

Andreas Krämer

Lattice-Boltzmann-Methoden zur Simulation inkompressibler Wirbelströmungen

Herausgeber: Holger Foysi und Thomas Seeger

**Schriftenreihe des Lehrstuhls für
Strömungs- und Thermodynamik**

Band 3

Lattice-Boltzmann-Methoden zur Simulation inkompressibler Wirbelströmungen

DISSERTATION

zur Erlangung des Grades eines Doktors
der Naturwissenschaften

vorgelegt von

Dipl.-Math. Andreas Krämer

Hauptreferent:	Prof. Dr.-Ing. Holger Foysi
Koreferent:	Prof. Dr.-Ing. Gunther Brenner
Vorsitzender:	Prof. Dr.-Ing. Holger Foysi
Dekan:	Prof. Dr. Dr. h. c. Ullrich Pietsch

Tag der Einreichung: 12.06.2017

Tag der mündl. Prüfung: 12.10.2017

eingereicht bei der Naturwissenschaftlich-Technischen Fakultät
der Universität Siegen

Siegen 2017

gedruckt auf alterungsbeständigem holz- und säurefreiem Papier

Impressum

Prof. Dr.-Ing., Dipl.-Phys. Holger Foyi

Lehrstuhl für Strömungsmechanik

Universität Siegen

57068 Siegen

Band 3

Zugl.: Siegen, Univ., Diss. 2017

Danksagung

Die vorliegende Dissertation entstand während meiner Tätigkeit als wissenschaftlicher Mitarbeiter an der Hochschule Bonn-Rhein-Sieg. Die Rechnungen wurden auf dem WR-Cluster der Hochschule Bonn-Rhein-Sieg und auf dem HorUS-Cluster der Universität Siegen durchgeführt.

An dieser Stelle möchte ich gerne allen Personen danken, die mich während meiner Promotion unterstützt haben. Mein besonderer Dank gilt Prof. Dr.-Ing. Holger Foyssi für die Gelegenheit, an seinem Lehrstuhl zu promovieren, für die inspirierenden Diskussionen, das kontinuierliche Interesse und den gewährten Freiraum.

Ein herzlicher Dank gilt auch Prof. Dr.-Ing. Gunther Brenner für die Bereitschaft, das Koreferat zu übernehmen sowie Prof. Dr. Volker Michel, Prof. Dr.-Ing. Thomas Seeger und Prof. Dr. Franz-Theo Suttmeier für ihre Bereitschaft zur Mitarbeit in der Promotionskommission.

Prof. Dr. Wolfgang Joppich und Prof. Dr. Dirk Reith möchte ich für die Begleitung meiner Arbeit an der Hochschule Bonn-Rhein-Sieg und die vielfältige Unterstützung danken. Dem Fachbereich EMT und dem Graduierteninstitut der Hochschule Bonn-Rhein-Sieg sowie den Arbeitsgruppen in Siegen und Sankt Augustin danke ich für die Schaffung guter Rahmenbedingungen für die Promotion. Ein herzlicher Dank gilt meinen Kollegen Knut Küllmer und Dominik Wilde, die mich während der letzten Jahre fachlich und persönlich bereichert haben.

Ich danke meinen langjährigen Kollegen Dr. Marco Hülsmann, Dr. Karl Kirschner und Martin Schenk für das harmonische Miteinander, allen Studenten, die ich in ihren Master-, Bachelor-, und Projektarbeiten mitbetreuen durfte für ihre Mitarbeit sowie allen Personen, die Teile meiner Arbeit Korrektur gelesen haben.

Besonders bedanken möchte ich mich bei meinen Eltern, meiner Familie und meinen Freunden für die fortwährende Unterstützung und Ermutigung.

Zum Schluss danke ich meiner Frau Lisa, deren Liebe, Freundschaft, Geduld und Loyalität mein Leben auf einzigartige Weise bereichert.

Abstract

The present work develops novel methodological extensions to the lattice Boltzmann method (LBM). These extensions enable the method to efficiently simulate incompressible vortical flows. They cure two major drawbacks of the standard LBM: its instability in under-resolved turbulence and its restriction to regular computational grids. At first, a *pseudo-entropic stabilizer* (PES) is developed, which combines ideas from multiple-relaxation-time (MRT) models and entropic models. The new PES is local, explicit, and flexible. It modifies the collision step in a way that enables stable simulations and produces qualitatively matching results, even on strongly under-resolved grids. To extend the LBM towards simulations on irregular grids, a recent discontinuous Galerkin lattice Boltzmann method is studied and enhanced by more stable time integrators. This study illustrates the severe shortcomings of existing off-lattice Boltzmann methods (OLBMs). Based on these findings, the present work succeeds in developing a *semi-Lagrangian lattice Boltzmann method* (SLLBM). This novel approach allows unstructured grids, large time steps, and high-order accurate representations of the solution to be used in a unique way. Applications to exemplary flows demonstrate how and why the new method outperforms other recent OLBMs in both efficiency and accuracy. Additionally, this work describes the development of a modular off-lattice Boltzmann code and shows that the method's convergence order can be increased by implicit multistep methods.

Zusammenfassung

In dieser Arbeit werden neuartige methodische Erweiterungen der Lattice-Boltzmann-Methode (LBM) entwickelt, die effizientere Simulationen inkompressibler Wirbelströmungen ermöglichen. Diese Erweiterungen beheben zwei Hauptprobleme der Standard-LBM: ihre Instabilität in unteraufgelösten turbulenten Simulationen und ihre Beschränkung auf reguläre Rechengitter. Dazu wird zunächst eine *Pseudo-Entropische Stabilisierung* (PES) entwickelt. Diese kombiniert Ansätze der Multiple-Relaxation-Time (MRT)-Modelle und der Entropischen LBM zu einem expliziten, lokalen und flexiblen Stabilisierungsoperator. Diese Modifikation des Kollisionsschritts erlaubt selbst auf stark unteraufgelösten Gittern stabile und qualitativ korrekte Simulationen. Zur Erweiterung der LBM auf irreguläre Rechengitter wird zunächst eine moderne Discontinuous-Galerkin-LBM untersucht und um stabilere Zeitintegratoren ergänzt. Diese Studie demonstriert die drastischen Schwächen existierender LBM-Ansätze auf irregulären Gittern. Basierend auf den gewonnenen Erkenntnissen gelingt die Formulierung einer neuartigen *Semi-Lagrangischen LBM* (SLLBM). Diese ermöglicht in einzigartiger Weise sowohl die Verwendung irregulärer Gitter und großer Zeitschritte als auch eine hohe räumliche Konvergenzordnung. Anhand von Beispielsimulationen wird demonstriert, wieso dieser Ansatz anderen aktuellen Off-Lattice-Boltzmann-Methoden (OLBMs) in Effizienz und Genauigkeit überlegen ist. Weitere neuartige Aspekte dieser Arbeit sind die Entwicklung eines modularen Off-Lattice-Boltzmann-Codes und die Erweiterung der LBM um implizite Mehrschrittverfahren, mit denen eine Erhöhung der zeitlichen Konvergenzordnung gelingt.

Inhaltsverzeichnis

1	Einleitung	1
2	Strömungssimulation mit der Lattice-Boltzmann-Methode	9
2.1	Makroskopische Grundgleichungen	9
2.2	Kinetische Gastheorie	11
2.3	Dimensionslose Kennzahlen	12
2.4	Lattice-Boltzmann-Methode	13
2.5	Entwicklung des Strömungslösers	22
2.6	Erste Testfälle	29
3	Kollisionsmodellierung	35
3.1	Stand der Technik	35
3.2	Pseudo-Entropische Stabilisierung	39
3.3	Simulationen mit der Pseudo-Entropischen Stabilisierung	47
3.4	Diskussion der Pseudo-Entropischen Stabilisierung	56
4	Räumliche Diskretisierungen: Eulersche Methoden	63
4.1	Stand der Technik	64
4.2	Discontinuous-Galerkin-Lattice-Boltzmann-Verfahren	68
4.3	Simulationen mit Discontinuous-Galerkin-Lattice-Boltzmann-Verfahren	73
4.4	Diskussion der Discontinuous-Galerkin-Lattice-Boltzmann-Verfahren .	84

5	Räumliche Diskretisierungen: Semi-Lagrangesche Methode	87
5.1	Semi-Lagrange-Verfahren	87
5.2	Stand der Technik: Interpolationsbasierte Lattice-Boltzmann-Methoden	89
5.3	Semi-Lagrangesches Lattice-Boltzmann-Verfahren	90
5.4	Simulationen mit der Semi-Lagrangeschen Lattice-Boltzmann-Methode	94
5.5	Diskussion der Semi-Lagrangeschen Lattice-Boltzmann-Methode . . .	112
6	Zusammenfassung und Ausblick	115
A	Mehrschritt-Lattice-Boltzmann-Methoden	121
A.1	Adams-Moulton-Verfahren vierter Ordnung	122
A.2	BDF-Verfahren zweiter Ordnung	123
A.3	Numerischer Test	123
A.4	Diskussion	123
B	Chapman-Enskog-Analyse der BGK-Gleichung	127
B.1	Mehrskalen-Entwicklung	129
B.2	Analyse der Momentengleichungen	129
B.3	Zusammensetzen der Skalen	130
C	Momentenmatrizen und Stabilisierungsoperatoren	133
C.1	Matrizen für den D2Q9-Stern	133
C.2	Matrizen für den D3Q19-Stern	135
D	Randbedingungen für die Semi-Lagrangesche Methode	139
D.1	Konstruktion der Verteilungsfunktionen an Randpunkten	139
D.2	Konstruktion der Verteilungsfunktionen an inneren Punkten	140
D.3	Bounce-Back-Randbedingung	141

Symbolverzeichnis

Lateinische Buchstaben

A	Matrix zur Berechnung der optimierten nicht-physikalischen Momente
A_c	Differentieller Wirkungsquerschnitt im Kollisionsintegral
a	Amplitude der Sinus-Geometrie
a_i	Koeffizienten von Mehrschrittverfahren
B	Matrix zur direkten Berechnung der optimierten Verteilungsfunktionen
$B(X, \mathbb{R})$	Raum der beschränkten skalaren Funktionen auf X
\hat{b}	Konstanter Vektor der Randbedingungen im SEDG-Strömungsoperator
b_i	Koeffizienten von Mehrschrittverfahren
c_s	Schallgeschwindigkeit
D_c	Gitterzelle
E	Totale Energie
E_{SLLBM}	Fehler der Semi-Lagrangeschen LBM
$E_{\rho u_i}$	Fehlerterm der Impulsgleichung in der LBM
e	Energiemoment in [42] und [47]
\mathbf{e}_i, e_{ij}	Diskrete Teilchengeschwindigkeiten
e_i	Diskrete Teilchengeschwindigkeiten in 1D
e_ν	Fehler in der effektiven Viskosität
$\mathbf{F}^{\text{ext}}, F_i^{\text{ext}}$	Externe Kraft
$f(t, \mathbf{x}, \boldsymbol{\xi}), \tilde{f}$	Molekulare Geschwindigkeitsverteilungsfunktion
$f_{1\text{D}}^{\text{eq}}$	Eindimensionale Maxwell-Boltzmann-Verteilung
$f_{\text{MB}}^{\text{eq}}$	Maxwell-Boltzmann-Verteilung
$f_{1\text{D}}$	Eindimensionale polynomielle Geschwindigkeitsverteilungsfunktion
f, f_i	Diskrete Geschwindigkeitsverteilungsfunktion
$f^{\text{eq}}, f_i^{\text{eq}}$	Diskrete Gleichgewichtsverteilung
f^{pc}	Postkollisionsverteilung
f^{reg}	Regularisierte Geschwindigkeitsverteilungsfunktion
f^{neq}	Nichtgleichgewichtsanteil der Geschwindigkeitsverteilungsfunktion

f^{filt}	Gefilterte Geschwindigkeitsverteilungsfunktion
f^{pe}	Pseudo-entropisch stabilisierte Geschwindigkeitsverteilungsfunktion
f_i^+	Einfließende Verteilungsfunktion
\hat{f}, \hat{f}_{jck}	Freiheitsgrade der diskretisierten Verteilungsfunktion
\bar{f}	Transformierte Verteilungsfunktion
$f^{[\kappa]}, f_i^{[\kappa]}$	Schritte des Mehrschrittverfahrens
$f^{\text{eq},[\kappa]}, f_i^{\text{eq},[\kappa]}$	Gleichgewichtsverteilung im Mehrschrittverfahren
$f^{(i)}$	Abweichungen vom Gleichgewicht verschiedener Ordnung
$g, (g^{\text{eq}})$	Polynomielle Approximation in der (Gleichgewichts-)Verteilung
H, \tilde{H}	Entropie und quadratische Pseudo-Entropie
\bar{h}	Mittlere Höhe der Sinus-Geometrie
J_x, J_y	Höheres Moment bei Dellar [42]
j_x, j_y, j_z	Impuls in x -, y - und z -Richtung
k	Kinetische Energie
Kn	Knudsen-Zahl
L	Linearer SEDG-Strömungsoperator
L_x	Länge der Sinus-Geometrie
l_0	Charakteristische Länge einer Strömung
l_m	Mittlere freie Weglänge
M, m_{ij}	Momentenmatrix
\hat{M}, \hat{m}_{ij}	Massenmatrix
$m_{x,y,z}$	Höheres Moment in [135] und [47]
Ma	Machzahl
m, m^{eq}	Momente der Geschwindigkeitsverteilung und Gleichgewichtsverteilung
$m_{\mathcal{P}}, m_{\mathcal{Q}}$	Hydrodynamische und nicht-hydrodynamische Momente
$m_{\mathcal{P}}^{\text{pc}}, m_{\mathcal{Q}}^{\text{pc}}$	Hydrodynamische und nicht-hydrodynamische Postkollisionsmomente
N	Ordnung der Ansatzpolynome
\hat{N}	Höheres Moment bei Dellar [42]
\mathbf{n}, n_i	Normaleneinheitsvektor
O	Landau-Symbol
p	Druck
$p(\zeta)$	Polynom
p_{xx}, p_{xy}	Schermomente bei [42] und [47]
p_{ww}, p_{xz}, p_{yz}	Schermoment bei D'Humieres et al. [47]
Q	Anzahl diskreter Geschwindigkeiten
\mathbf{q}, q_i	Wärmefluss
$q_{x,y,z}$	Höheres Moment in [135] und [47]
Re	Reynoldszahl
Re_{eff}	Effektive Reynoldszahl
R, R^{eq}	Linearer und nichtlinearer Anteil des Kollisionsoperators
\hat{R}, \hat{r}_{ij}	Matrix für die Disretisierung des Randintegrals
S	Diagonale Matrix von Relaxationsparametern

\hat{S}, \hat{s}_{ij}	Steifigkeitsmatrix
s	Anzahl Schritte im Mehrschrittverfahren
T, t_{ij}	Inverse der Momentenmatrix
t	Zeit
\mathbf{u}, u_i	Geschwindigkeitsvektor
\mathbf{u}', u'_i	Geschwindigkeitsfluktuationen
u	Geschwindigkeit in 1D
u_0	Charakteristische Geschwindigkeit einer Strömung
\mathbf{u}_w, u_w	Wandgeschwindigkeit
u_τ	Wandschubspannungsgeschwindigkeit
W	Diagonalmatrix der reziproken Wichtungsfaktoren
X, X'	Rechengitter und temporäres Rechengitter
\mathbf{x}, x_i	Koordinatenvektor
x	Kartesische Koordinate
x_r	Gitterkoordinate auf einem kartesischen Gitter
$\tilde{\mathbf{x}}, \tilde{x}_i, \tilde{x}$	Punkt auf der Einheitszelle
$\tilde{\mathbf{x}}_k, \tilde{x}_{ki}, \tilde{x}_k$	Stützpunkt auf der Einheitszelle
\mathbf{x}_{ck}, x_{cki}	Stützpunkt einer Gitterzelle
$\mathbf{x}_{cji}^w, \mathbf{x}_{cji}^{\text{BB}}$	Schnittpunkt von Trajektorie und Rand, Bounce-Back-Ausgangspunkt
y	Kartesische Koordinate
y_r	Gitterkoordinate auf einem kartesischen Gitter
z	Kartesische Koordinate

Griechische Buchstaben

α_i	Gewichte der Gauß-Hermite-Quadratur
ΔS	Abweichung vom Gleichgewicht
ΔS_{med}	Median der Abweichungen in der Nachbarschaft eines Gitterpunkts
δ	Halbe Kanalhöhe
δ_{ij}	Kronecker-Symbol
δ_t, δ_x	Zeitschritt und Maschenweite des Rechengitters
ε	Parameter zur Aufteilung der Verteilungsfunktion in verschiedene Skalen
$\tilde{\varepsilon}$	Höheres Moment bei Lallemand und Luo [135]
ϵ	Turbulente Dissipationsrate
η	Kolmogorowsche Mikroskala
$\vartheta, \vartheta_i, \vartheta_{ij}$	Lagrangemultiplikatoren
θ_{cji}^w	Zeitpunkt, an dem die Trajektorie den Rand trifft
ζ_i	Stützstellen der Gauß-Hermite-Quadratur
κ	Parameter in der Mehrschritt-LBM
λ	Relaxationszeit
λ_m	Frequenz in der analytischen Fourierreihe der Couette-Strömung

μ	Dynamische Viskosität
ν	Kinematische Viskosität
ν_{eff}	Effektive Viskosität
$\xi, \tilde{\xi}, \xi_i$	Teilchengeschwindigkeit
ξ	Teilchengeschwindigkeit in 1D
ξ_0	Skalierung der diskreten Molekülgeschwindigkeiten
$\Pi^{(1)}, \Pi_{ij}^{(1)}$	Spannungstensor erster Ordnung in [140]
$\Pi_{xx}, \Pi_{xy}, \Pi_{yy}$	Schermomente bei Dellar [42]
π_{xx}, π_{ww}	Höhere Momente bei D’Humières et al. [47]
ρ	Dichte
ρ_0	Mittlere/Initiale Dichte einer LBM-Simulation
$\hat{\rho}$	Spektralradius
$\tau, (\tau_0, \tau_{-1})$	Dimensionslose Relaxationsparameter der LBM (und Mehrschritt-LBM)
$\boldsymbol{\tau}, \tau_{ij}$	Viskoser Spannungstensor
Υ	Skalare Strömungsgröße im Semi-Lagrange-Verfahren
Φ	Semi-Lagrangescher Advektionsoperator im Einheitsintervall
χ	Indikatorfunktion
Ψ, Ψ'	Interpolationsoperatoren der Vorwärts- und Rückwärts-Diskretisierung
Ψ_x	Scherflussfaktor
ψ, ψ_1, ψ_2	Testfunktionen
ψ_k	Ansatzfunktion
$\tilde{\psi}_k$	Ansatzfunktion auf der Einheitszelle
Ω, Ω_i	Kollisionsoperator
$\bar{\Omega}_{\delta t}$	Transformierter Kollisionsoperator
ω	Wirbelstärke (in 2D)
ω_i	Wichtungsfaktoren der diskreten Geschwindigkeitsverteilung
$\boldsymbol{\omega}$	Wirbelstärkevektor (in 3D)
$\tilde{\omega}_i$	Gewichte einer Gauß-Quadratur

Sonstige Symbole

$\partial_{t1}, \partial_{t2}$	Zeitableitung auf zwei verschiedenen Skalen
$\partial_i^+, \partial_i^-$	Ränder einer Zelle, über die eine Verteilungsfunktion ein- bzw. ausfließt
\mathcal{H}_{ij}	Hermite-Basispolynom
\mathcal{E}	Enstrophie
l	Verfeinerungsstufe des Gitters
\mathcal{M}_c	Mapping-Funktion
\mathcal{P}	Menge der Indizes der physikalischen Momente
\mathcal{P}^+	Turbulente Produktion in Wandeinheiten
\mathcal{Q}	Menge der Indizes der nicht-physikalischen Momente

Abkürzungsverzeichnis

BBGKY(-Hierarchie)	Bogoliubov–Born–Green–Kirkwood–Yvon-Hierarchie
BGK(-Modell)	Bhatnagar-Gross-Krook-Modell der Boltzmann-Gleichung
CFD	Computational Fluid Dynamics
CFL	Courant-Friedrichs-Lewy
DNS	Direkte Numerische Simulation
DUGKS	Discrete Unified Gas-Kinetic Scheme von Guo et al. [80]
KBC	Entropisches Two-Relaxation-Time-Modell von Karlin, Bösch und Chikatamarla [117]
LES	Large-Eddy-Simulation
LBM	Lattice-Boltzmann-Methode
MLUPS	Million Lattice Updates per Second
MRT(-Modell)	Multiple-Relaxation-Time-Modell
OLBM	Off-Lattice-Boltzmann-Methode
PES	Pseudo-Entropische Stabilisierung
RANS	Reynolds-Averaged Navier-Stokes
SEDG	Spektral-Element-Discontinuous-Galerkin
SLLBM	Semi-Langrangesche Lattice-Boltzmann-Methode
TRT(-Modell)	Two-Relaxation-Time-Modell

Zur Notation:

- Summenkonvention: Über doppelte Indizes eines Terms wird summiert.
- Geklammerte Indizes sind von der Summenkonvention ausgeschlossen.
- Fettgedruckte mathematische Symbole bezeichnen Vektoren und Tensoren, die die Dimension des Raums besitzen.
- $(a_i)_i$ bezeichnet den Vektor, der die Elemente a_i enthält.
- $(a_{ij})_{ij}$ bezeichnet die Matrix, die die Elemente a_{ij} enthält.

1. Einleitung

„Science may be described as the art of systematic over-simplification – the art of discerning what we may with advantage omit.“

— Karl Popper, *The Open Universe: An Argument for Indeterminism*
(1982)

Computersimulationen als Motor der modernen Naturwissenschaft und Technik

Der wohl prägendste Aspekt des gegenwärtigen Zeitalters ist die digitale Revolution. Der Aufstieg moderner Computer durchdringt heute in Deutschland nicht nur das gesellschaftliche Leben, sondern hat auch Wissenschaft und Technik in nahezu allen Belangen drastisch verändert. Durch die stetige Entwicklung immer effizienterer Rechner steigt in vielen Disziplinen die Bedeutung von Computersimulationen. Diese haben sich in den Naturwissenschaften als Alternative zu Laborexperimenten etabliert und beschleunigen in vielen technischen Anwendungen die Entwicklung von Prototypen.

