions in the lattice Boltzmann method", Physical Review E **66**, 026311 (2002).

[45] G. Karniadakis and A. Beşkök, *Micro flows: fundamentals and simulation* (Springer, New York, 2002), 340 pp.