Modellbildung und Simulation bilden ein Paradebeispiel für eine interdisziplinäre Wissenschaft, die sich hauptsächlich aus Aspekten der Mathematik, Physik und Informatik zusammensetzt. Wenn Popper im einleitenden Zitat die Wissenschaft als „Kunst der Übereinfachung“ beschreibt, so gilt dies in besonderer Weise für die Simulationswissenschaften, in denen sich eine hohe Modellkomplexität normalerweise in langen Simulationszeiten niederschlägt. Aus diesem Grund verwenden Simulationsmethoden im Idealfall einfache Modelle, die dennoch in der Lage sind, die auf der betrachteten Skala relevanten physikalischen Effekte wiederzugeben.

Simulationen werden heute auf allen Zeit- und Längenskalen verwendet, von der quantenmechanischen Berechnung subatomarer Energien ($< 10^{-10}$ m) bis hin zur Simulation astrophysikalischer Vorgänge ($> 10^{10}$ m) [13, 225]. In der Mitte dieser extremen

Skalen liegen technisch relevante Problemstellungen, in denen die numerische Strömungsmechanik (*Computational Fluid Dynamics*, CFD) eine besonders wichtige Rolle spielt [105].

Klassische Strömungssimulation: Direkte Lösung der Navier-Stokes-Gleichungen

Die CFD simuliert die Dynamik von Fluiden, also von Flüssigkeiten und Gasen. Über die letzten Jahrzehnte ist sie zu einem unverzichtbaren Werkzeug in den Ingenieurwissenschaften geworden, beispielsweise zur Vorhersage aerodynamischer Eigenschaften von Flug- und Fahrzeugen [45, 84]. Für die meisten Anwendungen können die Fluide in einer guten Näherung durch die Navier-Stokes-Gleichungen modelliert werden, welche die Erhaltung von Masse, Impuls und Energie beschreiben [131]. Klassische Verfahren der Strömungsmechanik lösen diese Gleichungen ausgehend von problem-spezifischen Anfangs- und Randbedingungen, wobei das Strömungsgebiet durch ein Rechengitter diskretisiert wird.

Besonders herausfordernd ist die Simulation turbulenter Strömungen, in denen der nichtlineare Term der Navier-Stokes-Gleichungen Wirbel auf immer kleineren Skalen erzeugt. Das Spektrum einer voll entwickelten turbulenten Strömung erstreckt sich von der charakteristischen Längenskala des Systems l_0 bis zur Skala der kleinsten Wirbel, der Kolmogorowschen Mikroskala $\eta \ll l_0$. Die Auflösung des gesamten turbulenten Spektrums erfordert fein aufgelöste Gitter und bedeutet in der Regel einen enormen Simulationsaufwand. Bei solchen Rechnungen spricht man von *Direkter Numerischer Simulation* (DNS). In der Praxis werden statt einer teuren DNS gerne entweder *Large-Eddy-Simulationen* (LES) verwendet, die nur die größeren Wirbel (*Eddies*) auflösen, oder die Reynolds-gemittelten Naviers-Stokes-Gleichungen gelöst (*Reynolds-Averaged Navier-Stokes*, RANS). Die Ergebnisqualität von LES- und RANS-Simulationen hängt stark von den verwendeten Turbulenzmodellen ab. Außerdem erfordern auch solche vergrößerten Rechnungen in der Praxis oft einen hohen Aufwand und damit hocheffiziente numerische Verfahren und Simulationscodes.

In klassischen CFD-Methoden werden die Erhaltungsgleichungen mit Standardverfahren der Numerik (Finite-Differenzen-, Finite-Elemente-, Finite-Volumen-, Discontinuous-Galerkin- und Spektral-Methoden) in große Systeme gewöhnlicher Differentialgleichungen überführt. Diese werden dann mit Zeitintegratoren iterativ gelöst. Jeder Zeitschritt des Integrators erfordert die (approximative) Berechnung aller Ableitungen der Navier-Stokes-Gleichungen und – im Fall inkompressibler Strömungen – die iterative Lösung der Poisson-Gleichung zur Bestimmung des Druckfelds. Der hohe rechnerische Aufwand ergibt sich damit aus der physikalischen Komplexität der Navier-Stokes-Gleichungen.

Lattice-Boltzmann: Indirekte Lösung der Navier-Stokes-Gleichungen

Aufsehen erregte deshalb um 1990 die Entwicklung einer neuen Methode, die diese Komplexität durch die Verwendung alternativer Paradigmen umgeht [103, 159]. Die Lattice-Boltzmann-Methode (LBM) ist im Gegensatz zu den etablierten Methoden der CFD in der Boltzmann-Gleichung verwurzelt, die die Dynamik von Teilchenverteilungen in verdünnten Gasen beschreibt. Die LBM ist daher *per definitionem* eine Mehrskalenmethode. Statt der makroskopischen Erhaltungsgleichungen modelliert sie eine diskrete *mesoskopische* Teilchendynamik auf einem regulären Rechengitter. In dieser gaskinetischen Betrachtung wird die gesamte Nichtlinearität der Strömung in einem lokalen Kollisionsschritt kodiert, was die Berechnung von Ableitungen vermeidet. Die LBM umgeht zudem die Lösung einer Poissongleichung, indem Druckfluktuationen durch schnelle lineare Strömungsschritte der Verteilungsfunktionen kommuniziert werden. Sie ist also im eigentlichen Sinn keine inkompressible Methode, hat sich aber im Stil eines pseudo-kompressiblen Ansatzes [38] primär zur Simulation inkompressibler Strömungen etabliert. Trotz ihrer Verwurzelung in der Gaskinetik ist die LBM konsistent mit den Navier-Stokes-Gleichungen und eignet sich daher auch zur Simulation von Flüssigkeiten.

Die Berechnung der diskreten Teilchendynamik in der LBM erfordert kleinere Zeitschritte und mehr Freiheitsgrade als klassische CFD-Methoden, allerdings sind die einzelnen Iterationen wesentlich einfacher und effizienter. Die bis heute steigende Attraktivität der Methode resultiert aus ihrer Kongruenz mit dem Trend zum parallelen Rechnen. Statt auf wenige leistungsstarke Prozessoren setzt die heutige Computerindustrie auf hochparallele Architekturen. In diesem Kontext ist gerade die Lokalität des Kollisionsschritts ein entscheidender Trumpf der LBM gegenüber klassischen CFD-Methoden, der im Windschatten moderner Rechnerarchitekturen stetig relevanter wird.

Andererseits birgt die LBM in ihrer Standardformulierung [186] auch einige Schwierigkeiten. Der Begriff *Standard-LBM* bezeichnet in diesem Zusammenhang die oft verwendete, einfache Version der LBM mit einem exakten Strömungsschritt auf einem regulären Gitter, wobei die Kollision lokal mit dem Bhatnagar-Gross-Krook (BGK)-Modell und einer quadratischen Gleichgewichtsverteilung durchgeführt wird [186]. Die offensichtlichste Einschränkung dieser Formulierung ist die Verwendung regulärer Rechengitter. In der traditionellen CFD werden standardmäßig Gitter verwendet, die der Geometrie des Strömungsgebiets und der Physik der Strömung angepasst sind. So müssen zum Beispiel in wandgebundener Turbulenz die wandnahen Bereiche wesentlich feiner aufgelöst werden, da in diesen Bereichen die höchsten Geschwindigkeitsgradienten auftreten und die meiste Turbulenz produziert wird. Eine solche angepasste Diskretisierung ist in der Standard-LBM nicht möglich, was die Methode für viele praktische Anwendungen disqualifiziert. Die effiziente Erweiterung auf irreguläre oder unstrukturierte Gitter ist problematisch, da die Diskretisierung der

Teilchenverteilungen direkt mit dem regulären Gitter gekoppelt ist.

Eine weitere Herausforderung ist die Formulierung stabiler Kollisionsmodelle. Das oft verwendete BGK-Modell relaxiert die diskreten Verteilungsfunktionen $f \in \mathbb{R}^Q$ zu einer diskreten Maxwell-Boltzmann-Verteilung $f^{\text{eq}} \in \mathbb{R}^Q$, sodass sich die Postkollisionsverteilung $f^{\text{pc}} \in \mathbb{R}^Q$ wie folgt ergibt:

$$f^{\text{pc}} = f - \frac{1}{\tau} (f - f^{\text{eq}}).$$

Die Relaxationszeit τ berechnet sich aus der Viskosität ν , dem Zeitschritt δ_t und der Schallgeschwindigkeit c_s als

$$\tau = 0.5 + \frac{\nu}{\delta_t c_s^2}.$$

Bei unteraufgelösten, niedrigviskosen Stömungen erhält man somit $\tau \approx 0.5$, was einer nahezu exakten Spiegelung von f an der Gleichgewichtsverteilung entspricht. Gerade im Fall unteraufgelöster Wirbelströmungen erzeugt dieses Schema kaum Dissipation und führt deshalb schnell zu instabilen Simulationen.

Ein Großteil der Forschung zur LBM befasst sich damit, die Probleme und Einschränkungen der Standard-LBM zu beheben. Die größte Herausforderung besteht dabei darin, die Methode so zu erweitern, dass die grundsätzlichen Vorteile der Methode nicht wesentlich beschädigt werden.

Die vorliegende Arbeit entwickelt neuartige Erweiterungen der LBM für Simulationen inkompressibler Wirbelströmungen. In der Standard-LBM sind aus den oben genannten Gründen sowohl die Instabilität des BGK-Modells als auch die Verwendung regulärer Gitter problematisch. Deshalb werden in diese Arbeit sowohl alternative Kollisionsmodelle als auch alternative räumliche Diskretisierungen erarbeitet.

Alternative Kollisionsmodelle als Schlüssel zur Stabilität

Die Entwicklung stabilerer Kollisionsmodelle rückte um das Jahr 2000 in den Fokus. Meilensteine waren die Entwicklung der Entropischen LBM [118] und der *Multiple-Relaxation-Time* (MRT)-Methoden [41, 42, 46, 47, 135]. Während die Entropische LBM durch eine iterative Prozedur ein Anwachsen der Entropie im Kollisionsschritt erzwingt, relaxieren die MRT-Modelle verschiedene Momente der Verteilungsfunktion mit verschiedenen Relaxationszeiten. Die MRT-Modelle basieren auf der Tatsache, dass die Viskosität in erster Linie durch die Relaxation der Momente zweiter Ordnung beeinflusst wird [135]. Beliebte wurde in diesem Zusammenhang die Regularisierte LBM [11, 139, 166, 167], die alle höheren Momente im Kollisionsschritt eliminiert. Da sowohl die Entropische LBM als auch die MRT-Methoden die Lokalität des Kollisionsschritts erhalten, bilden sie bis heute den Ausgangspunkt für die meisten veröffentlichten Kollisionsmodelle (zum Beispiel [43, 68, 70, 74, 117, 139, 157]). Allerdings existieren nur wenige Arbeiten zur Beziehung der beiden Ansätze [228].

Eine interessante Entwicklung ist in diesem Zusammenhang das entropische MRT-Modell von Karlin, Bösch und Chikatamarla (KBC) [18, 117], welches Aspekte der entropischen und MRT-Modelle kombiniert. KBC adaptiert den freien Relaxationsparameter in einem *Two-Relaxation-Time* (TRT)-Modell über die Maximierung einer quadratischen Pseudo-Entropie-Funktion. Das resultierende Modell ist lokal, explizit, hat gute Stabilitätseigenschaften [155] und besitzt keine frei wählbaren Parameter.

Allerdings stellt die Beschränkung auf zwei unabhängige Relaxationszeiten eine künstliche Einschränkung dar. Obwohl in [115] eine mögliche Erweiterung auf mehrere Relaxationszeiten skizziert wurde, ist nicht klar, ob und inwieweit dadurch die Eigenschaften des Modells verbessert werden können. Außerdem ist unklar, welche Rolle die Entropie in anderen MRT-Modellen spielt. Die Formulierung und Untersuchung besserer Kollisionsmodelle für die LBM wird bis heute kontrovers diskutiert [120, 150, 151] und ist weiterhin eins der wichtigsten und aktivsten Themenfelder in der Forschung zur LBM, siehe zum Beispiel [43, 74, 117, 156].

Alternative räumliche Diskretisierungen

Seit den Anfängen der LBM wurde versucht, die Methode effizient auf irreguläre Gitter zu übertragen, da sich durch örtlich verfeinerte Diskretisierungen in den meisten Anwendungen viele Gitterpunkte einsparen lassen. Die meisten Lattice-Boltzmann-Codes (unter anderem PowerFLOW [1], Palabos [2], OpenLB [102], Musubi [88] und WaLBerla [62]) nutzen dazu reguläre hierarchische Verfeinerungen [53, 66, 134, 189]. Auf der einen Seite besteht der Vorteil von hierarchischen Verfeinerungen darin, dass der Strömungsschritt wie bei der Standard-LBM exakt bleibt. Der Zeitschritt δ_t wird lokal mit der Maschenweite des Gitters δ_x verkleinert, sodass auf jeder Verfeinerungsstufe die diskreten Teilchengeschwindigkeiten e_i mit dem Gitter gekoppelt sind:

$$\delta_x = \delta_t \xi_0,$$

wobei ξ_0 die Skalierung der diskreten Teilchengeschwindigkeiten bezeichnet. Auf der anderen Seite ist es jedoch unbefriedigend, dass weiterhin kartesische Gitterzellen verwendet werden müssen, während die Physik der Strömung oft nur eine Verfeinerung entlang einer Koordinate erfordert.

Aus diesem Grund wurden bis heute viele verschiedene *Off-Lattice-Boltzmann-Methoden* (OLBMs) entwickelt, welche die LBM auf irreguläre Gitter übertragen. Auf die erste Finite-Volumen-LBM im Jahr 1992 [171] folgten bis heute viele ähnliche Ansätze, die die Diskrete BGK-Gleichung mit Finite-Volumen-, Finite-Elemente, Finite-Differenzen-, Discontinuous-Galerkin- oder Spektral-Verfahren lösen [26, 78, 99, 142, 144, 147, 161, 171, 175]. Diese sogenannten Eulerschen OLBMs [142] entkoppeln das Rechengitter von der Diskretisierung des Geschwindigkeitsraums (siehe Abbildung 1.1). Alternativ zu diesen Eulerschen OLBMs wurden einige wenige interpolationsbasierte Ansätze vorgeschlagen [85, 93, 187, 196, 197]. Im Gegensatz

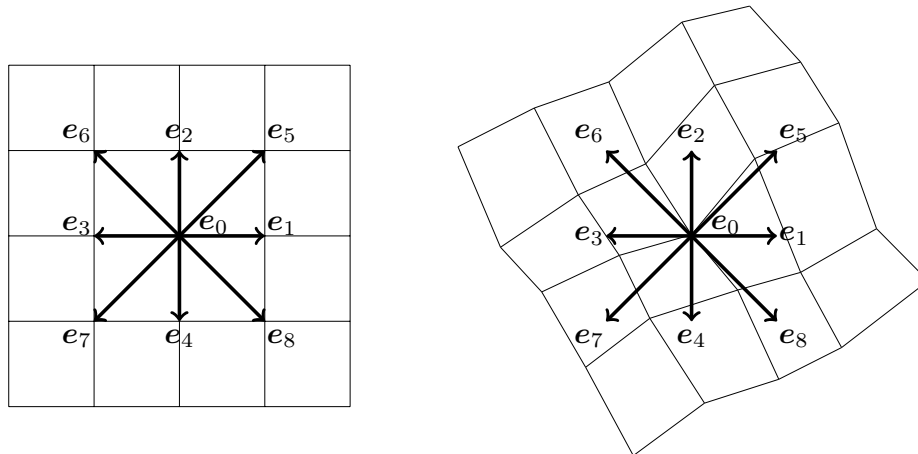


Abbildung 1.1: Die diskreten Geschwindigkeiten e_i sind in der LBM so gewählt, dass der Strömungsschritt auf einem regulären Gitter (links) die Verteilungsfunktionen exakt zwischen Gitterpunkten austauscht. Auf dem irregulären Gitter einer OLBM (rechts) ist das Gitter von der Diskretisierung des Geschwindigkeitsraums entkoppelt.

zu modernen Eulerschen LBMs [95, 144, 161] erlauben diese interpolationsbasierten LBMs aber weder eine höhere Konvergenzordnung noch große Zeitschritte. Weiterhin sind die meisten dieser Verfahren nicht effizient formuliert [85, 89, 197] oder erzeugen viel numerische Dissipation [135], sodass sie bis heute keine weitreichende Anwendung finden konnten.

Die OLBMs erhöhen einerseits die geometrische Flexibilität der LBM, sodass bei Simulationen in komplexen Geometrien viele Gitterpunkte eingespart werden können [165, 235]. Andererseits untergraben sie aber die Effizienz des Lattice-Boltzmann-Ansatzes gerade bei hochgradig transienten Strömungen. Dies zeigt sich beispielsweise in [14], wo die Simulation einer turbulenten Kanalströmung mit einer Finite-Volumen-LBM einen extrem kleinen Zeitschritt erfordert. Die defizitäre Performanz der Eulerschen OLBMs resultiert hauptsächlich aus drei Faktoren: Zunächst muss für jede diskrete Geschwindigkeit e_i eine lineare Advektionsgleichung gelöst werden, was gerade für dreidimensionale Strömungen aufwendig ist. Weiterhin verwenden die meisten OLBMs Runge-Kutta-Verfahren zur Zeitintegration, sodass die Ableitungsoperatoren mehrfach pro Strömungsschritt ausgewertet werden müssen. Am problematischsten ist aber die Courant-Friedrichs-Lewy (CFL)-Bedingung, die in Bezug auf die Teilchengeschwindigkeiten formuliert ist. Da diese die Strömungsgeschwindigkeit in der Regel um eine Größenordnung übersteigen, erzwingt die CFL-Bedingung in OLBMs sehr kleine Zeitschritte. Diese Schwächen der Eulerschen OLBMs sind besonders bei der Simulation von transienten Wirbelströmungen kritisch. Trotz vieler Versuche ist es also bisher nicht gelungen, die LBM so auf irreguläre Gitter zu übertragen, dass turbulente Strömungen effizient simuliert werden können.

Ziel und Aufbau der Arbeit

Die vorliegende Arbeit zielt darauf ab, die beiden Grundprobleme der Standard-LBM bei der Simulation inkompressibler Wirbelströmungen in effizienter Weise zu beheben. Um unteraufgelöste turbulente Rechnungen zu ermöglichen, wird eine *Pseudo-Entropische Stabilisierung* (PES) entwickelt, die Ideen der MRT-Methoden und der Entropischen LBM kombiniert. Der Stabilisierungsoperator der PES erhält die Effizienz und Lokalität des BGK-Kollisionsschritts und lässt sich flexibel mit anderen Kollisionsmodellen kombinieren. Anwendungen auf Beispielströmungen zeigen, dass die PES die Stabilität gegenüber der Standard-LBM drastisch verbessert und auch auf stark unteraufgelösten Gittern qualitativ korrekte Ergebnisse liefert. Zur Erweiterung der LBM auf irreguläre Gitter wird eine neuartige *Semi-Lagrangesche LBM* (SLLBM) entwickelt. Diese erlaubt als bisher einzige LBM die gleichzeitige Verwendung irregulärer Gitter und großer Zeitschritte sowie eine hohe Konvergenzordnung. Aufgrund dieser Schlüsseigenschaften gelingt der SLLBM im Vergleich mit einer anderen modernen OLBM [14, 80] eine vierfache Reduktion der Rechenzeit bei gleicher Qualität der Ergebnisse.

Die Arbeit gliedert sich wie folgt. Kapitel 2 gibt einen Überblick über die LBM und ihre physikalischen Grundlagen. Außerdem wird der im Rahmen dieser Arbeit entwickelte Simulationscode [126] beschrieben und auf zwei Beispielströmungen angewandt. In Kapitel 3 wird die PES formuliert. An zwei Beispielproblemen werden die exzellenten Stabilitätseigenschaften der neuen Methode demonstriert. Ein theoretischer Vergleich mit Kollisionsmodellen aus der Literatur verdeutlicht auch die Rolle der Entropie in bestehenden MRT-Modellen.

Kapitel 4 wendet sich dann den Eulerschen OLBMs zu. Hier wird besonders die Spektralelement-Discontinuous-Galerkin (SEDG)-LBM von Min und Lee [161] betrachtet und um verschiedene Zeitintegratoren erweitert. Ein Laufzeitvergleich mit der Standard-LBM zeigt, dass die Methode aufgrund ihrer hohen Konvergenzordnung und ihrer geometrischen Flexibilität Vorteile besitzt. An der Simulation einer turbulenten Kanalströmung werden aber auch die universellen Schwachpunkte Eulerscher OLBMs herausgearbeitet. Dies dient zur Motivation der neuartigen SLLBM, die in Kapitel 5 vorgestellt wird. Die in dieser Arbeit entwickelte SLLBM wird theoretisch analysiert und auf Standardprobleme der CFD angewandt. Ein Vergleich mit einer modernen Eulerschen OLBM [14, 80] demonstriert die Vorteile der SLLBM in Bezug auf Stabilität und Effizienz. Kapitel 6 fasst die Arbeit zusammen.

Eine weitere Neuerung findet sich in Anhang A, wo die LBM um Mehrschrittverfahren erweitert wird. Mit der entwickelten Mehrschritt-LBM gelingt eine Verbesserung der zeitlichen Konvergenzordnung der LBM, allerdings auf Kosten der bedingungslosen Stabilität. Dies kann mit der zweiten Dahlquist-Barriere [136] erklärt werden, wodurch sich zeigt, dass die Standard-LBM in Bezug auf ihre Zeitdiskretisierung optimal ist.

2. Strömungssimulation mit der Lattice-Boltzmann-Methode

2.1. Makroskopische Grundgleichungen

Klassische Methoden der Strömungssimulation lösen makroskopische Erhaltungsgleichungen. Diese partiellen Differentialgleichungen beschreiben die Dynamik von Flüssigkeiten und Gasen unter der Einwirkung äußerer und innerer Kräfte. Einen besonders wichtigen Spezialfall bilden hierbei reibungsbehaftete Strömungen, welche im Allgemeinen durch die Navier-Stokes-Gleichungen modelliert werden. Die approximative Lösung der Navier-Stokes-Gleichungen ist somit die zentrale Aufgabe der numerischen Strömungsmechanik [6].

Dieser Abschnitt widmet sich der Formulierung dieser Gleichungen, die sich zum Beispiel aus dem Reynolds-Transport-Theorem [131] herleiten lassen und die Erhaltung von Masse, Impuls und Energie beschreiben. Dabei wird, wie auch im weiteren Verlauf dieser Arbeit, die Einstein-Notation verwendet, das heißt, dass über doppelte Indizes eines Terms summiert wird. Für kompressible Strömungen lauten die Navier-Stokes-Gleichungen wie folgt:

- Massenerhaltung

$$\frac{\partial \rho}{\partial t} + \frac{\partial \rho u_i}{\partial x_i} = 0, \quad (2.1)$$

- Impulserhaltung

$$\frac{\partial \rho u_i}{\partial t} + \frac{\partial \rho u_i u_j}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \frac{\partial \tau_{ij}}{\partial x_j} + \rho F_i^{\text{ext}} \quad (2.2)$$

- und Energieerhaltung

$$\frac{\partial \rho E}{\partial t} + \frac{\partial \rho E u_j}{\partial x_j} = -\frac{\partial u_j p}{\partial x_j} - \frac{\partial q_j}{\partial x_j} + \frac{\partial u_i \tau_{ij}}{\partial x_j}$$

Dabei bezeichnet t die Zeit, \mathbf{x} den Vektor der räumlichen Koordinaten, ρ die Dichte, \mathbf{u} den Geschwindigkeitsvektor, p den Druck, $\boldsymbol{\tau}$ den Spannungstensor, \mathbf{F}^{ext} eine externe Kraft, E die totale Energie und \mathbf{q} den Wärmefluss. Die totale Energie setzt sich dabei aus innerer und kinetischer Energie zusammen.

In vielen praxisrelevanten Strömungen ist die Rückkopplung der Energieerhaltungsgleichung auf Massen- und Impulserhaltung vernachlässigbar. In diesen Situationen werden daher nur die zwei Letzteren gelöst. Diese vereinfachen sich weiter bei der Betrachtung inkompressibler Strömungen, wie beispielsweise flüssigen Wassers. Hier wird in der Regel der Stokesche Spannungstensor verwendet

$$\tau_{ij} = \mu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right),$$

wobei μ die dynamische Viskosität bezeichnet. Unter diesen Annahmen ergeben sich nach einigen Umformungen die inkompressiblen Navier-Stokes-Gleichungen:

- die inkompressible Kontinuitätsgleichung

$$\frac{\partial u_i}{\partial x_i} = 0$$

- und die inkompressible Impulsgleichung

$$\frac{\partial u_i}{\partial t} + u_j \frac{\partial u_i}{\partial x_j} = -\frac{1}{\rho} \frac{\partial p}{\partial x_i} + \nu \frac{\partial^2 u_i}{\partial x_j \partial x_j} + F_i^{\text{ext}}.$$

Die kinematische Viskosität $\nu = \mu/\rho$ ist dabei eine stoff- und temperaturabhängige Konstante. Die herausragende Wichtigkeit der Navier-Stokes-Gleichungen für die Strömungssimulation ergibt sich in erster Linie aus ihrer allgemeinen Gültigkeit. In den weitaus meisten Fällen modellieren sie die makroskopischen Effekte hinreichend genau, sowohl für zähflüssige Substanzen als auch für Gase [131].

Obwohl die inkompressiblen Navier-Stokes-Gleichungen sich für fast keine anwendungsrelevanten Strömungen analytisch lösen lassen, können durch Strömungssimulationen numerische Lösungen erzeugt werden. Dabei werden in aller Regel Standardverfahren der Numerik verwendet, wie zum Beispiel Finite-Differenzen-, Finite-Volumen- oder Finite-Elemente-Methoden. Diese Methoden diskretisieren entweder die Ableitungen in den oben beschriebenen Gleichungen oder die Integrale in den äquivalenten Integralgleichungen. Daraus ergeben sich große nichtlineare Differentialgleichungssysteme.

Die zeitabhängige Lösung dieser Systeme erfordert numerische Zeitintegratoren und nichtlineare Löser, welche in jedem Iterationsschritt das Geschwindigkeits- und Druckfeld berechnen. Letzteres kann über verschiedene Ansätze bestimmt werden, die je nach Verfahren die Lösung einer Poisson-Gleichung [37, 87] oder die Einführung künstlicher Kompressibilität [36] erfordern. Während die Poisson-Gleichungen

(zum Beispiel in Projektionsmethoden [37]) in jedem Zeitschritt iterativ gelöst werden müssen, erzeugt die künstliche Kompressibilität numerische Fehler, die nur für stationäre Strömungen verschwinden [132]. Insgesamt ist die effiziente numerische Lösung der Navier-Stokes-Gleichungen nur mit einem hohen algorithmischen Aufwand zu realisieren.

Im Gegensatz zu klassischen CFD-Lösern [36, 37, 87] nimmt die Lattice-Boltzmann-Methode einen Umweg über die zugrundeliegende Partikeldynamik. Zwar baut ihre mesoskopische Modellierung auf der kinetischen Gastheorie auf [86]. Wegen ihrer Konsistenz mit den Navier-Stokes-Gleichungen im Kontinuumsbereich [113] ist sie jedoch auch zur Simulation von Flüssigkeiten geeignet. Bis heute ist der primäre Anwendungsbereich der LBM die Simulation inkompressibler Strömungen [208], da die Modellierung der LBM die Lösung einer Poisson-Gleichung und den damit verbundenen Aufwand umgeht.

2.2. Kinetische Gastheorie

Da die kinetische Gastheorie die Grundlage der Lattice-Boltzmann-Modellierung bildet, werden im Folgenden die wesentlichen Aspekte dieser Theorie skizziert. Detailliertere Ausführungen finden sich zum Beispiel in [86].

In der kinetischen Gastheorie wird ein Gas durch die molekulare Geschwindigkeitsverteilungsfunktion $f(t, \mathbf{x}, \boldsymbol{\xi})$ modelliert. Diese beschreibt die Dichte der Moleküle mit Geschwindigkeit $\boldsymbol{\xi}$ zum Zeitpunkt t an einem Ort \mathbf{x} . Im thermodynamischen Gleichgewicht lässt sich diese Dichtefunktion durch eine skalierte multivariate Normalverteilung beschreiben, die Maxwell-Boltzmann-Verteilung

$$f_{\text{MB}}^{\text{eq}} = \frac{\rho}{(\sqrt{2\pi} c_s)^3} \exp\left(-\frac{(\boldsymbol{\xi} - \mathbf{u}_i)(\boldsymbol{\xi} - \mathbf{u}_i)}{2c_s^2}\right).$$

Hierbei bezeichnet c_s die Schallgeschwindigkeit.

Durch die Skalierung der Normalverteilung mit der Dichte ρ ergeben sich wichtige makroskopische Größen als Momente dieser Verteilung, wie beispielsweise

$$\rho = \int_{\mathbb{R}^3} f_{\text{MB}}^{\text{eq}} d\boldsymbol{\xi} \quad \text{und} \quad \rho \mathbf{u} = \int_{\mathbb{R}^3} f_{\text{MB}}^{\text{eq}} \boldsymbol{\xi} d\boldsymbol{\xi}.$$

Für Abweichungen vom thermodynamischen Gleichgewicht lässt sich die Dynamik der molekularen Geschwindigkeitsverteilungsfunktion durch die Boltzmann-Gleichung beschreiben:

$$\frac{\partial f}{\partial t} + \xi_i \frac{\partial f}{\partial x_i} + F_i^{\text{ext}} \frac{\partial f}{\partial \xi_i} = \int_{\tilde{\boldsymbol{\xi}}} \int_{A_c} (f' \tilde{f}' - f \tilde{f}) (\tilde{\boldsymbol{\xi}} - \boldsymbol{\xi}) dA_c d\tilde{\boldsymbol{\xi}}.$$

Während die linke Seite die Advektion der Geschwindigkeitsverteilungsfunktion beschreibt, modelliert die rechte Seite die Kollision von Teilchen. Hierbei bezeichnen

$f' = f(t, \mathbf{x}, \boldsymbol{\xi}')$ und $\tilde{f}' = f(t, \mathbf{x}, \tilde{\boldsymbol{\xi}}')$ die Teilchenverteilungen nach der Kollision und A_c den differentiellen Wirkungsquerschnitt.

Das Kollisionsintegral berücksichtigt dabei nur die elastische Kollision zweier Teilchen und vernachlässigt kompliziertere intermolekulare Wechselwirkungen. Aus diesem Grund eignet sich die Boltzmann-Gleichung nur zur Beschreibung von Gasen. Um Flüssigkeiten zu modellieren, müsste der Kollisionsterm durch einen komplizierteren Ausdruck beschrieben werden, wie zum Beispiel in der BBGKY-Hierarchie [17]. Die Lattice-Boltzmann-Methode möchte jedoch nur die makroskopische Dynamik beschreiben und zielt nicht auf eine genaue Modellierung der mikroskopischen Effekte ab. Vielmehr wird eine minimale Beschreibung der Kinetik bevorzugt, um die Rechnung möglichst effizient zu gestalten.

In diesem Sinne wird die Boltzmann-Gleichung weiter vereinfacht. Das H-Theorem besagt, dass die Teilchenkollisionen in ihrer Summe einen Anstieg der Entropie bewirken. Da die Maxwell-Boltzmann-Verteilung die Entropie maximiert, lassen sich die Kollisionen als eine Relaxation hin zu dieser Gleichgewichtsverteilung pauschalisieren. Dies wird beschrieben durch die Bhatnagar-Gross-Krook (BGK)-Näherung [12]

$$\frac{\partial f}{\partial t} + \xi_i \frac{\partial f}{\partial x_i} + F_i^{\text{ext}} \frac{\partial f}{\partial \xi_i} = -\frac{1}{\lambda} (f - f_{\text{MB}}^{\text{eq}}),$$

wobei die Relaxationszeit λ sich aus der kinematischen Viskosität und der Schallgeschwindigkeit berechnet:

$$\lambda = \frac{\nu}{c_s^2}.$$

Sowohl für die Boltzmann-Gleichung als auch für die BGK-Gleichung lässt sich mithilfe einer Multiskalenanalyse zeigen, dass die nullten und ersten Momente ihrer Lösungen, ρ und $\rho \mathbf{u}$, in einem bestimmten Grenzbereich asymptotisch gegen Lösungen der Navier-Stokes-Gleichungen konvergieren. Diese sogenannte Chapman-Enskog-Analyse [29, 86] wird in Anhang B für die diskretisierte Version der BGK-Gleichung exemplarisch durchgeführt. Diese asymptotische Eigenschaft der BGK-Gleichung ermöglicht ihre Verwendung in der Strömungssimulation und bildet daher die Grundlage der Lattice-Boltzmann-Methode.

2.3. Dimensionslose Kennzahlen

Wesentliche Aspekte der beschriebenen Gleichungen lassen sich über dimensionslose Kennzahlen ausdrücken. Die wichtigste Kennzahl in der Strömungsmechanik ist die Reynoldszahl. Diese beschreibt das Verhältnis von Trägheitskräften und viskosen Kräften in einer Strömung und ist definiert als

$$Re = \frac{u_0 l_0}{\nu}.$$

Dabei bezeichnen u_0 und l_0 die charakteristische Geschwindigkeit und die charakteristische Länge. Die Reynolds-Zahl stellt also eine Gewichtung des nichtlinearen Terms der Impulsgleichung gegenüber dem viskosen Term dar. Während der nichtlineare Term den Zerfall von großen Wirbeln in kleinere Wirbel begünstigt, dämpft der viskose Term die Verwirbelung ab. Deshalb sind Strömungen mit hoher Reynoldszahl im Allgemeinen von kleinskaligen Wirbelstrukturen geprägt. Strömungen mit gleicher Reynoldszahl verhalten sich *Reynolds-ähnlich*, was zum Beispiel Versuche im Windkanal mit verkleinerten Modellen rechtfertigt.

Eine weitere wichtige dimensionslose Kennzahl ist die Machzahl

$$Ma = \frac{u_0}{c_s},$$

die das Verhältnis aus charakteristischer Geschwindigkeit und Schallgeschwindigkeit bezeichnet. Damit stellt sie ein Maß für die Kompressibilität einer Strömung dar.

Für die kinetische Gastheorie ist darüber hinaus die Knudsen-Zahl von Bedeutung. Diese ist definiert als

$$Kn = \frac{l_m}{l_0},$$

wobei l_m die mittlere freie Weglänge bezeichnet, also die Strecke, die ein Teilchen im Mittel zwischen zwei Kollisionen zurücklegt. Somit beschreibt die Knudsen-Zahl das Verhältnis zwischen der Mikro-Skala der Teilchenkollisionen und der charakteristischen Skala des betrachteten Systems und stellt ein Maß für die Abweichung vom Gleichgewicht dar. Für $Kn \ll 1$ ist ein System als Kontinuum zu betrachten; in diesem Limit $Kn \rightarrow 0$ verschwinden die Nichtgleichgewichtsanteile der Geschwindigkeitsverteilungsfunktion.

In der kinetischen Gastheorie sind die drei beschriebenen Kennzahlen über die Beziehung

$$Kn \sim \frac{Ma}{Re}$$

miteinander gekoppelt [86].

2.4. Lattice-Boltzmann-Methode

Die Lattice-Boltzmann-Methode in ihrer Standardform löst eine diskretisierte Form der BGK-Gleichung, die sogenannte Lattice-BGK-Gleichung. Um diese Gleichung herzuleiten, werden im Folgenden der Geschwindigkeitsraum, die Zeitachse und der kartesische Raum diskretisiert. Dabei wird die Transformation der diskreten Geschwindigkeitsverteilungen eingeführt, eine Grundvoraussetzung zur effizienten Simulation turbulenter Strömungen mit der LBM. Des Weiteren wird der Zusammenhang zwischen der Lattice-Boltzmann-Methode und den Navier-Stokes-Gleichungen verdeutlicht.

2.4.1. Diskretisierung des Geschwindigkeitsraums in 1D

Die Diskretisierung des Geschwindigkeitsraums resultiert aus einer Gauß-Hermite-Quadratur der Maxwell-Boltzmann-Verteilung. In dieser Weise kann die BGK-Gleichung auf eine endliche Anzahl von Geschwindigkeiten reduziert werden. Dies wird hier nur im Eindimensionalen hergeleitet, die Rechnung kann aber direkt auf zwei- und dreidimensionale Strömungen übertragen werden, siehe zum Beispiel [92].

Dazu wird die eindimensionale Maxwell-Boltzmann-Verteilung wie folgt umgeschrieben:

$$\begin{aligned}
 f_{1D}^{\text{eq}}(\xi) &= \frac{\rho}{\sqrt{2\pi} c_s} \exp\left(-\frac{(\xi - u)^2}{2c_s^2}\right) \\
 &= \frac{\rho}{\sqrt{2\pi} c_s} \exp\left(-\frac{\xi^2}{2c_s^2}\right) \exp\left(\frac{2\xi u - u^2}{2c_s^2}\right) \\
 &\approx \frac{1}{c_s \sqrt{2\pi}} \exp\left(-\frac{\xi^2}{2c_s^2}\right) g^{\text{eq}}(\xi).
 \end{aligned} \tag{2.3}$$

Das Polynom

$$g^{\text{eq}}(\xi) = \rho \left(1 + \frac{2\xi u - u^2}{2c_s^2} + \frac{1}{2} \left(\frac{\xi u}{c_s^2} \right)^2 \right)$$

entsteht hierbei aus einer Taylor-Entwicklung der Exponentialfunktion, wobei nur Terme bis zur Ordnung $O(u^2)$ berücksichtigt werden [186]. Ebenso werden Nichtgleichgewichtsverteilungen durch beliebige Polynome zweiten Grades g ausgedrückt.

Zur Integration dieser Approximation der Form $\exp(-\zeta^2)g(\zeta)$ bietet sich die Gauß-Hermite-Quadratur an, welche eigens für Integrale dieser Art definiert ist:

Lemma 2.1 *Die Gauß-Hermite-Quadratur mit den Stützstellen $\zeta_0 := 0$, $\zeta_1 := \sqrt{3/2}$ und $\zeta_2 := -\sqrt{3/2}$ sowie den Gewichten $\alpha_0 := 2\sqrt{\pi}/3$ und $\alpha_1 = \alpha_2 = \sqrt{\pi}/6$ ist für Polynome $p(\zeta)$ vom Grad ≤ 5 exakt, das heißt*

$$\int_{-\infty}^{\infty} e^{-\zeta^2} p(\zeta) d\zeta = \alpha_i p(\zeta_i).$$

□

Mit der Substitution $\zeta = \xi/(\sqrt{2}c_s)$ erhält man aus Gleichung (2.3) die folgende Approximation:

$$f_{1D}(e_i) = \frac{1}{c_s \sqrt{2\pi}} \alpha_{(i)} \cdot \sqrt{2} c_s g(e_i) = \left\{ \begin{array}{ll} 2/3 \cdot g(e_i), & i = 0 \\ 1/6 \cdot g(e_i), & i = 1, 2 \end{array} \right\} =: f_i.$$

Dabei sind die Stützstellen

$$e_i = \sqrt{2} c_s \zeta_i = \begin{cases} 0, & i = 0 \\ \sqrt{3} c_s, & i = 1 \\ -\sqrt{3} c_s, & i = 2 \end{cases} .$$

Die oben berechneten Gewichtungsfaktoren werden in der Folge mit $\omega_i := f_i/g(e_i)$ bezeichnet. Diese Diskretisierung fasst man unter dem Namen D1Q3-Stern zusammen, da sie im Eindimensionalen mit drei diskreten Geschwindigkeiten operiert. Q bezeichnet damit im Folgenden die Anzahl der diskreten Geschwindigkeiten. Die relevanten Größen sind noch einmal in Tabelle 2.1 zusammengefasst.

Tabelle 2.1: Diskrete Geschwindigkeiten e_i und Gewichte ω_i in der D1Q3-Diskretisierung

i	0	1	2
e_i	0	$\sqrt{3} c_s$	$-\sqrt{3} c_s$
ω_i	2/3	1/6	1/6

2.4.2. Diskretisierung des Geschwindigkeitsraums in 2D und 3D

Tabelle 2.2: Diskrete Geschwindigkeiten e_i und Gewichte ω_i in der D2Q9-Diskretisierung. ξ_0 bezeichnet die Skalierung der diskreten Geschwindigkeiten.

i	0	1	2	3	4	5	6	7	8
e_{i1}/ξ_0	0	1	0	-1	0	1	-1	-1	1
e_{i2}/ξ_0	0	0	1	0	-1	1	1	-1	-1
ω_i	4/9	1/9	1/9	1/9	1/9	1/36	1/36	1/36	1/36

Durch Verwendung von Mehrfachintegralen kann dieses Vorgehen auf die zweidimensionale und dreidimensionale BGK-Gleichung übertragen werden [92]. Konkret werden in dieser Arbeit die D2Q9- und die D3Q19-Diskretisierung verwendet. Diese sind in den Tabellen 2.2 und 2.3 definiert. Dabei wird die Skalierung des Geschwindigkeitssterns als $\xi_0 := \sqrt{3} c_s$ definiert.

Die diskrete Gleichgewichtsverteilung (2.3) schreibt sich unabhängig von der Dimension als

$$f_i^{\text{eq}} = \omega_{(i)} \rho \left(1 + \frac{2e_{(i)j} u_j - u_j u_j}{2c_s^2} + \frac{1}{2} \left(\frac{e_{(i)j} u_j}{c_s^2} \right)^2 \right). \quad (2.4)$$

Tabelle 2.3: Diskrete Geschwindigkeiten e_i und Gewichte ω_i in der D3Q19-Diskretisierung. ξ_0 bezeichnet die Skalierung der diskreten Geschwindigkeiten.

i	0	1	2	3	4	5	6	7	8	9
e_{i1}/ξ_0	0	1	0	-1	0	0	0	1	-1	-1
e_{i2}/ξ_0	0	0	0	0	0	-1	1	0	0	0
e_{i3}/ξ_0	0	0	1	0	-1	0	0	1	1	-1
ω_i	1/3	1/18	1/18	1/18	1/18	1/18	1/18	1/36	1/36	1/36
i	10	11	12	13	14	15	16	17	18	
e_{i1}/ξ_0	1	1	1	-1	-1	0	0	0	0	
e_{i2}/ξ_0	0	-1	1	1	-1	-1	1	1	-1	
e_{i3}/ξ_0	-1	0	0	0	0	1	1	-1	-1	
ω_i	1/36	1/36	1/36	1/36	1/36	1/36	1/36	1/36	1/36	

Das Endresultat dieser Diskretisierung ist eine diskrete Form der BGK-Gleichung

$$\frac{\partial f_i}{\partial t} + e_{(i)j} \frac{\partial f_{(i)}}{\partial x_j} = \Omega_i(f) = -\frac{1}{\lambda}(f_i - f_i^{\text{eq}}). \quad (2.5)$$

(Über geklammerte Indizes wird nicht summiert.) $\Omega = (\Omega_i)_i$ bezeichnet dabei den Kollisionsterm.

Die Verwendung der Gauß-Hermite-Quadratur zur Diskretisierung des Geschwindigkeitsraums hat dabei einige wichtige Implikationen. Zunächst ergeben sich einige wichtige Eigenschaften der Lattice-Tensoren $\sum_i \omega_{(i)} e_{(i)j} e_{(i)k} \dots$, die im Anhang in Lemma B.1 aufgelistet sind. Die Quadratur ist exakt für alle Polynome bis zum Grad fünf. Damit lassen sich alle relevanten Momente der diskreten Geschwindigkeitsverteilungen exakt berechnen. Zum Beispiel ergeben sich die Dichte und Geschwindigkeit als

$$\rho = \sum_{i=0}^{Q-1} f_i$$

$$\rho \mathbf{u} = \sum_{i=0}^{Q-1} f_{(i)} \mathbf{e}_{(i)}.$$

Die zweiten und dritten Momente der Gleichgewichtsverteilung sind außerdem in Anhang B aufgeführt.

Obwohl die Momente durch die Hermite-Quadratur exakt berechnet werden können, löst die diskrete BGK-Gleichung natürlich eine radikal vereinfachte Teilchendynamik. Trotz dieser Simplifizierung der mesoskopischen Vorgänge approximieren die Momente aber Lösungen der Navier-Stokes-Gleichungen. Deshalb kann die LBM – obwohl

in der Gastheorie verankert – auch zur Simulation von Flüssigkeiten und sogar darüber hinaus verwendet werden. Der folgende Unterabschnitt skizziert die Beziehung zwischen den gaskinetischen Gleichungen und den makroskopischen Erhaltungsgleichungen.

2.4.3. Von der Gaskinetik zu den Navier-Stokes-Gleichungen

Die hydrodynamischen Limiten der gaskinetischen Gleichungen lassen sich über verschiedene Wege herleiten [28]. Oft wird dazu die Geschwindigkeitsverteilungsfunktion in eine asymptotische Reihe entwickelt:

$$f = f^{(0)} + \varepsilon f^{(1)} + \varepsilon^2 f^{(2)} + \dots$$

Dabei bezeichnet $f^{(0)}$ die Gleichgewichtsverteilung und die höheren Glieder jeweils Abweichungen verschiedener Ordnung vom Gleichgewicht. Der Parameter ε kann mit der Knudsen-Zahl identifiziert werden, da die Knudsen-Zahl ein Maß für die Abweichung vom Gleichgewicht darstellt. Im hydrodynamischen Limit liegt ein Kontinuum vor, in dem die Knudsen-Zahl und damit ε sehr klein werden. Da sich die wesentlichen hydrodynamischen Effekte allerdings in den Nichtgleichgewichtsanteilen manifestieren, müssen diese in der Analyse besonders berücksichtigt werden.

Bei der Chapman-Enskog-Analyse [29] werden die Ableitungsoperatoren als $\partial_x = \varepsilon \partial_{x_1}$ und $\partial_t = \varepsilon \partial_{t_1} + \varepsilon^2 \partial_{t_2}$ umdefiniert. Dadurch wird den mikroskopischen Einflüssen auf große Zeit- und Längenskalen Rechnung getragen. Über die Aufspaltung der Zeitableitung kann die Kopplung physikalischer Effekte analysiert werden, die sich gaskinetisch auf verschiedenen Skalen abspielen [28]. Diese Mehrskalenganalyse wurde ursprünglich für die Boltzmann- und die BGK-Gleichung entwickelt und mit dem Aufkommen von Lattice-Boltzmann-Verfahren auf diese übertragen [186]. Obwohl sie mathematisch auf sehr losen Annahmen basiert [218], hat sie sich im Kontext der LBM bewährt, um makroskopische Erhaltungsgleichungen für diskrete kinetische Modelle herzuleiten [30, 129]. Die Chapman-Enskog-Analyse der diskreten BGK-Gleichung wird exemplarisch in Anhang B durchgeführt.

Verschiedene Autoren haben Alternativen zur Chapman-Enskog-Analyse vorgeschlagen. So wird zum Beispiel bei Junk [113, 114] auf eine Aufspaltung der Zeitskala verzichtet. Diese wird ersetzt durch eine diffusive Skalierung $\partial_x \sim \varepsilon$, $\partial_t \sim \varepsilon^2$. Während die Chapman-Enskog-Analyse zu den kompressiblen Navier-Stokes-Gleichungen führt, erhält man durch die diffusive Skalierung die inkompressiblen Navier-Stokes-Gleichungen.

In [107] wird statt der asymptotischen Entwicklung der Geschwindigkeitsverteilung sogar eine Taylor-Entwicklung der Lattice-Boltzmann-Gleichung durchgeführt, um Fehlerabschätzungen für die LBM herzuleiten. Dieser Ansatz kommt mit schwächeren Grundannahmen aus, eine Erweiterung auf allgemeinere kinetische Modelle liegt aber nicht auf der Hand.

Insgesamt bleibt die mathematisch rigorose Analyse gaskinetischer Gleichungen ein aktives Forschungsthema [218]. Durchschlagende Ergebnisse wurden im vergangenen Jahrzehnt von Saint-Raymond und Golse erzielt [75, 191]. Hier wurden die BGK-Gleichung und die Boltzmann-Gleichung unter vergleichsweise schwachen Grundannahmen in die Navier-Stokes-Gleichungen überführt. Es ist jedoch nicht klar, inwiefern dieser Ansatz auf die diskrete BGK-Gleichung und damit auf die LBM übertragbar ist.

Trotz der problematischen theoretischen Analyse hat sich die LBM in zahlreichen praktischen Anwendungen als Strömungslöser bewährt [3, 30, 56, 78, 107, 113, 129, 188, 207]. Die praktisch erzielten Konvergenzraten stimmen dabei gut mit theoretischen Voraussagen überein [107, 113, 188]. Die Betrachtung der verschiedenen Fehlerterme wird im weiteren Verlauf dieser Arbeit konkretisiert und vertieft.

2.4.4. Zeitliche Diskretisierung

Die Chapman-Enskog-Analyse der diskreten BGK-Gleichung (2.5) zeigt, dass die Relaxationszeit wie in der BGK-Gleichung über $\lambda = \nu/c_s^2$ mit der Viskosität gekoppelt ist. Bei der numerischen Lösung führt dies zu Problemen, da der Vorfaktor des Kollisionsterms betragsmäßig sehr groß wird und dadurch sehr kleine Zeitschritte erzwungen werden [141].

Diese Problematik kann durch eine Transformation der Verteilungsfunktionen umgangen werden [10, 81, 186]. Diese wird in der Literatur typischerweise für die diskrete BGK-Gleichung durchgeführt [10, 161]. Die folgende Herleitung überträgt diese Transformation auf eine allgemeinere Klasse von Kollisionsmodellen und findet sich in ähnlicher Form bei Dellar [42, 43].

Zunächst wird die diskrete Boltzmann-Gleichung mit einem diskreten Zeitschritt δ_t entlang ihrer Charakteristiken $(\delta_t, \delta_t \mathbf{e}_i)$ integriert. Dabei wird das Integral des Kollisionsterms mit der Trapezregel approximiert, woraus folgt

$$f_{(i)}(t + \delta_t, \mathbf{x} + \delta_t \mathbf{e}_{(i)}) - f_i(t, \mathbf{x}) = \frac{\delta_t}{2} [\Omega_{(i)}(f(t + \delta_t, \mathbf{x} + \delta_t \mathbf{e}_{(i)})) + \Omega_i(f(t, \mathbf{x}))]. \quad (2.6)$$

Es werden nun Kollisionsmodelle der Form $\Omega(f) = Rf + R^{\text{eq}}(f)$ betrachtet, wobei R ein linearer Operator ist und der nichtlineare Anteil R^{eq} die Identität

$$R^{\text{eq}}(f) = R^{\text{eq}}(\bar{f})$$

erfüllt. Die transformierten Verteilungsfunktionen \bar{f} sind dabei definiert als

$$\bar{f} = f - \frac{\delta_t}{2} \Omega(f). \quad (2.7)$$

Diese Transformation wird in Gleichung (2.6) eingesetzt

$$\bar{f}_{(i)}(t + \delta_t, \mathbf{x} + \delta_t \mathbf{e}_{(i)}) - \bar{f}_i(t, \mathbf{x}) = \delta_t \Omega_i(f(t, \mathbf{x})). \quad (2.8)$$

Als Letztes muss der Kollisionsoperator als Funktion der transformierten Geschwindigkeitsverteilungen ausgedrückt werden. Dazu wird zuerst die oben definierte Form des Operators genutzt

$$\Omega(f) = Rf + R^{\text{eq}}(f) = R\left(\bar{f} + \frac{\delta_t}{2}\Omega(f)\right) + R^{\text{eq}}(\bar{f}) = \Omega(\bar{f}) + \frac{\delta_t}{2}R\Omega(f).$$

Somit ergibt sich

$$\Omega(f) = \left(I - \frac{\delta_t}{2}R\right)^{-1} \Omega(\bar{f})$$

und durch Einsetzen in Gleichung (2.8) die Lattice-Boltzmann-Gleichung

$$\left(\bar{f}_{(i)}(t + \delta_t, \mathbf{x} + \delta_t \mathbf{e}_{(i)}) - \bar{f}_i(t, \mathbf{x})\right)_i = \delta_t \left(I - \frac{\delta_t}{2}R\right)^{-1} \Omega(\bar{f}) =: \bar{\Omega}_{\delta_t}(\bar{f}).$$

Für den BGK-Kollisionsoperator führt diese Transformation auf die sogenannte Lattice-BGK-Gleichung [186]

$$\bar{f}_i(t + \delta_t, \mathbf{x} + \delta_t \mathbf{e}_i) - \bar{f}_i(t, \mathbf{x}) = -\frac{1}{\tau}(\bar{f}_i - \bar{f}_i^{\text{eq}}), \quad (2.9)$$

wobei

$$\tau = \frac{\lambda}{\delta_t} + \frac{1}{2} = \frac{\nu}{c_s^2 \delta_t} + \frac{1}{2}$$

der dimensionslose Relaxationsparameter ist.

Diese Transformation erlaubt eine explizite Behandlung des Kollisionsterms, die aber den Charakter eines impliziten Verfahrens hat. Dadurch verbessert sich die Stabilität des Verfahrens für große Reynoldszahlen drastisch. Außerdem führt die Transformation zu einer Erhöhung der Konvergenzordnung ohne zusätzlichen rechnerischen Aufwand.

In der frühen Entwicklung von Off-Lattice-Boltzmann-Verfahren wurden diese Aspekte oft übersehen [10], siehe zum Beispiel [141, 160, 231]. Letztendlich sind sie aber ein wesentlicher Faktor für den Erfolg der LBM [81]. In Anhang A wird gezeigt, dass sich eine solche Transformation auch für implizite Mehrschrittverfahren herleiten lässt. Damit können die Stabilitätseigenschaften und die Konvergenzordnung der LBM verbessert werden.

Im weiteren Verlauf der Arbeit werden wegen ihrer offensichtlichen Vorteile die transformierten Geschwindigkeitsverteilungsfunktionen verwendet. Diese werden in der Folge mit f statt \bar{f} bezeichnet, um in der Notation konsistent mit der Literatur zu bleiben. Aus dem gleichen Grund wird in der Folge Ω statt $\bar{\Omega}_{\delta_t}$ verwendet.

2.4.5. Formulierung der Lattice-Boltzmann-Methode

Die LBM löst die Lattice-BGK-Gleichung (2.9) auf regulären Gittern, den *Lattices*. Diese räumliche Diskretisierung begründet sich historisch im Ursprung der LBM aus den zellulären Automaten für Lattice-Gase [103, 104, 159, 229]. Darüber hinaus sind reguläre Gitter auch die natürliche Wahl im Hinblick auf die Diskretisierung des Geschwindigkeitsraums.

In der Regel setzt man dabei die Schallgeschwindigkeit auf $c_s = 1/\sqrt{3}$, sodass die Komponenten der diskreten Geschwindigkeiten (siehe Tabellen 2.1, 2.2 und 2.3) ganzzahlige Werte annehmen. Setzt man nun $\delta_t = \delta_x$, so lässt sich die Lattice-Boltzmann-Gleichung auf einem regulären Gitter mit Maschenweite δ_x ohne numerische Differentiation oder Interpolationen berechnen.

Im Gegensatz zu dieser kanonischen Wahl werden in dieser Arbeit die Partikelgeschwindigkeiten statt der Strömungsgeschwindigkeit skaliert, um die Machzahl einzustellen. Beide Methoden sind äquivalent. Auf einem regulären Gitter macht die Wahl $c_s = 1/\sqrt{3}$ aber mehr Sinn, weil die Indizes der Gitterpunkte dann mit den skalierten Ortskoordinaten übereinstimmen. Da der Großteil der in dieser Arbeit untersuchten und entwickelten Methoden auf irregulären Gittern operiert, ist eine Skalierung der Partikelgeschwindigkeiten naheliegend. Dies umgeht die Umrechnung der Strömungen in Lattice-Einheiten.

Die Lattice-Boltzmann-Gleichung wird in zwei Schritten gelöst:

1. Kollision $f_i^{\text{pc}}(t, \mathbf{x}) := \Omega_i(f(t, \mathbf{x}))$ und
2. Strömung (Advektion) $f_{(i)}(t + \delta_t, \mathbf{x} + \delta_t \mathbf{e}_{(i)}) = f_i^{\text{pc}}(t, \mathbf{x})$.

Hierbei ist $\mathbf{x} + \delta_t \mathbf{e}_i$ ein benachbarter Lattice-Knoten. Einerseits führt diese Art der Diskretisierung zu einem effizienten und genauen Verfahren, das die Entwicklung von Phasen- und Amplitudenfehlern im Strömungsschritt ausschließt. Andererseits beschränken die regulären Gitter die geometrische Flexibilität und die Fehlerordnung der LBM. Im Verlauf dieser Arbeit spielen die Entwicklung von Off-Lattice-Boltzmann-Verfahren auf irregulären Gitter sowie die Erhöhung der Fehlerordnung eine große Rolle.

2.4.6. Fehlerbetrachtung

Bevor jedoch Erweiterungen der LBM untersucht und entwickelt werden, sollen zunächst die Diskretisierungsfehler der Standard-Methode beschrieben werden. Da die von der Standard-LBM implizierten hydrodynamischen Gleichungen nur im inkompressiblen Limit den Navier-Stokes-Gleichungen entsprechen, beschränkt sich die Konvergenzbetrachtung auf diesen Fall.

Die Chapman-Enskog-Analyse der diskreten BGK-Gleichung (siehe Anhang B) liefert einen Kompressibilitätsfehler der Ordnung $O(Ma^2)$ in Bezug auf die Navier-Stokes-Gleichungen im inkompressiblen Limit. Dies stimmt mit der theoretischen und praktischen Analyse von Junk et al. [113] überein und gilt sowohl für den Druck als auch für die Geschwindigkeit. In einer anderen Schreibweise findet sich die gleiche Ordnung auch in der theoretischen Analyse von Holdych et al. [107]. Hier tauchen Fehlerterme der Ordnung $(\delta_t/\delta_x)^2$ in Bezug auf die Impulsgleichung auf. Wegen $\delta_x \sim \delta_t c_s$ sind diese Terme zweiter Ordnung in der Machzahl. Eine Konsequenz dieser Überlegung ist, dass eine reine Gitterverfeinerung – die sogenannte akustische Skalierung $\delta_t \sim \delta_x$, $Ma = \text{const}$ – den Kompressibilitätsfehler konstant hält. Deshalb konvergieren die Lösungen in dieser Skalierung im Allgemeinen nicht gegen Lösungen der Navier-Stokes-Gleichungen. Die Konvergenz kann jedoch zum Beispiel in einer diffusiven Skalierung $\delta_t \sim \delta_x^2 \sim Ma^2$ erreicht werden, siehe unten.

Die räumliche Diskretisierung durch reguläre Gitter induziert einen Fehler der Ordnung $O(\delta_x^2)$. Dies folgt unmittelbar aus der Taylor-Entwicklung der räumlichen Ableitung in der Chapman-Enskog-Analyse der Lattice-Boltzmann-Gleichung [30] und ergibt sich in gleicher Weise für die anderen genannten Analysen [3, 107, 113, 205].

Die Transformation der Verteilungsfunktionen in Unterabschnitt 2.4.4 bedingt einen Fehler der Ordnung $O(\delta_t^2)$ durch die Verwendung der Trapezregel. Die gleiche zeitliche Ordnung findet man auch in der Chapman-Enskog-Analyse der Lattice-Boltzmann-Gleichung [30]. Somit ergibt sich ein Gesamtfehler der Ordnung $O(Ma^2 + \delta_x^2 + \delta_t^2)$. Deshalb wird die LBM zum Beispiel in [30] als Verfahren zweiter Ordnung betrachtet.

Bei Junk et al. [113] findet man jedoch einen zeitlichen Fehler erster Ordnung. Dies hängt mit der Betrachtungsweise zusammen. Im Rahmen der Standard-LBM sind der Zeitschritt und die Mach-Zahl über $\delta_t \sim \delta_x Ma$ miteinander gekoppelt. Somit addieren sich die Fehlerordnungen in der diffusiven Skalierung $\delta_t \sim \delta_x^2$ zu

$$O(Ma^2 + \delta_x^2 + \delta_t^2) = O\left(\left(\frac{\delta_t}{\delta_x}\right)^2 + \delta_t^2 + \delta_x^2\right) = O(\delta_x^2 + \delta_t^2 + \delta_t) = O(\delta_x^2 + \delta_t).$$

Aus diesem Grund verhält sich die LBM in der diffusiven Skalierung tatsächlich wie ein Verfahren erster Ordnung [138] in der Zeit. Dabei ist jedoch in Betracht zu ziehen, dass der tatsächliche Einfluss des Kompressibilitätsfehlers sehr stark von der konkreten Strömung abhängt [138].

Da in dieser Arbeit nicht die charakteristische Geschwindigkeit, sondern die Partikelgeschwindigkeiten mit der Machzahl skaliert werden, ändert sich hier die Formulierung dieses Fehlers, nicht aber die Essenz dieser Überlegung. In der hier gewählten Formulierung überträgt sich die Konvergenz erster Ordnung äquivalent auf die Courant-Friedrich-Lewy (CFL)-Zahl. Die CFL-Zahl ist dabei definiert als

$$CFL = \frac{\delta_t}{\min \delta_x} \max_{(i)} |e_{(i)}|.$$

Bei konstanter CFL-Zahl ist das Verfahren dann wieder erster Ordnung in der Zeit. Mit dieser Fehlerbetrachtung schließt die Beschreibung und Analyse der Standard-Lattice-Boltzmann-Methode. Im Folgenden wird auf Erweiterungen eingegangen, die im Rahmen dieser Arbeit entwickelt wurden, um die Genauigkeit und Stabilität der Methode zur Simulation von Wirbelströmungen zu verbessern.

2.5. Entwicklung des Strömungslösers

Im Rahmen dieser Arbeit wurde ein Strömungslöser entwickelt, um verschiedene Erweiterungen der Lattice-Boltzmann-Methode zu implementieren und zu testen. Zur effizienten Simulation von Wirbelströmungen wurde zum einen ein neuartiger Kollisionsschritt entwickelt. Zum anderen wurden Verallgemeinerungen der Lattice-Boltzmann-Methode für irreguläre Rechengitter untersucht und weiterentwickelt. Die Erweiterung auf irreguläre Gitter geschieht dabei durch eine Veränderung des Strömungsschritts.

Während sich der neu entwickelte Kollisionsschritt einfach in bestehende Lattice-Boltzmann-Codes integrieren lässt, erfordert die Anpassung des Strömungsschritts eine grundlegende Überarbeitung der Algorithmen und Datenstrukturen. Dazu zählen neben der Definition des Rechengitters unter anderem dünnbesetzte Matrizen, Gleichungssystemlöser, finite Elemente und Zeitintegratoren. Da reine Lattice-Boltzmann-Löser wie Palabos [2], OpenLB [102], waLBerla [62] oder Musubi [88] diese Funktionalitäten nicht unterstützen, wurde für die Methoden und Simulationen der vorliegenden Arbeit ein Code entwickelt, der besonders auf die Implementierung sogenannter Off-Lattice-Boltzmann-Verfahren ausgerichtet ist.

Dieser Code, NATrium, war zunächst zur Implementierung von Numerik und Algorithmen für Tribologie unter Verwendung von Mehrkernprozessoren gedacht, hat sich aber im Laufe dieser Arbeit zu einem allgemeiner ausgerichteten Off-Lattice-Boltzmann-Code entwickelt. Die wesentlichen Aspekte von NATrium werden in [126] publiziert.

Der Fokus dieser Arbeit liegt auf der methodischen Weiterentwicklung der Lattice-Boltzmann-Methode und nicht auf deren optimierter Implementierung. Um in diesem Rahmen den Test möglichst vieler verschiedener Konzepte zu vereinfachen, ist der Code modular aufgebaut, sodass verschiedene Kollisions- und Strömungsoperatoren, Zeitintegratoren, Ansatzfunktionen etc. als austauschbare Komponenten der Methode betrachtet werden. Diese Module verwenden dabei zum großen Teil Datenstrukturen, die durch externe Bibliotheken bereitgestellt werden. Diese Bibliotheken unterstützen zum einen eine dimensionsunabhängige Programmierung und zum anderen eine effiziente Parallelisierung. Die technische Umsetzung von NATrium zielt auf eine kontinuierliche Erweiterung und Weiterentwicklung ab, was durch verschiedene Standardpraktiken des Software Engineering unterstützt wird. Diese Aspekte

des Codes werden in den folgenden Unterabschnitten vertieft.

2.5.1. Externe Bibliotheken

NATrium baut zu einem großen Teil auf der Finite-Elemente-Bibliothek *deal.II* [7–9] auf. Diese implementiert grundlegende Konzepte von Finite-Elemente und Discontinuous-Galerkin-Methoden, wie zum Beispiel irreguläre Rechengitter, Ansatzfunktionen, Numerische Quadraturformeln, dünnbesetzte Matrizen, lineare Löser und Zeitintegratoren. Die Funktionalitäten von *deal.II* werden ergänzt durch Schnittstellen zu anderen Bibliotheken, wie zum Beispiel *p4est* [25], *Trilinos* [100] und *boost* [192]. Die Bibliothek *p4est* implementiert sogenannte *Octree*-Gitter, also hierarchische Gitterstrukturen, deren Baumstruktur eine effiziente Parallelisierung ermöglicht. Mithilfe der *deal.II*-Schnittstelle werden die *Octrees* auf irreguläre Gitter abgebildet. Um auf diesen Gittern verteilte Vektoren und Matrizen zu definieren, verwendet NATrium das *Epetra*-Paket von *Trilinos*. Dieses implementiert neben dünnbesetzten Matrizen auch verschiedene lineare Löser und Prädiktionierer. Aus den *boost*-Bibliotheken wird neben einigen Werkzeugen zur Interaktion mit dem Betriebssystem auch das Unit-Test-Framework verwendet.

2.5.2. Dimensionsunabhängige Programmierung

Ein wesentlicher Aspekt von *deal.II* ist die dimensionsunabhängige Programmierung. Diese ist auch in NATrium umgesetzt, sodass die meisten Klassen die Dimension als Template-Parameter (`dim`) enthalten. Auf diese Weise lassen sich weite Teile des Codes universell für zwei- und dreidimensionale Strömungen implementieren. Diese Vorgehensweise spart nicht nur viel Programmiercode ein, sondern trägt auch zur Erweiterbarkeit und zur Vermeidung von Programmierfehlern bei.

2.5.3. Modularer Aufbau

NATrium ist in *C++* umgesetzt, sodass sich die modulare Codestruktur in der Klassenhierarchie widerspiegelt. Die wichtigsten Klassen dieser Struktur und deren Zusammenspiel sind in Abbildung 2.1 dargestellt. Grundlegend ist dabei die Aufteilung der Software in Problembeschreibung und Löser. Durch diese Zweiteilung lassen sich neu implementierte Methoden sofort für alle bereits definierten Strömungen testen. Die Problembeschreibung (`ProblemDescription<dim>`) bildet eine abstrakte Schnittstelle zu den Klassen, die die Strömungen definieren. Dort werden alle problemabhängigen Größen definiert, also das Gitter, sämtliche Randbedingungen (`BoundaryCollection<dim>`) und Anfangsbedingungen sowie physikalische Parameter wie die Viskosität, die charakteristische Länge und die charakteristische Geschwindigkeit. Strömungen, die eine analytische Lösung besitzen, können als Subklasse der

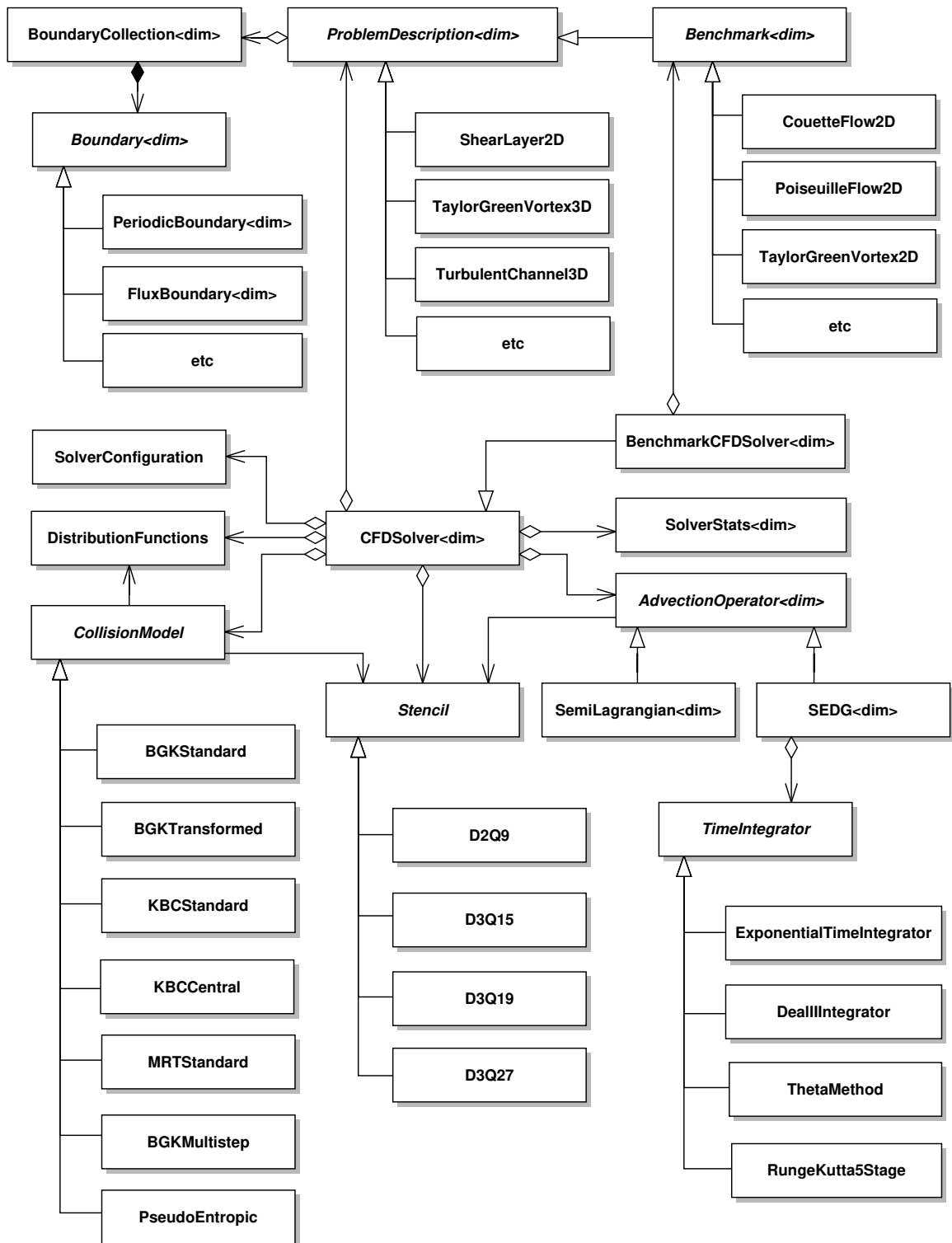


Abbildung 2.1: UML-Klassendiagramm der wichtigsten Klassen und Beziehungen im Simulationscode NATriuM. Namen von abstrakten Klassen sind kursiv gedruckt. Die Endung <dim> bedeutet, dass es sich bei der Klasse um ein dimensionsabhängiges Klassentemplate handelt.

Klasse `Benchmark<dim>` definiert werden.

Auf der Seite des Strömungslösers koordiniert die Klasse `CFDSolver<dim>` die Simulation (oder für Strömungen mit analytischer Lösung `BenchmarkCFDSolver<dim>`). Dort werden die Verteilungsfunktionen initialisiert, die Kollisions- und Advektionsroutinen aufgerufen und Output-Dateien geschrieben. Die Auswertung der Lösung übernimmt dabei die Klasse `SolverStats<dim>`. Der Strömungslöser wird konfiguriert über eine Konfigurationsklasse `SolverConfiguration`, die über Setter-Funktionen, über eine Konfigurationsdatei oder über Argumente der Kommandozeile verändert werden kann. Die Verteilungsfunktionen sind in der Klasse `DistributionFunctions` definiert. Verschiedene Diskretisierungen des Geschwindigkeitsraums können über die Klassen `D2Q9`, `D3Q15`, `D3Q19` und `D3Q27` ausgewählt werden, die von der abstrakten Klasse `Stencil` abgeleitet sind.

Die zentralen Operationen der Lattice-Boltzmann-Methode sind der Kollisionsschritt und der Advektionsschritt. Die abstrakte Klasse `CollisionModel` bildet die Schnittstelle zu verschiedenen Kollisionsmodellen. `NATrium` unterstützt neben dem BGK-Modell (`BGKStandard`) auch eine Version des BGK-Modells mit transformierten Verteilungsfunktionen (`BGKTransformed`, [201]). Darüber hinaus sind stabilere Kollisionsmodelle implementiert, wie zum Beispiel verschiedene Versionen des Modells von Karlin, Bösch und Chikatamarla (`KBCStandard` und `KBCCentral`, [117]), ein Multiple-Relaxation-Time-Modell (`MRTStandard`, [47, 135]) sowie ein Mehrschrittverfahren (`BGKMultistep`) und ein stabilisiertes Modell (`PseudoEntropic`, [127]), die beide in dieser Arbeit entwickelt wurden. Diese Kollisionsmodelle werden in Kapitel 3 und Anhang A näher beschrieben.

Auf Seiten der Advektionslöser sind zwei Verfahren implementiert: ein Spektral-Element-Discontinuous-Galerkin-Verfahren (`SEDG<dim>`) und der in dieser Arbeit entwickelte Semi-Lagrangesche Strömungsschritt (`SemiLagrangian<dim>`) [125], der als Spezialfall auch den Standard-Strömungsschritt der LBM enthält. Diese beiden Verfahren sind als Spezialisierungen der abstrakten Klasse `AdvectionOperator<dim>` umgesetzt und werden in den Kapiteln 4 und 5 erklärt und analysiert. Der Discontinuous-Galerkin-Löser benötigt einen Zeitintegrator. Dazu stehen verschiedene Verfahren zur Auswahl, welche als Spezialisierungen der Klasse `TimeIntegrator` implementiert sind.

Der modulare Aufbau von `NATrium` erlaubt die Auswahl der einzelnen Komponenten über die Konfiguration des Strömungslösers. Auf diese Weise können viele verschiedene Varianten der LBM getestet und effektiv miteinander kombiniert werden. Dabei wird sichergestellt, dass die rechenintensiven Teile der Simulation, nämlich Kollision und Advektion, hardwarenah und effizient umgesetzt sind.

Da der aufwendige Teil des Advektionsschritts in großen dünnbesetzten Matrix-Vektor-Multiplikationen besteht, kann dessen Effizienz durch Verwendung optimierter Bibliotheken sichergestellt werden. Die Kollisionsoperatoren sind dagegen so umgesetzt, dass alle rechnerisch aufwendigen Schritte soweit wie möglich außerhalb der

lokalen Iterationen implementiert sind. Dies beinhaltet die Allokierung von Speicherplatz, den Aufruf virtueller Funktionen und das Dereferenzieren von Zeigern. Innerhalb der Schleife des Kollisionsschritts werden nur *Inline*-Funktionen aufgerufen. Um auch auf dieser Ebene eine Modularität des Codes zu gewährleisten, nutzt NATrium Template-Funktionen. Im Gegensatz zu virtuellen Funktionen werden diese vom Präprozessor instanziiert [206], sodass schon zum Zeitpunkt der Kompilierung bekannt ist, welche Instanz der Templatefunktion aufgerufen wird. Moderne Compiler und Linker sind somit in der Lage, diese Funktionen *inline* zu schreiben [206], sodass trotz der Modularität kein Mehraufwand entsteht.

Außerdem wird auf jedes Vektorelement im Kollisionsschritt nur einmal lesend und einmal schreibend zugegriffen, um die Speicherbandbreite möglichst effizient auszunutzen. Während der Implementierung und Optimierung des Kollisionsschritts wurde mit *KCachegrind* [226] sichergestellt, dass der Code den größten Teil seiner Laufzeit in elementaren Rechenoperationen und unverzichtbaren Speicherzugriffen verbringt. Die komplexe Struktur von NATrium unterstützt somit die Erweiterbarkeit und Lesbarkeit, ohne auf der anderen Seite die Performanz zu beschädigen.

2.5.4. Parallelisierung

Zurzeit beschränkt sich die Parallelisierung in NATrium auf das *Message Passing Interface* (MPI), da die Kombination der eingebundenen Bibliotheken zum aktuellen Zeitpunkt keine effiziente Einbindung von geteiltem Speicher oder Grafikprozessoren erlaubt. In nächster Zeit ist allerdings eine Einbindung der Tpetra-Bibliothek von Trilinos in deal.II geplant. Durch diese Weiterentwicklung kann auch NATrium in Zukunft auf GPUs erweitert werden.

Viele Elemente der Parallelisierung sind in p4est, deal.II und Trilinos verankert. Dabei verteilt p4est das Rechengitter durch eine Gebietszerlegung auf mehrere Speicherbereiche. Die mit den Gitterpunkten assoziierten Freiheitsgrade (in diesem Fall die Verteilungsfunktionen) werden dann durch deal.II-Routinen auf diese Speicherbereiche verteilt und die Indizes der jeweiligen Freiheitsgrade werden in sogenannten *IndexSets* abgelegt.

Der Kollisionsschritt in der LBM greift nur auf lokale Verteilungsfunktionen zu und kann daher trivial parallelisiert werden, indem nur über die lokalen Indizes iteriert wird. Die in NATrium implementierten Strömungsoperatoren werden hingegen durch dünnbesetzte Matrizen realisiert, um eine möglichst allgemeine Formulierung des Strömungsschritts zu unterstützen. Die dazu benötigten parallelen Datenstrukturen sowie die MPI-Parallelisierung der Matrix-Vektor-Multiplikationen werden durch Trilinos bereitgestellt. Zur Assemblierung der Strömungsoperatoren muss auf die lokalen Gitterzellen und auf deren unmittelbare Nachbarzellen zugegriffen werden. Die Nachbarzellen können dabei auch in anderen Speicherbereichen abgelegt sein und werden durch die Definition einer *Halo*-Schicht in den Speicherbereichen mehrerer

MPI-Prozesse verfügbar gemacht [7]. Die Implementierung dieser Konzepte und das effiziente Zusammenspiel von p4est, deal.II und Trilinos sind in [7] dokumentiert. Dort wird durch die Kombination dieser drei Bibliotheken auf bis zu 8 000 Prozessoren eine gute parallele Effizienz erreicht.

Um die Parallelisierung von NATrium zu testen, wurden in dieser Arbeit Simulationen auf bis zu 64 Knoten des HorUS-Rechners der Universität Siegen durchgeführt. Dieser Test ist auch in [126] dokumentiert. Jeder Knoten war ausgestattet mit zwei Intel Xeon X5650 Sechs-Kern-Prozessoren und 48 GB RAM. Jedem Prozessor wurde genau ein MPI-Prozess zugewiesen.

Als Testproblem wurde eine dreidimensionale Couette-Strömung auf dem Strömungsgebiet $[0, L] \times [0, 1] \times [0, 1]$ gerechnet. Durch Variationen der Gebietslänge L und des Verfeinerungs-Levels ℓ wurden unterschiedlich große Probleme erzeugt. In x - und y -Richtung wurden periodische und in z -Richtung Dirichlet-Randbedingungen gesetzt. Der Kollisionsschritt wurde mit dem BGK-Kollisionsmodell und der Strömungsschritt mit einem Discontinuous-Galerkin-Verfahren der Ordnung $N = 1$ durchgeführt. Die Anzahl der Gitterzellen ergibt sich daraus als $2^{3\ell} \cdot L$, wobei jede Zelle $(N + 1)^3$ Gitterpunkte besitzt.

Es wurden zwei Testreihen durchgeführt. Zunächst wurde die Problemgröße pro MPI-Prozess durch $\ell = 5$ und $L = 4 \cdot \#\text{Knoten}$ konstant gehalten (sogenannte *schwache* Skalierung). In der zweiten Testreihe wurde die Gesamtgröße des Problems durch $\ell = 5$ und $L = 32$ konstant gehalten (*starke* Skalierung).

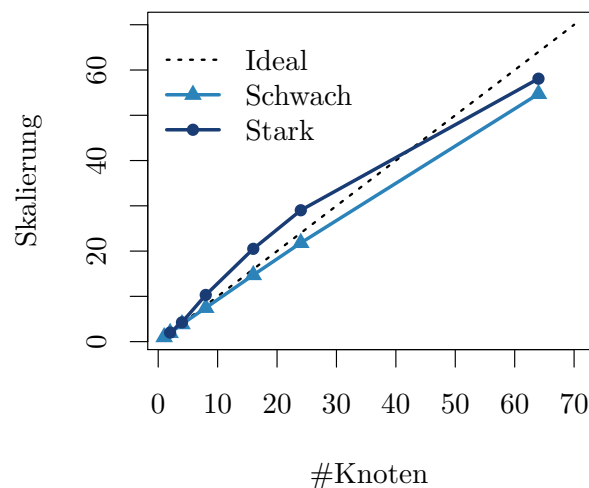


Abbildung 2.2: Schwache und starke Skalierung des NATrium-Codes auf bis zu 64 Rechenknoten. Bei der schwachen Skalierung hatte das Gitter 131 072 Zellen pro Knoten und bei der starken Skalierung insgesamt 1 048 576 Zellen.

Abbildung 2.2 zeigt, dass in beiden Tests ein guter Speedup erreicht werden konnte. Bei der schwachen Skalierung lag die parallele Effizienz auf 64 Knoten bei 84% und

bei der starken Skalierung bei 90%. Es fällt auf, dass im Bereich von zwei bis 24 Knoten ein Speedup von über 100% erreicht wurde. Dies ergibt sich dadurch, dass bei der Verwendung mehrerer Knoten ein größerer Teil der Daten im Cache der Prozessoren Platz findet. Die dadurch erzielte Beschleunigung der Zugriffe auf Matrix- und Vektorelemente überwiegt in diesem Bereich den Kommunikationsoverhead. Dies zeigt, dass die Speicherzugriffe bei größeren Problemen den Rechenaufwand dominieren, insbesondere durch den großen Speicherbedarf des Advektionsoperators. Aus diesem Grund sollte für die Weiterentwicklung von NATrium eine matrixfreie Implementierung des Strömungsschritts in Betracht gezogen werden. Insgesamt ist die Skalierung des Codes aber für die im Rahmen dieser Arbeit zur Verfügung stehenden Hardwareressourcen mehr als ausreichend.

2.5.5. Sicherung der Codequalität

Um die kontinuierliche Weiterentwicklung von NATrium auch über diese Arbeit hinaus zu unterstützen, wurde die Entwicklung von Anfang an in einem *git*-Repository archiviert. Somit stehen alle alten Versionen des Codes nach wie vor zur Verfügung. Bei Simulationen mit NATrium wird die Versionsnummer in der Regel sowohl auf der Kommandozeile als auch in der Log-Datei ausgegeben, was die Reproduzierbarkeit der Ergebnisse gewährleistet. Die *git*-Versionsverwaltung unterstützt zudem ein kollaboratives Entwickeln des Codes und dokumentiert kleinschrittig den Entwicklungsprozess.

Zusätzlich wurde im Code eine *Doxygen*-Dokumentation angelegt. Doxygen generiert aus den im Code speziell formatierten Kommentaren automatisch pdf- und html-Dateien, die die Funktionalitäten aller Klassen und Funktionen beschreiben.

NATrium umfasst neben dem Code und der Dokumentation auch eine Testumgebung, die die Funktionalität aller Module in jedem Entwicklungsschritt sicherstellt. Diese besteht zum einen aus einem *Unit-Test-Framework*, das in *boost* umgesetzt ist und für jede Klasse einen Satz von Testfunktionen definiert. In diesem Framework wird jede Klasse weitestgehend unabhängig vom Rest des Codes getestet. Zum anderen besteht die Test-Umgebung aus einem Satz von Integrationstests. Diese führen kleine Simulationen durch und testen damit das Zusammenspiel der einzelnen Klassen und andere wichtige Aspekte, wie zum Beispiel die Konvergenz des Codes. Ein Durchlauf aller Tests dauert im Ganzen wenige Minuten, was eine regelmäßige Überprüfung aller Komponenten ermöglicht.

Abgerundet werden diese Features durch dokumentierte Beispielsimulationen sowie eine kurze, allgemeine Einführung in die Installation und Benutzung des Codes. Insgesamt ist damit eine kontinuierliche Weiterentwicklung des Codes vorbereitet. NATrium soll noch im laufenden Jahr online gestellt werden und ist damit voraussichtlich der erste offene, allgemein angelegte Lattice-Boltzmann-Code, der Simulationen auf unstrukturierten Gittern unterstützt.

2.6. Erste Testfälle

Zum Abschluss dieses Kapitels werden Simulationsergebnisse für einen zweidimensionalen Taylor-Green-Wirbel und eine zweidimensionale Scherschicht gezeigt. Diese verbinden die verschiedenen Abschnitte dieses Kapitels, indem zum einen die Fehlerordnung der LBM demonstriert und damit zum anderen der Code validiert wird. Die Simulation der Scherschicht mit der Standard-LBM zeigt die Schwachpunkte der Standard-LBM und dient als Motivation für die in dieser Arbeit entwickelten Erweiterungen.

2.6.1. Konvergenztest

Der zweidimensionale Taylor-Green-Wirbel [212] wurde auf dem Strömungsgebiet $[0, 2\pi] \times [0, 2\pi]$ mit periodischen Randbedingungen in beiden Richtungen simuliert. Der Taylor-Green-Wirbel eignet sich zur Validierung, da zum einen keine Randbedingungen berücksichtigt werden müssen und zum anderen in diesem Spezialfall eine analytische Lösung der inkompressiblen Navier-Stokes-Gleichungen bekannt ist:

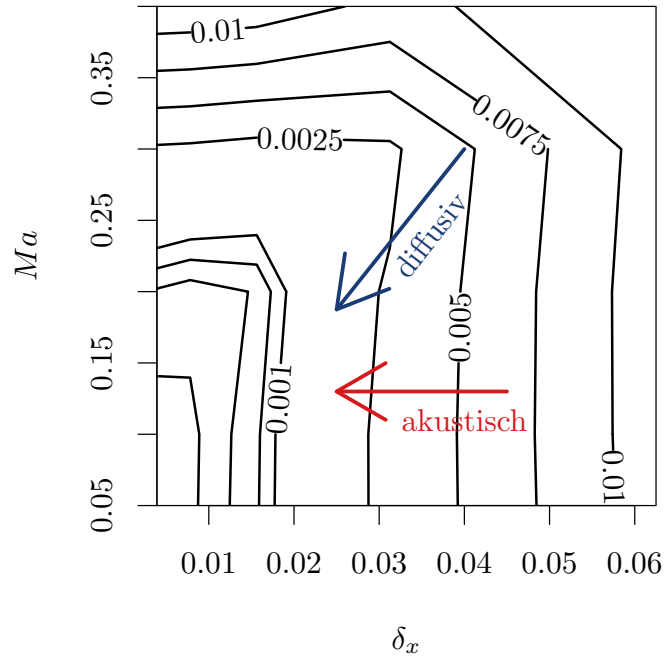
$$\begin{aligned} u_1^{\text{ref}}(x, y, t) &= \sin(x) \cos(y) \exp(-2\nu t), \\ u_2^{\text{ref}}(x, y, t) &= -\cos(x) \sin(y) \exp(-2\nu t), \\ p^{\text{ref}}(x, y, t) &= \frac{1}{4} (\cos(2x) + \cos(2y)) \exp(-4\nu t). \end{aligned}$$

Die Reynoldszahl für den Konvergenztest war $Re = \frac{u_0 l_0}{\nu} = 10$ mit $u_0 = 1$ und $l_0 = 2\pi$. Alle Verteilungsfunktionen wurden mit Gleichgewichtsverteilungen

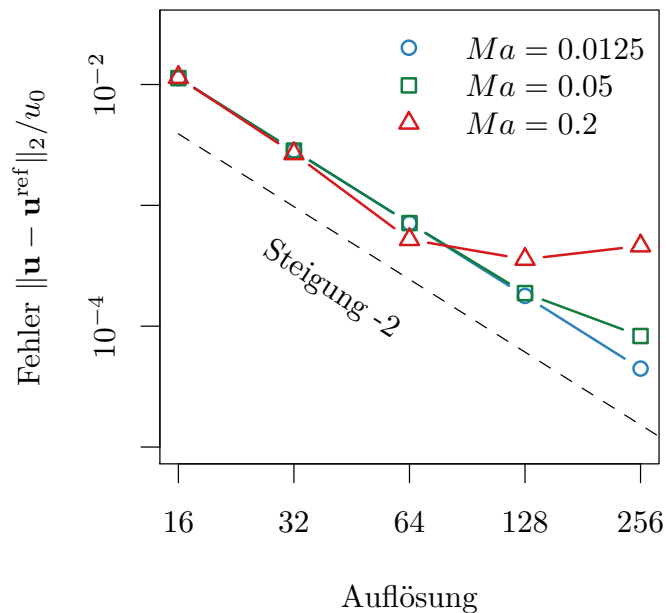
$$f^{\text{eq}} \left(\mathbf{u} = \mathbf{u}^{\text{ref}}|_{t=0}, \rho = 1 + \frac{p^{\text{ref}}|_{t=0}}{c_s^2} \right)$$

initialisiert und dann mit der Standard-LBM mit BGK-Kollisionsmodell simuliert. Die numerischen Fehler wurden zum Zeitpunkt t_{max} mit $0.1 = \exp(-2\nu t_{\text{max}})$ berechnet, also nachdem die Wirbel in der analytischen Lösung um eine Größenordnung abgeklungen waren. Zur Fehlerberechnung wurde eine Gauß-Quadraturformel zweiten Grades verwendet. Die Machzahl Ma und die Gitterauflösung wurden für verschiedene Simulationsläufe variiert. Dabei wurden zunächst die Machzahlen $Ma = 0.4, 0.3, 0.2, 0.1, 0.05$ und Gitterauflösungen von 16×16 bis 256×256 Punkten getestet.

Abbildung 2.3 zeigt die numerischen Fehler in Abhängigkeit der Gitterauflösung und der Machzahl. Das Konturdiagramm des Fehlers (Abbildung 2.3a) verdeutlicht, dass der Fehler für $Ma \rightarrow 0$ und $\delta_x \rightarrow 0$ gegen die inkompressible Lösung konvergiert. Dieses Konvergenzverhalten entspricht der theoretischen Analyse in der diffusiven



(a) Niveaulinien des numerischen Fehlers $\|\mathbf{u} - \mathbf{u}^{\text{ref}}\|_2 / u_0$ in Abhängigkeit der Machzahl Ma und Maschenweite des Gitters δ_x . Die drei niedrigsten Niveaulinien liegen bei Fehlern von 0.00075, 0.0005 und 0.00025. Die beiden eingezeichneten Pfeile zeigen in die Richtung der diffusiven Skalierung $\delta_t \sim \delta_x^2$ und der akustischen Skalierung $\delta_t \sim \delta_x$, die in der theoretischen Analyse der LBM verwendet werden.



(b) Fehler für verschiedene Machzahlen in Abhängigkeit der Gitterauflösung.

Abbildung 2.3: Numerische Fehler bei der Simulation des zweidimensionalen Taylor-Green-Wirbels mit der Standard-LBM.

Skalierung [113]. Weiter wird deutlich, dass bei einer akustischen Skalierung der Fehler in der Machzahl nicht verschwindet. Der verbleibende Term der Größenordnung $O(Ma^2)$ hat keine physikalische Bedeutung und beschränkt den Anwendungsbereich der Standard-LBM auf Simulationen von schwach kompressiblen Strömungen.

In Abbildung 2.3b ist der Fehler in Abhängigkeit von der Auflösung dargestellt. Bei einer ausreichend kleinen Machzahl konvergierte der Fehler in Abhängigkeit von der Auflösung quadratisch. Die mit NATrium ermittelte Fehlerordnung der Standard-LBM stimmt somit mit der theoretischen Voraussage überein [30, 107, 113].

2.6.2. Problematik der Standard-Lattice-Boltzmann-Methode

Der Taylor-Green-Wirbel ist keine große Herausforderung für Verfahren der Strömungssimulation, da er während der gesamten Simulationszeit nur aus einer Fourier-Mode besteht und keine steilen Gradienten auftreten. Um deutlich zu machen, dass die LBM sich in ihrer Standard-Form nicht besonders gut zur Simulation von Wirbelströmungen eignet, werden daher noch einige Ergebnisse für einen schwierigen Fall gezeigt.

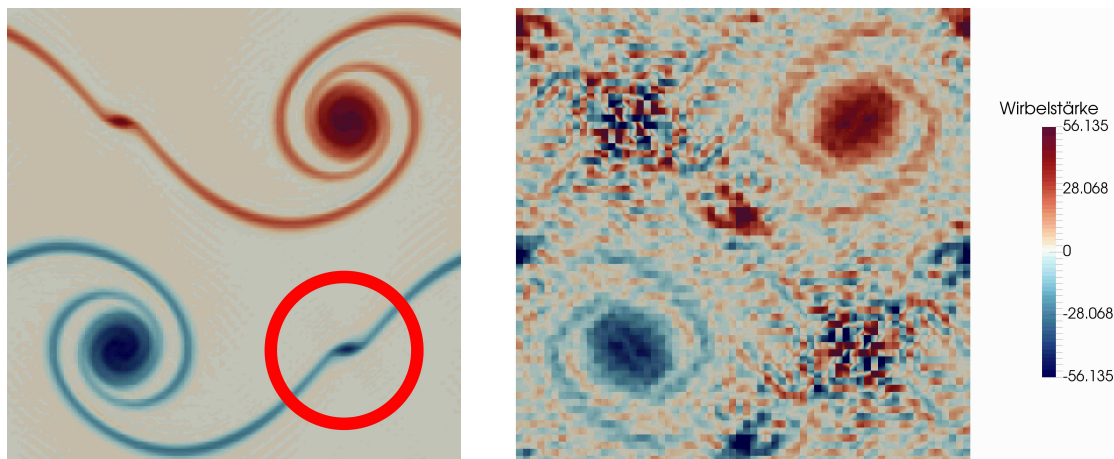
Dabei handelt es sich um die Simulation einer Kelvin-Helmholtz-Instabilität, bei der kleine Störungen in einer Scherschicht zu größeren Wirbelstrukturen anwachsen. Dieser Testfall einer zweidimensionalen periodischen Scherschicht wurde in vielen Arbeiten simuliert (zum Beispiel [23, 42, 149, 162]), da er von einer Simulationsmethode verlangt, gleichzeitig Wirbelströmungen und scharfe Grenzbereiche wiederzugeben.

Nach [162] wurde die Strömung im Strömungsgebiet $[0, 1]^2$ initialisiert durch

$$\begin{aligned} u_1(x, y) &= \begin{cases} \tanh[80 \cdot (y - 0.25)], & y \leq 0.5 \\ \tanh[80 \cdot (0.75 - y)], & \text{sonst;} \end{cases} \\ u_2(x, y) &= 0.05 \cdot \sin[2\pi(x + 0.25)]; \\ p(x, y) &= 0. \end{aligned}$$

Mit $u_0 := 1$ und $l_0 := 1$ ergibt sich eine Reynoldszahl von $Re = 1/\nu$, die wie in [162] als $Re = 10\,000$ gewählt wurde. Die Machzahl wurde auf $Ma = \sqrt{3}/25$ gesetzt. Die Strömung wurde für zwei verschiedene Auflösungen, 128×128 und 64×64 Gitterpunkte, bis zum Zeitpunkt $t_{\max} := 1$ simuliert. Dies entspricht einer *eddy turnover time*.

Durch die kleine initiale Störung in der vertikalen Geschwindigkeitskomponente u_2 bildeten sich wie in der Referenzlösung Wirbel im Bereich der stärksten Scherung. Abbildung 2.4 zeigt die Wirbelstärke im Strömungsgebiet zum Zeitpunkt $t = 0.75$. Bei der höheren Auflösung (Abbildung 2.4a) bildeten sich Wirbel an vier Stellen. Simulationen mit einer feineren Auflösung von etwa 512×512 Gitterpunkten zeigen, dass die kleineren Wirbel unphysikalisch sind und sich nur durch das numerische Verfahren ergeben [18].



(a) Gitterauflösung: 128×128 Punkte. (b) Gitterauflösung: 64×64 Punkte.

Abbildung 2.4: Wirbelstärke $\omega = \partial u_2 / \partial x - \partial u_1 / \partial y$ zum Zeitpunkt $t = 0.75$ in der Simulation einer zweidimensionalen Scherschicht mit der Standard-LBM. Der rote Kreis markiert einen der beiden unphysikalischen sekundären Wirbel.

Bei einer noch niedrigeren Auflösung von 64×64 Punkten verteilten sich die unphysikalischen Störungen in kurzer Zeit über das gesamte Strömungsgebiet (siehe Abbildung 2.4b). Diese Störungen mündeten in einer instabilen Simulation, da die LBM in der in diesem Kapitel beschriebenen Standardformulierung kaum numerische Dissipation erzeugt. Auf der einen Seite bietet die fehlende numerische Dissipation der LBM einen Vorteil, der in anderen Methoden der numerischen Strömungsmechanik nur durch einen hohen rechnerischen Aufwand erzielt werden kann (kleine Zeitschritte und Verfahren hoher Ordnung). Die Simulation der Scherschicht zeigt aber auch den Nachteil der fehlenden numerischen Dissipation: Hochfrequente Moden werden kaum gefiltert, was leicht zu einem Anwachsen unphysikalischer Effekte führen kann [76, 118].

Die kleine numerische Dissipation ergibt sich durch den Strömungsschritt, der auf regulären Gittern *exakt* (und äußerst effizient) durchgeführt werden kann. Dieser Vorteil der regulären Gitter erzwingt aber auf der anderen Seite Einschränkungen bezüglich der geometrischen Flexibilität der Methode. Während in direkten Navier-Stokes-Lösern in der Regel mit geometrieangepassten Gittern gearbeitet wird [6], ist dies in der LBM nicht trivial umsetzbar.

Ein weiterer Nachteil gegenüber klassischen Lösern ist der höhere Speicheraufwand der LBM, der durch das Abspeichern aller Verteilungsfunktionen zustande kommt. Durch die damit verbundenen Speicherzugriffe wird die Performanz der Standard-LBM durch die Speicherbandbreite und nicht durch die Anzahl an Rechenoperationen limitiert [73].

Eine weitere Problematik der LBM ist die Simulation stationärer Strömungen. Da

die LBM *per definitionem* eine transiente Methode ist, ist es für stationäre Probleme im Allgemeinen günstiger, klassische Strömungslöser zu verwenden. In diesen können die Zeitableitungen von vornherein aus den Gleichungen eliminiert werden [6].

Die größte Herausforderung bei der Weiterentwicklung der LBM ist es, diese Schwächen zu kompensieren, ohne dabei die Effizienz der Methode wesentlich zu beeinträchtigen. Die vorliegende Arbeit zielt darauf ab, die LBM zum einen im Hinblick auf ihre Stabilität zu verbessern und zum anderen eine effizientere LBM-Simulation auf irregulären Gittern zu ermöglichen. Im Folgenden wird dazu zunächst ein neuartiger Kollisionsschritt entwickelt, der die praktische Anwendbarkeit der LBM auf Probleme wie die gerade eingeführte Scherschicht verbessert und auch auf groben Gittern eine stabile Simulation von Wirbelströmungen ermöglicht.

3. Kollisionsmodellierung

In ihrer Standardformulierung ist die Lattice-Boltzmann-Methode nur bedingt zur Simulation von Wirbelströmungen und insbesondere turbulenten Strömungen geeignet. Vor allem bei großen Reynoldszahlen werden Simulationen schnell instabil. Dies hängt zum einen damit zusammen, dass im Strömungsschritt keine numerische Dissipation erzeugt wird. Zum anderen entspricht die Kollision im BGK-Modell einer kaum gedämpften Überrelaxation, falls die Viskosität sehr klein ist. Diese beiden Faktoren begünstigen das Anwachsen von Oszillationen. Turbulente Simulationen mit dem BGK-Modell erfordern daher kleine Zeitschritte, was den numerischen Aufwand in die Höhe treibt – insbesondere weil der Zeitschritt in der Standard-LBM direkt mit der Gitterauflösung gekoppelt ist ($\delta_x = \delta_t \xi_0$).

Um die Stabilität der Lattice-Boltzmann-Methode auch auf groben Gittern zu verbessern, muss also entweder im Strömungsschritt oder im Kollisionsschritt numerische Dissipation erzeugt werden. Erweiterungen des Strömungsschritts werden später betrachtet, siehe Kapitel 4 und 5. In diesem Kapitel liegt das Augenmerk auf Erweiterungen des Kollisionsschritts. Zuerst werden dabei die wichtigsten Kollisionsmodelle aus der Literatur beschrieben (Abschnitt 3.1). Dann wird die im Rahmen dieser Arbeit entwickelte *Pseudo-Entropische Stabilisierung* vorgestellt (Abschnitt 3.2), auf Beispielströmungen angewandt (Abschnitt 3.3) und diskutiert (Abschnitt 3.4).

3.1. Stand der Technik

Die Kollisionsmodelle bilden den Kern der LBM. Sie definieren die lokale Evolution der Verteilungsfunktionen und bestimmen damit die globale Dynamik der makroskopischen Momente. Der Kollisionsoperator kann somit als *kinetischer Generator* („kinetic generator“ [208]) für partielle Differentialgleichungen betrachtet werden. Die LBM kann also nicht nur auf die klassische Hydrodynamik angewandt werden, sondern zum Beispiel auch auf Probleme aus der Quantenmechanik [209] oder sogar der Finanzmathematik [240]. Interessante physikalische Anwendungsbereiche, die über die Navier-Stokes-Gleichungen hinausgehen, sind die Simulation von Nichtgleichge-

wichtsströmungen bei finiten Knudsen-Zahlen [65, 80, 213] oder die Verwendung als implizites Lösungsmittel in molekularen Simulationen [52].

Die Lösung hydrodynamischer Erhaltungsgleichungen ist aber nach wie vor der primäre Anwendungsbereich der LBM. So existieren verschiedene Ansätze zur Simulation von Mehrphasenströmungen [130, 157, 194, 211] oder kompressiblen Strömungen [68, 108, 122]. In der vorliegenden Arbeit liegt der Fokus auf Simulationen von inkompressiblen Strömungen. Die hierzu entwickelten Modelle werden im Folgenden diskutiert. Ein umfassender Überblick über andere Kollisionsmodelle findet sich bei Guo und Shu [78].

3.1.1. Dynamische Subgrid-Modelle

Eine klassische Methode zur Simulation von turbulenten Strömungen auf unteraufgelösten Gittern ist die Verwendung von *Subgrid*-Modellen. Um die Simulationen zu stabilisieren, wird lokal die Viskosität erhöht und somit zusätzliche Dissipation erzeugt. Dieser Ansatz wurde schon früh von klassischen Lösern auf die LBM übertragen [49, 77, 118, 124, 152, 184, 185, 190, 198, 242].

Besonders viel Aufsehen hat in diesem Zusammenhang die sogenannte Entropische LBM [118] erregt [15, 34, 51, 68, 76, 151, 174, 208]. Diese nutzt die mikroskopische Information des diskreten kinetischen Modells in Form einer Entropiefunktion

$$H(f) = \sum_{i=0}^{Q-1} f_{(i)} \log \left(\frac{f_{(i)}}{\omega_{(i)}} \right). \quad (3.1)$$

Das Boltzmannsche H-Theorem besagt, dass die Kollision die Entropie nicht verringert, was aber gerade bei kleinen Viskositäten in der LBM aufgrund der starken Überrelaxation nicht immer gegeben ist. Die Entropische LBM erzwingt diese Eigenschaft. Dazu wird im Kollisionsschritt eine sogenannte *Mirror*-Verteilung definiert, die die gleiche Entropie wie die ursprüngliche Geschwindigkeitsverteilungsfunktion besitzt. Die Relaxation wird dann in Bezug auf die *Mirror*-Verteilung durchgeführt, was ein Wachstum der Entropie erzwingt und bei unteraufgelösten Simulationen zu einer lokalen Erhöhung der Viskosität führt. Zur Berechnung der *Mirror*-Verteilung muss eine nichtlineare Gleichung gelöst werden. Dies erhöht den rechnerischen Aufwand des Kollisionsschritts um einen Faktor zwei bis drei [116]. Dafür eignen sich die Entropische LBM und ihre Erweiterungen gut zur Simulation von Strömungen mit steilen Gradienten, wie zum Beispiel Mehrphasenströmungen und kompressibler Turbulenz [68, 157].

Allerdings handelt es sich bei der Entropiefunktion (3.1) eigentlich um Informationsentropie [228], da die diskreten Verteilungsfunktionen in Folge der Transformation (2.7) als rein numerische Größen betrachtet werden können. Die aus den f_i berechneten „physikalischen Verteilungen“ sind in der Praxis oft negativ und $H(f)$ somit

undefiniert. Somit hat die Entropie in der Entropischen LBM vor allem die Funktion, die transformierten Geschwindigkeitsverteilungen ausreichend zu randomisieren, um die Stabilität der Simulationen zu gewährleisten. Im Gegensatz zu klassischen Subgrid-Modellen wird dabei allerdings direkt die kinetische Information aus der mesoskopischen Modellierung genutzt.

3.1.2. Entropische Filter

Entropiefunktionen spielen auch in anderen Kollisionsmodellen eine Rolle. So wird zum Beispiel bei sogenannten Entropischen Filtern eine quadratische Approximation der Entropie als Lyapunov-Funktion verwendet [24, 76]. Mit dieser Funktion wird an jedem Punkt die Abweichung vom Gleichgewicht ΔS definiert. Mithilfe der benachbarten Gitterpunkte wird der Median der Abweichungen ΔS_{med} bestimmt. Bei großen Abweichungen vom Gleichgewicht wird eine Filterungsoperation

$$f^{\text{filt}} = f^{\text{eq}} + \frac{\Delta S_{\text{med}}}{\Delta S} (f - f^{\text{eq}})$$

durchgeführt. Dieser Schritt bewirkt, dass die gefilterten Verteilungsfunktionen f^{filt} nur noch so weit vom Gleichgewicht abweichen wie der Median der benachbarten Verteilungsfunktionen.

Ein Nachteil der entropischen Filter ist, dass sie die Lokalität des Kollisionsschritts und somit einen Kernaspekt der LBM verletzen.

3.1.3. Multiple-Relaxation-Time-Modelle und Erweiterungen

Eine weitere wichtige Klasse von Kollisionsmodellen bilden die Multiple-Relaxation-Time (MRT)-Modelle [42, 47, 135]. Diese relaxieren die Momente der diskreten Geschwindigkeitsverteilungsfunktion mit verschiedenen Raten. Dazu wird zunächst eine Momentenmatrix $M \in \mathbb{R}^{Q \times Q}$ definiert, welche die Geschwindigkeitsverteilungsfunktion $f \in \mathbb{R}^Q$ auf eine Auswahl ihrer Momente $m \in \mathbb{R}^Q$ abbildet. Der Kollisionsoperator schreibt sich dann als

$$\Omega(f) = M^{-1} S (M f - m^{\text{eq}}),$$

wobei m^{eq} die Momente der Gleichgewichtsverteilung bezeichnet und S eine Diagonalmatrix ist, welche die Relaxationsraten enthält. Das BGK-Modell ergibt sich als Spezialfall für $S = \text{diag}(-1/\tau, \dots, -1/\tau)$. Bei geeigneter Wahl der Relaxationsparameter können MRT-Modelle die Stabilität der Simulationen verbessern [42], welche Relaxationsparameter geeignet sind, ist jedoch *a priori* nicht klar.

Viele MRT-Modelle nutzen wie die Pionier-Publikationen [46, 47, 135] Momente, die bezüglich des kartesischen Skalarprodukts orthogonal sind. Diese Wahl erlaubt eine

Relaxation der höheren Momente, die die konservativen und Scher-Momente lokal nicht beeinflusst. Allerdings sind die Momente global auf der Ebene der Differentialgleichungen miteinander gekoppelt [74, 135].

Als Konsequenz dieser Überlegung wurden die orthogonalen Momente in manchen Modellen durch andere „Observablen“ ausgetauscht. So definiert Dellar [42] die Momente als Koeffizienten einer Hermite-Entwicklung der Verteilungsfunktionen. Durch die Wahl der Observablen können bestimmte Eigenschaften der LBM verbessert werden. Beispielsweise ist die LBM infolge der Diskretisierung des Geschwindigkeitsraums (im Gegensatz zu den physikalischen Erhaltungsgleichungen) nicht Galilei-invariant. Dies manifestiert sich zum Beispiel im geschwindigkeitsabhängigen Fehlerterm (B.18) aus der Chapman-Enskog-Analyse. In der *Cascaded LBM* [70] werden die Momente daher in einem Bezugsrahmen berechnet, der sich mit der Fluidgeschwindigkeit bewegt: $\sum_i f_{(i)}(e_{(i)j} - u_j)(e_{(i)k} - u_k) \dots$. Ein Derivat dieses Modells [72] wird auch im kommerziellen Löser *XFlow* verwendet [74]. Durch die Verwendung dieser zentralen Momente kann der numerische Fehler in der Galilei-Invarianz reduziert werden [71]. Dieses Modell wurde in der Folge zur *Cumulant LBM* weiterentwickelt. Diese verwendet statt der linearen Momente (nichtlineare) Kumulanten der Geschwindigkeitsverteilungsfunktion [74], die zur Entkopplung der Observablen beitragen. Dadurch können zum einen Fehler in der Galilei-Invarianz und in den Hyper-Viskositäten gegenüber MRT-Modellen reduziert werden. (Hyper-Viskositäten sind Transportkoeffizienten höherer Ordnung, die die Dissipation der kleinen Skalen bestimmen.) Zum anderen sind die Relaxationsparameter der höheren Kumulanten frei in dem Sinne, dass sie nicht zur Elimination von Hyper-Viskositäten oder Fehlern in der Galilei-Invarianz optimiert werden müssen.

Besonders stabile MRT-Modelle können durch eine entropiegesteuerte Wahl der Relaxationsparameter erzeugt werden. So wurden in [18, 117] *Two-Relaxation-Time*-Modelle entwickelt, die die physikalischen und nicht-physikalischen Momente mit verschiedenen Parametern relaxieren. Während die physikalischen Momente gemäß der vorgegebenen Viskosität relaxiert werden, wird der Relaxationsparameter für die nicht-physikalischen Momente über eine Entropie-Maximierung an die lokale Strömung angepasst.

3.1.4. Regularisiertes Modell

Eine weitere Methode zur Reduktion der hochfrequenten Oszillationen ist die Regularisierte LBM [139]. Diese ersetzt die Verteilungsfunktionen vor dem Kollisionsschritt durch regularisierte Verteilungsfunktionen. Dies geschieht durch die Approximation

$$f^{\text{reg}} = f^{\text{eq}} + f^{(1)}$$

mit

$$f_i^{(1)} = \frac{\omega^{(i)}}{2c_s^4} (e_{(i)j}e_{(i)l} - c_s^2\delta_{jl}) \left(\sum_{k=0}^{Q-1} (f_{(k)} - f_{(k)}^{\text{eq}}) e_{(k)j}e_{(k)l} \right)$$

Die große Klammer beschreibt den Nichtgleichgewichtsanteil der zweiten Momente. Somit verschwinden die Beiträge der höheren Momente zur regularisierten Geschwindigkeitsverteilungsfunktion, was zu einer Verbesserung der Stabilität und Genauigkeit führt [139]. Das regularisierte Modell kann auch als MRT-Modell mit voller Relaxation der höheren Momente aufgefasst werden [139]. Die Momente sind dabei orthogonal bezüglich eines gewichteten Skalarprodukts, siehe unten.

3.2. Pseudo-Entropische Stabilisierung

Im Rahmen dieser Arbeit wurde eine neue Klasse von Kollisionsmodellen entwickelt, die Ideen der Regularisierten LBM und der entropischen MRT-Modelle kombiniert. Diese *Pseudo-Entropische Stabilisierung* wird in [127] publiziert. Im Folgenden wird das Verfahren zunächst hergeleitet und formuliert (Unterabschnitt 3.2.1). Darauf folgt eine theoretische Analyse (Unterabschnitt 3.2.2).

3.2.1. Herleitung und Formulierung

Die Herleitung der PES gliedert sich in eine Beschreibung der Grundidee (Paragraph 3.2.1.1), eine Darlegung der zugrundeliegenden Momentendarstellung und Pseudo-Entropie-Funktion (Paragraph 3.2.1.2) sowie die analytische Maximierung dieser Pseudo-Entropie (Paragraph 3.2.1.3). Im Anschluss daran wird der Algorithmus zunächst allgemein formuliert (Paragraph 3.2.1.4) und dann für zwei- und dreidimensionale Strömungen spezialisiert (Paragraphen 3.2.1.5 und 3.2.1.6).

3.2.1.1. Grundidee

Während die Regularisierte LBM *a priori* den Einfluss der nicht-hydrodynamischen Momente minimiert, werden in den hier vorgestellten Modellen die nicht-hydrodynamischen Momente genutzt, um die Stabilität des Verfahrens zu verbessern. Diese Weiterentwicklung zielt darauf ab, im Stil einer LES turbulente Strömungen auf groben Gittern zu simulieren.

Dazu werden die hydrodynamischen Momente $m_{\mathcal{P}}$ wie im BGK-Modell gemäß der vorgegebenen Viskosität relaxiert:

$$m_{\mathcal{P}}^{\text{pc}} = m_{\mathcal{P}} - \frac{1}{\tau} (m_{\mathcal{P}} - m_{\mathcal{P}}^{\text{eq}}).$$

Dabei bezeichnet $m_{\mathcal{P}}^{\text{pc}}$ die hydrodynamischen Postkollisionsmomente und $m_{\mathcal{P}}^{\text{eq}}$ die hydrodynamischen Gleichgewichtsmomente.

Im Stil des des entropischen MRT-Modells wird dann eine Entropiefunktion \tilde{H} maximiert, um die optimalen nicht-hydrodynamischen Momente $m_{\mathcal{Q}}^{\text{pc}}$ zu bestimmen:

$$m_{\mathcal{Q}}^{\text{pc}} := \arg \max_{m_{\mathcal{Q}}} \tilde{H}(f) \quad (3.2)$$

Mit diesen optimierten nicht-hydrodynamischen Momenten wird dann die Postkollisionsverteilung gebildet.

Dieses Vorgehen lässt sich sowohl physikalisch als auch numerisch motivieren. Die nicht-hydrodynamischen Momente tauchen nicht in den makroskopischen Erhaltungsgleichungen auf und können daher als freie Parameter des Verfahrens interpretiert werden. Aus physikalischer Sicht werden die Momente so gewählt, dass die makroskopischen Beziehungen wiedergegeben werden, während die nicht-hydrodynamischen Anteile ihre wahrscheinlichsten Werte annehmen. Da in der Thermodynamik wie in der Informationstheorie die Wahrscheinlichkeit mit der Entropie wächst, geschieht dies effektiv durch die Maximierung der Entropie. Aus numerischer Sicht dämpft dieses Vorgehen die Oszillationen in den nicht-hydrodynamischen Anteilen. Dabei soll die Stabilität verbessert werden ohne direkten Einfluss auf die Viskosität zu nehmen.

3.2.1.2. Momentenmatrix und Pseudo-Entropiefunktion

Dieser Ansatz wird im Folgenden für allgemeine Lattice-Boltzmann-Modelle formuliert. Dazu sei $M_{\mathcal{P}}$ die Momentenmatrix der hydrodynamischen Momente, also

$$m_{\mathcal{P}} = M_{\mathcal{P}} f. \quad (3.3)$$

Genauso sei $M_{\mathcal{Q}}$ die Momentenmatrix der hydrodynamischen Momente, so dass gilt

$$M_{\mathcal{P}} M_{\mathcal{Q}}^T = 0 \quad (3.4)$$

und $\mathcal{P} \cup \mathcal{Q} = \{0, 1, \dots, Q-1\}$. Dies bedeutet, dass die hydrodynamischen und nicht-hydrodynamischen Momente paarweise orthogonal sind und somit eine Optimierung von $m_{\mathcal{Q}}$ keinen unmittelbaren Einfluss auf $m_{\mathcal{P}}$ hat. Für die Simulationen in dieser Arbeit werden hierzu die Momentenmatrizen von Lallemand und Luo [135] (D2Q9) und D’Humières et al. [47] (D3Q19) verwendet. Diese sind in Anhang C angegeben. Später wird jedoch auch gezeigt, dass die Wahl der höheren Momente keinen Einfluss auf das Verfahren hat (siehe Satz 1).

In gleicher Weise wie die Momentenmatrix M wird auch die inverse Momentenmatrix

$$T := M^{-1} \quad (3.5)$$

in Spalten aufgeteilt. Die Elemente dieser beiden Matrizen werden in der Folge mit m_{ij} und t_{ij} bezeichnet.

Statt der logarithmischen Entropiefunktion (3.1) wird hier eine quadratische Pseudo-Entropie verwendet

$$\tilde{H}(f) := - \sum_{i=0}^{Q-1} \left(\frac{f_{(i)}^2}{\omega_{(i)}} - f_{(i)} \right) = - \sum_{i=0}^{Q-1} \frac{f_{(i)}^2}{\omega_{(i)}} + \rho. \quad (3.6)$$

Diese entsteht durch eine Taylor-Entwicklung des Logarithmus in der Entropie (3.1) um den Entwicklungspunkt $f_{(i)}/w_{(i)} = 1$. Diese Wahl ist motiviert durch

$$f_i^{\text{eq}}(\rho = 1, \mathbf{u} = 0) = w_i.$$

Für Strömungen mit mittlerer Geschwindigkeit $\langle \mathbf{u} \rangle = 0$ und normierter Dichte $\rho_0 = 1$ bedeutet das, dass die Verteilungsfunktionen während der Simulation um w_i fluktuieren, also $\langle f_i \rangle \approx w_i$.

Während die logarithmische Entropie auf ein iteratives Schema führen würde, erhält die Pseudo-Entropie den expliziten Charakter der LBM. Ähnliche quadratische Approximationen wurde auch in den Referenzen [76, 119] diskutiert. Eine wichtige Eigenschaft der hier verwendeten Entropie-Funktion ist, dass sie ihr Maximum unter der Fixierung der hydrodynamischen Momente an der Gleichgewichtsverteilung (2.4) annimmt. Dies wird in Satz 2 bewiesen, siehe Paragraph 3.2.2.2.

In der Folge wird diese Pseudo-Entropie auf dem Raum der nicht-hydrodynamischen Momente maximiert.

3.2.1.3. Maximierung der Pseudo-Entropie

Die notwendige Bedingung des Optimierungsproblems (3.2) lässt sich schreiben als

$$\frac{\partial \tilde{H}}{\partial m_q} = 0, \quad q \in \mathcal{Q}.$$

Da die Pseudo-Entropie konkav ist, ist diese Bedingung auch hinreichend. Die Verteilungsfunktion ergibt sich aus den Momenten als $f_i = t_{ij} m_j$. Somit lässt sich die hinreichende Bedingung umformulieren zu

$$0 = \frac{\partial \tilde{H}}{\partial m_q} = - \sum_{i=0}^{Q-1} \frac{\partial f_{(i)}}{\partial m_q} \cdot \frac{2f_{(i)}}{\omega_{(i)}} = \sum_{i=0}^{Q-1} t_{(i)q} \left[- \frac{2}{\omega_{(i)}} (t_{(i)p} m_p + t_{(i)\hat{q}} m_{\hat{q}}) \right].$$

Dabei wird der Index p nur über \mathcal{P} summiert und \hat{q} nur über \mathcal{Q} . Beachte, dass die Dichte in Gleichung (3.6) auf dem Raum der nicht-hydrodynamischen Momente

konstant ist und daher in den Ableitungen verschwindet. Die Optimalitätsbedingung lässt sich umschreiben zu

$$\sum_{i=0}^{Q-1} \frac{t_{(i)q} t_{(i)q}}{\omega_{(i)}} m_{\hat{q}} = - \sum_{i=0}^{Q-1} \frac{t_{(i)q} t_{(i)p}}{\omega_{(i)}} m_p.$$

Mit $W = \text{diag}(1/\omega_0, 1/\omega_1, \dots, 1/\omega_{Q-1})$ lässt sich dies nach m_Q auflösen

$$m_Q = \underbrace{-(T_Q^T W T_Q)^{-1} (T_Q^{-1} W T_P)}_{=:A} \cdot m_P. \quad (3.7)$$

Diese Gleichung bestimmt also die optimierten nicht-hydrodynamischen Momente aus den hydrodynamischen Momenten. Die Matrix A hängt nur von der Momentenmatrix ab und ist somit während des gesamten Simulationslaufs und für alle Gitterpunkte konstant.

3.2.1.4. Allgemeine Formulierung des Algorithmus

Der resultierende pseudo-entropische Kollisionsschritt lässt sich also folgendermaßen formulieren (“Algorithmus 1”):

1. Berechne die hydrodynamischen Momente $m_P = M_P f$.
2. Bestimme die Gleichgewichte der hydrodynamischen Momente $m_P^{\text{eq}} = M_P f^{\text{eq}}$.
3. Relaxiere die hydrodynamischen Momente gemäß der vorgegebenen Relaxationszeit

$$m_P^{\text{pc}} := m_P - \frac{1}{\tau} (m_P - m_P^{\text{eq}}).$$

4. Bestimme die optimierten nicht-hydrodynamischen Momente $m_Q^{\text{pc}} = A \cdot m_P^{\text{pc}}$.
5. Transformiere die Momente zurück in den Raum der Geschwindigkeitsverteilungsfunktionen $f^{\text{pc}} = T \cdot m^{\text{pc}}$.

Dieses Verfahren kann in einer alternativen Formulierung als Stabilisierungsoperator B mit den meisten Kollisionsmodellen aus der Literatur kombiniert werden. Der Operator B ist definiert als

$$B = T \begin{pmatrix} I & 0 \\ A & 0 \end{pmatrix} M. \quad (3.8)$$

und kombiniert hierbei die Schritte 1, 4 und 5 von Algorithmus 1. Somit ergibt sich der Kollisionsschritt als (“Algorithmus 2”):

1. Kollidiere die Verteilungen f^{pc} mit einem beliebigen Kollisionsmodell.
2. Wende den pseudo-entropischen Operator B an $f^{\text{pc}} := B \cdot f^{\text{pc}}$.

In dieser Formulierung lässt sich die Pseudo-Entropische Stabilisierung mit wenig Aufwand in Lattice-Boltzmann-Codes einpflegen.

3.2.1.5. Spezialisierung für zweidimensionale Hydrodynamik

Im Folgenden werden Matrizen A und B für zwei- und dreidimensionale Strömungen bestimmt. Die Wahl der Momente folgt dabei [47, 135]. Die Momente und Transformationsmatrizen dieser Modelle sind in Anhang C angegeben.

Für das D2Q9-Modell werden die Momente aufgeteilt in hydrodynamische Momente $m_{\mathcal{P}} = (\rho \ j_x \ j_y \ e \ p_{(x)(x)} \ p_{xy})^T$ und nicht-hydrodynamische Momente $m_{\mathcal{Q}} = (\hat{\varepsilon} \ q_x \ q_y)^T$. Mit der Momentenmatrix aus Anhang C.1 ergeben sich die Operatoren der PES als

$$\begin{pmatrix} \hat{\varepsilon}^{\text{pc}} \\ q_x^{\text{pc}} \\ q_y^{\text{pc}} \end{pmatrix} = m_{\mathcal{Q}}^{\text{pc}} = A \cdot m_{\mathcal{P}}^{\text{pc}} = \begin{pmatrix} -1 & 0 & 0 & -1 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} \rho \\ j_x \\ j_y \\ e^{\text{pc}} \\ p_{(x)(x)}^{\text{pc}} \\ p_{xy}^{\text{pc}} \end{pmatrix}$$

und B , siehe Gleichung (C.1).

Somit ergeben sich die folgenden nicht-hydrodynamischen Momente:

$$\begin{aligned} \hat{\varepsilon}^{\text{pc}} &= -\rho - e^{\text{pc}}, \\ q_x^{\text{pc}} &= -j_x, \quad \text{and} \\ q_y^{\text{pc}} &= -j_y. \end{aligned}$$

Während q_x^{pc} und q_y^{pc} auf ihre Gleichgewichte gesetzt werden [135], hängt $\hat{\varepsilon}^{\text{pc}}$ explizit von einem Postkollisionsmoment zweiter Ordnung ab.

3.2.1.6. Spezialisierung für dreidimensionale Hydrodynamik

Mit der Momentenmatrix von d'Humieres et al. [47], siehe Anhang C.2, ergeben sich die hydrodynamischen Momente zu $(\rho \ e \ j_x \ j_y \ j_z \ 3p_{(x)(x)} \ p_{(w)(w)} \ p_{xy}p_{yz} \ p_{xz})^T$ und die nicht-hydrodynamischen Momente zu $(\hat{\varepsilon} \ q_x \ q_y \ q_z \ 3\pi_{(x)(x)} \ \pi_{(w)(w)} \ m_x \ m_y \ m_z)^T$.

Die so berechneten Matrizen A und B finden sich in Anhang C.2. Die nicht-hydrodynamischen Momente berechnen sich damit aus den hydrodynamischen Momenten als

$$\begin{aligned}\hat{\varepsilon}^{\text{pc}} &= -\frac{7}{38}\rho - \frac{11}{38}e^{\text{pc}}, \\ q_{x,y,z}^{\text{pc}} &= -\frac{2}{3}j_{x,y,z}, \\ 3\pi_{(x)(x)}^{\text{pc}} &= -\frac{1}{2}(3p_{(x)(x)}^{\text{pc}}), \\ \pi_{(w)(w)}^{\text{pc}} &= -\frac{1}{2}p_{(w)(w)}^{\text{pc}}, \\ m_{x,y,z}^{\text{pc}} &= 0.\end{aligned}$$

Hier werden $q_{x,y,z}^{\text{pc}}$ und $m_{x,y,z}^{\text{pc}}$ auf ihre Gleichgewichtsmomente [47] gesetzt, während die übrigen nicht-hydrodynamischen Momente von den Momenten zweiter Ordnung abhängen.

3.2.2. Theoretische Untersuchung

Im Verlauf dieser Arbeit werden die entwickelten Kollisionsmodelle im Hinblick auf ihre Stabilität und Genauigkeit getestet. Zunächst werden allerdings einige theoretische Aspekte diskutiert. Dieser Abschnitt zeigt die Unabhängigkeit des Verfahrens von der Wahl der nicht-hydrodynamischen Momente (Paragraph 3.2.2.1) und beweist die oben genannte Konsistenz der Gleichgewichtsverteilung mit der Pseudo-Entropiefunktion (Paragraph 3.2.2.2) .

3.2.2.1. Unabhängigkeit von der Wahl der Momente höherer Ordnung

Die PES ist unabhängig von der Definition der Momente höherer Ordnung. Solange die Momentenmatrix M regulär ist und die oben formulierte Orthogonalität erfüllt ist, ist das Verfahren in diesem Sinne universell. Diese Aussage wird im folgenden Satz bewiesen.

Satz 1 *Die stabilisierten Verteilungsfunktionen f^{pe} hängen nicht von der Wahl der Matrix $M_{\mathcal{Q}}$ ab, solange $M_{\mathcal{P}}M_{\mathcal{Q}}^T = 0$ und $M_{\mathcal{Q}}$ vollen Rang hat.*

BEWEIS Zunächst wird gezeigt, dass $T_{\mathcal{P}}$ und $T_{\mathcal{Q}}$ die Pseudoinversen von $M_{\mathcal{P}}$ und $M_{\mathcal{Q}}$ sind. Diese ergeben sich wegen der linearen Unabhängigkeit der Zeilen als

$$M_{\mathcal{P}}^+ = M_{\mathcal{P}}^T(M_{\mathcal{P}}M_{\mathcal{P}}^T)^{-1} \quad \text{und} \quad M_{\mathcal{Q}}^+ = M_{\mathcal{Q}}^T(M_{\mathcal{Q}}M_{\mathcal{Q}}^T)^{-1}$$

Wegen der Orthogonalität $M_{\mathcal{P}}M_{\mathcal{Q}}^T = 0$ gilt

$$\begin{pmatrix} M_{\mathcal{P}} \\ M_{\mathcal{Q}} \end{pmatrix} (M_{\mathcal{P}}^+ \ M_{\mathcal{Q}}^+) = \begin{pmatrix} I & 0 \\ 0 & I \end{pmatrix} = I.$$

Da die Inverse eindeutig ist, folgt $M_{\mathcal{P}}^+ = T_{\mathcal{P}}$ und $M_{\mathcal{Q}}^+ = T_{\mathcal{Q}}$. Demzufolge hängt $T_{\mathcal{P}}$ nicht von $M_{\mathcal{Q}}$ ab. Außerdem ist $M_{\mathcal{P}}T_{\mathcal{Q}} = 0$ und daher $\text{im}(T_{\mathcal{Q}}) = \ker(M_{\mathcal{P}})$.

Mit diesen Vorüberlegungen ergeben sich die stabilisierten Verteilungsfunktionen als

$$\begin{aligned} f^{\text{pe}} &\leftarrow \max_{m_{\mathcal{Q}}} ! \tilde{H}(f) \Leftrightarrow \max_{m_{\mathcal{Q}}} ! \tilde{H}(T_{\mathcal{P}} \cdot m_{\mathcal{P}}^{\text{pc}} + T_{\mathcal{Q}} \cdot m_{\mathcal{Q}}) \\ &\Leftrightarrow \max_{T_{\mathcal{Q}} \cdot m_{\mathcal{Q}} \in \text{im}(T_{\mathcal{Q}})} ! \tilde{H}(T_{\mathcal{P}} \cdot m_{\mathcal{P}}^{\text{pc}} + T_{\mathcal{Q}} \cdot m_{\mathcal{Q}}) \\ &\Leftrightarrow \max_{h \in \ker(M_{\mathcal{P}})} ! \tilde{H}(M_{\mathcal{P}}^T (M_{\mathcal{P}}M_{\mathcal{P}}^T)^{-1} \cdot m_{\mathcal{P}}^{\text{pc}} + h). \end{aligned}$$

In dieser Darstellung hängen die stabilisierten Verteilungsfunktionen nur noch von $M_{\mathcal{P}}$ und den relaxierten hydrodynamischen Momenten ab. Das Verfahren ist somit für alle $M_{\mathcal{Q}}$ mit den geforderten Eigenschaften äquivalent. □

Trotz dieser Äquivalenz hängt die Matrix A natürlich von der Wahl der Momente ab. Die Matrix B ist jedoch universell (solange die Verteilungsfunktionen in der gleichen Reihenfolge angeordnet sind).

3.2.2.2. Maximierung der Pseudo-Entropie

Als nächstes wird gezeigt, dass die quadratische Gleichgewichtsverteilung die Pseudo-Entropie maximiert, wenn die hydrodynamischen Momente so beschränkt werden wie von der Chapman-Enskog-Entwicklung gefordert.

Satz 2 Die quadratische Gleichgewichtsverteilung des BGK-Modells

$$f_i^{\text{eq}} := \omega_{(i)} \rho \left(1 + \frac{u_j e_{(i)j}}{c_s^2} + \frac{(u_j e_{(i)j})^2}{2c_s^4} - \frac{u_j u_j}{2c_s^2} \right)$$

maximiert die Pseudo-Entropie $\tilde{H}(f)$ unter den Nebenbedingungen

$$\sum_{i=0}^{Q-1} f_i = \rho, \tag{3.9}$$

$$\sum_{i=0}^{Q-1} f_{(i)} e_{(i)j} = \rho u_j, \tag{3.10}$$

$$\sum_{i=0}^{Q-1} f_{(i)} e_{(i)j} e_{(i)k} = \rho c_s^2 \delta_{jk} + \rho u_j u_k, \tag{3.11}$$

BEWEIS Für die (konkave) Pseudo-Entropiefunktion bildet sich eine notwendige und hinreichende Optimalitätsbedingung aus den Nebenbedingungen und der Gleichung

$$0 = \frac{\partial \mathcal{L}}{\partial f_i} = -\frac{2f_i}{\omega_i} + 1 + \vartheta + \vartheta_j e_{ij} + \vartheta_{jk} e_{(i)j} e_{(i)k}$$

Hierbei bezeichnen

$$\mathcal{L}(f, \vartheta, (\vartheta_j)_j, (\vartheta_{jk})_{jk})$$

die Lagrange-Funktion des Optimierungsproblems und ϑ, ϑ_j und ϑ_{jk} die Lagrange-multiplikatoren der Nebenbedingungen (3.9)–(3.11). Somit hat die optimale Verteilungsfunktion die Form

$$f_i = \frac{\omega^{(i)}}{2} (1 + \vartheta + \vartheta_j e_{(i)j} + \vartheta_{jk} e_{(i)j} e_{(i)k}). \quad (3.12)$$

Diese Form wird nun in die Nebenbedingungen eingesetzt. Mit den Gittertensoren aus Lemma B.1 ergibt sich

$$\rho = \frac{1 + \vartheta}{2} + \frac{c_s^2}{2} \vartheta_{jj}, \quad (3.13)$$

$$\rho u_j = c_s^2 \vartheta_j, \quad (3.14)$$

$$c_s^2 \rho \delta_{jk} + \rho u_j u_k = c_s^2 \delta_{jk} \left(\frac{1 + \vartheta}{2} + \frac{c_s^2}{2} \vartheta_{ll} \right) + c_s^4 \vartheta_{jk}. \quad (3.15)$$

Gleichung (3.14) liefert

$$\vartheta_j = \frac{\rho u_j}{c_s^2}.$$

Einsetzen von Gleichung (3.13) in (3.15) gibt

$$\vartheta_{jk} = \frac{\rho u_j u_k}{c_s^4},$$

und somit

$$\vartheta = 2\rho - 2\frac{u_j u_j}{c_s^2} - 1.$$

Einsetzen der Lagrangemultiplikatoren in die Form (3.12) liefert die quadratische Gleichgewichtsverteilung.

□

Dieses Ergebnis zeigt unter anderem, dass die PES für $\tau = 1$ mit dem BGK-Modell übereinstimmt. Im Folgenden werden die beiden Modelle in Simulationen unter aufgelöster Wirbelströmungen miteinander verglichen.

3.3. Simulationen mit der Pseudo-Entropischen Stabilisierung

Die Effektivität der oben hergeleiteten PES wird an zwei Beispielströmungen demonstriert: an zweidimensionalen Scherschichten [162] und an dreidimensionalen Taylor-Green-Wirbeln [21] bei verschiedenen Reynoldszahlen.

3.3.1. Zweidimensionale Scherschichten

In der Einleitung wurde die Scherschicht bereits als motivierendes Beispiel für die Entwicklung neuer Kollisionsmodelle herangezogen. Um den Effekt der PES besonders deutlich zu machen, wurde die Reynoldszahl für die Simulationen in diesem Abschnitt von $Re = 10\,000$ auf $Re = 30\,000$ erhöht, wodurch die Simulationen schneller instabil werden. Auch eine Verlängerung der Simulationszeit auf $t_{\max} = 3t_c$ sorgt für die Generierung kleinskaligerer Strukturen, die in unteraufgelösten Simulationen Instabilitäten auslösen können.

In [162] wurden verschiedene inkompressible Löser für zweidimensionale Scherschichten getestet. Alle dort betrachteten Verfahren erzeugten bei $Re = 10\,000$ und Auflösungen von weniger als 256×256 Gitterpunkten sekundäre Strukturen. Auch die in der Einleitung diskutierten Simulationen mit dem BGK-Modell wiesen auf einem Gitter mit 128×128 Punkten sekundäre Wirbel auf, siehe Abbildung 2.4. In [18] wurden verschiedene KBC-Varianten und die Entropische LBM auf Scherströmungen mit $Re = 30\,000$ angewandt. Auf einem Gitter mit 128×128 Punkten konnte nur eine von vier KBC-Varianten die Entstehung der unphysikalischen Wirbel verhindern. Diese Variante nutzte eine zentrale Momentenbasis und bezog ein zweites Moment, nämlich die Spur des Spannungstensors, in die Optimierung ein. Die Entropische LBM erzeugte sogar schon in einer früheren Phase der Simulation sekundäre Wirbel als die KBC-Modelle. Diese Ergebnisse zeigen, dass das hier betrachtete Problem auch für hoch entwickelte LBMs und für inkompressible Navier-Stokes-Löser problematisch ist.

In dieser Arbeit wurden mit dem BGK-Modell und der PES Simulationen mit verschiedenen Auflösungen durchgeführt. Die Verteilungsfunktionen wurden durch Gleichgewichtsverteilungen initialisiert. Während die Rechnungen mit dem BGK-Modell erst ab einer Auflösung von 256×256 stabil liefen, waren mit der PES alle Simulationen stabil.

Abbildung 3.1 zeigt die Wirbelstärkefelder zu den Zeitpunkten $t/t_c = 1.25$ und $t/t_c = 2.5$. In keiner Simulation traten unphysikalische Wirbel auf. Selbst bei einer Auflösung von 16×16 Punkten blieb die Struktur der grobskaligen Wirbel erhalten.

Um die Ergebnisse auch quantitativ auszuwerten, wurden der zeitliche Verlauf der normierten kinetischen Energie $\langle u_i u_i \rangle / u_0^2$ und Enstrophie $\langle \omega^2 \rangle / u_0^2$ sowie deren nor-

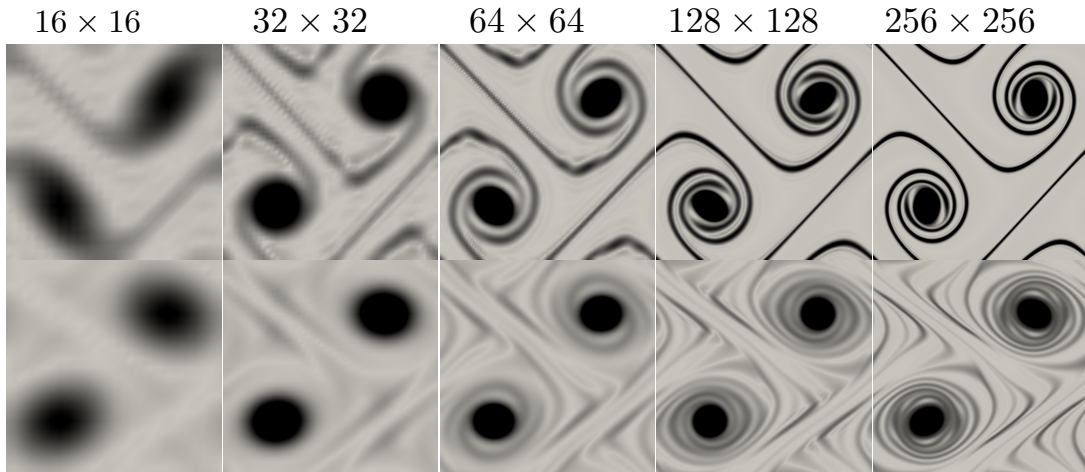


Abbildung 3.1: Betrag der Wirbelstärke $|\omega|$ zu den Zeitpunkten $t/t_c = 1.25$ (obere Reihe) und $t/t_c = 2.5$ (untere Reihe) für Simulationen der zweidimensionalen Scherschicht mit der PES. Die Wirbelstärkefelder waren für alle Auflösungen von 16×16 bis 256×256 Gitterpunkten frei von sekundären, unphysikalischen Wirbeln. Mit dem BGK-Modell waren die Simulationen erst ab der höchsten Auflösung stabil. Die Farbskala wurde bei $|\omega| = 0.7$ abgeschnitten, um die Struktur der Wirbel zu betonen.

mierte Standardabweichungen $\text{Std}[u_i u_i] / \langle u_j u_j \rangle$ und $\text{Std}[\omega^2] / \langle \omega^2 \rangle$ mit einer Referenzlösung aus [18] verglichen. Dabei bezeichnet $\langle \cdot \rangle$ die Mittelung über das Strömungsgebiet und $\text{Std}[\cdot] := \sqrt{\langle \cdot^2 \rangle - \langle \cdot \rangle^2}$ die Standardabweichung.

Bei einer Auflösung von 256×256 Gitterpunkten stimmten die Ergebnisse der PES und des BGK-Modells exzellent miteinander überein, siehe Abbildung 3.2. Jedoch lieferte die PES auch mit der halben Auflösung noch insgesamt sehr ähnliche Statistiken, während die Simulationen mit dem BGK-Modell schon vor $t = t_c$ instabil wurden, siehe Abbildung 3.3.

3.3.2. Dreidimensionale Taylor-Green-Wirbel

Um die Stabilität der PES auch für dreidimensionale Strömungen zu bestätigen, wurden dreidimensionale Taylor-Green-Wirbel [21] simuliert. Im Gegensatz zum zweidimensionalen Taylor-Green-Wirbel aus Abschnitt 2.6.1 zerfallen die Wirbel im Dreidimensionalen in immer kleinere Strukturen. Diese Transition zu einer turbulenten Strömung ist bei unteraufgelösten Simulationen mit dem BGK-Modell problematisch. Die Anfangsbedingungen wurden auf dem periodischen Gebiet $[0, 2\pi]^3$ vorgegeben

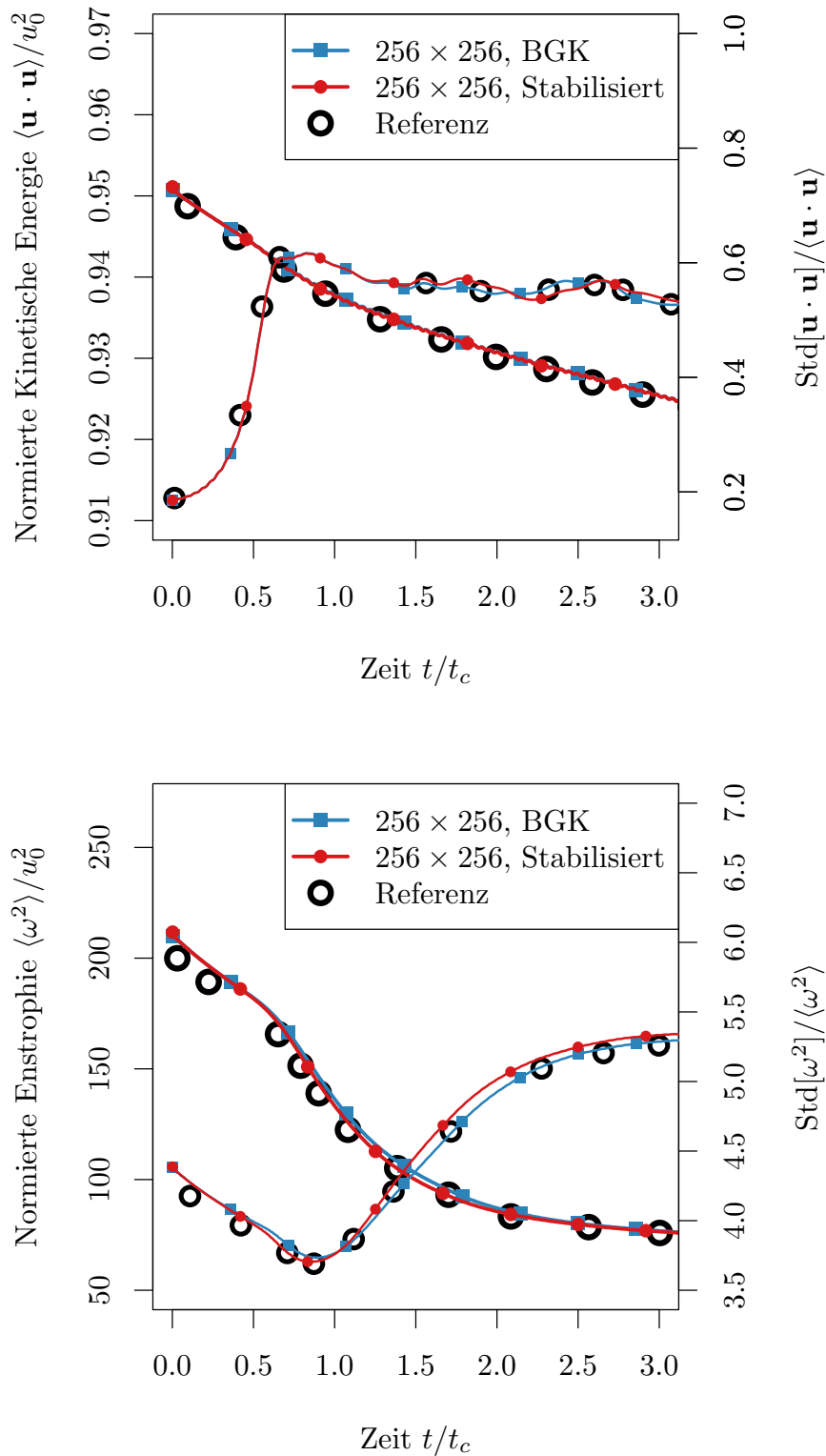


Abbildung 3.2: Evolution der normierten kinetischer Energie und Enstrophie in Simulationen der zweidimensionalen Scherschicht mit der PES und dem BGK-Modell im Vergleich zur Referenz von Bösch et al. [18]. Dicke Linien und Punkte repräsentieren die normierte kinetische Energie und Enstrophie, dünne Linien und Punkte deren normierte Standardabweichungen. Die Auflösung war 256×256 Gitterpunkte.

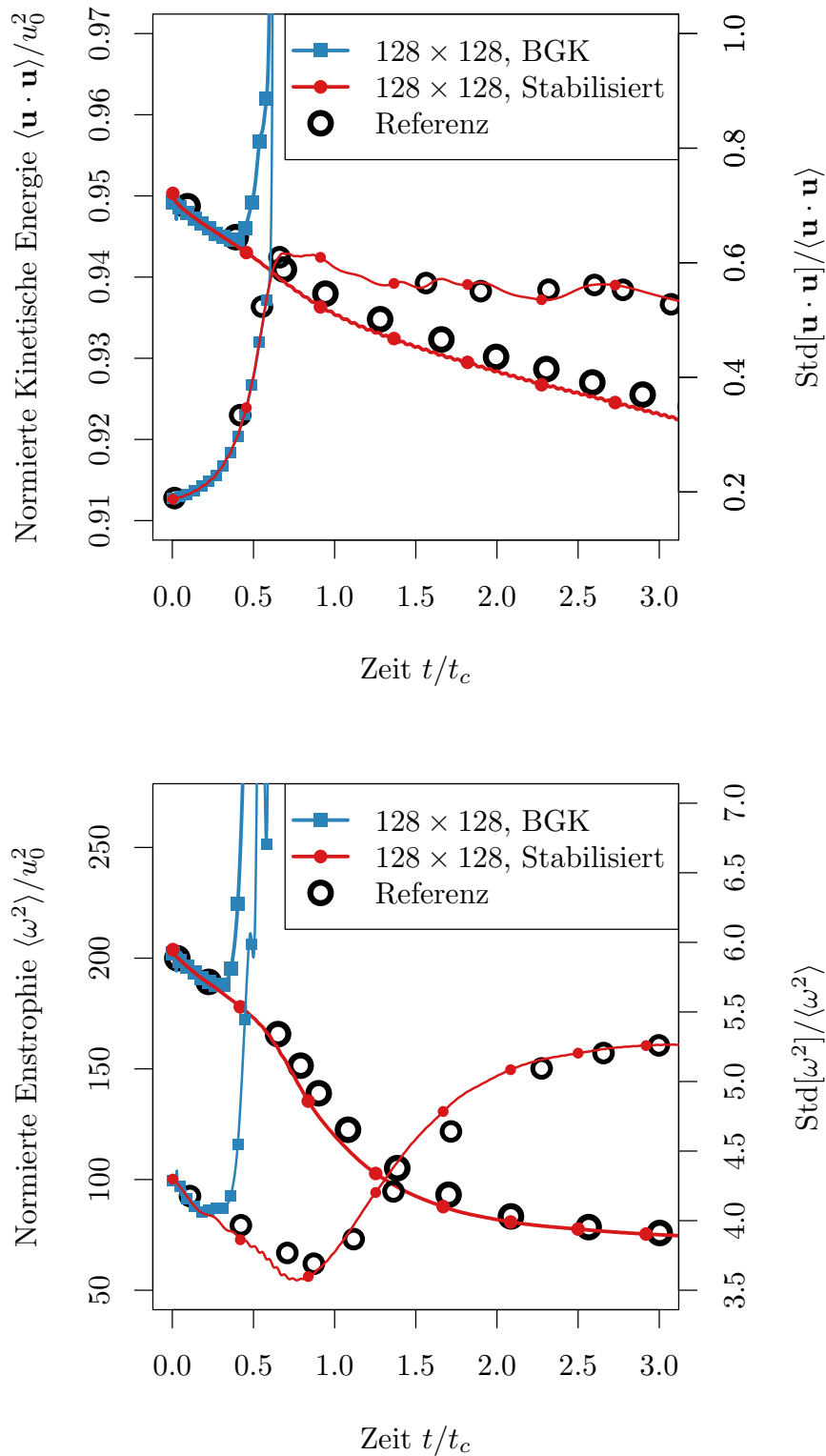


Abbildung 3.3: Evolution der normierten kinetischer Energie und Enstrophie in Simulationen der zweidimensionalen Scherschicht mit der PES und dem BGK-Modell im Vergleich zur Referenz von Bösch et al. [18]. Dicke Linien und Punkte repräsentieren die normierte kinetische Energie und Enstrophie, dünne Linien und Punkte deren normierte Standardabweichungen. Die Auflösung war 128×128 Gitterpunkte.

durch

$$\begin{aligned} u_1(x, y, z) &= \sin x \cos y \cos z, \\ u_2(x, y, z) &= -\cos x \sin y \cos z, \\ u_3(x, y, z) &= 0, \\ p(x, y, z) &= \frac{1}{16} [\cos(2x) + \cos(2y)] [\cos(2z + 2)]. \end{aligned}$$

Die Reynoldszahl war definiert als $Re = 1/\nu$.

Es wurden die Evolution der kinetischen Energie k und Enstrophie \mathcal{E} betrachtet, die wie folgt definiert waren

$$k = \frac{1}{(2\pi)^3} \int_0^{2\pi} \int_0^{2\pi} \int_0^{2\pi} \frac{1}{2} u_i u_i \, dx \, dy \, dz$$

und

$$\mathcal{E} = \frac{1}{(2\pi)^3} \int_0^{2\pi} \int_0^{2\pi} \int_0^{2\pi} \boldsymbol{\omega} \cdot \boldsymbol{\omega} \, dx \, dy \, dz \quad \text{mit} \quad \boldsymbol{\omega} = \nabla \times \mathbf{u}.$$

In inkompressiblen Strömungen sind diese beiden Observablen über die Bilanzgleichung der turbulenten kinetischen Energie gekoppelt. Da in homogenen Strömungen die Druckdilatation, der turbulente Transport und die Produktion verschwinden, vereinfacht sich die Bilanzgleichung in diesem Fall zu

$$-\frac{dk}{dt} = \epsilon.$$

Die turbulente Dissipationsrate ϵ lässt sich im Inkompressiblen als $\epsilon = \nu \mathcal{E}$ aus der Enstrophie bestimmen. Somit hängen kinetische Energie und Enstrophie in inkompressibler, homogener Turbulenz wie folgt zusammen:

$$-\frac{dk}{dt} = \nu \mathcal{E}.$$

Berechnet man nun die Dissipationsrate $-dk/dt$ über zeitliche finite Differenzen, so beschreibt die linke Seite dieser Gleichung alle physikalischen Einflüsse auf die kinetische Energie, schließt aber numerische Einflüsse mit ein. Letztere können zum Beispiel durch Filterung, Zeitdiskretisierung, Pseudo-Kompressibilität oder Subgrid-Modelle zustande kommen. Im Gegensatz dazu ergibt sich die rechte Seite aus der exakten Definition der viskosen Einflüsse auf die lokalen Geschwindigkeitsgradienten und lässt damit numerische Effekte außen vor, siehe dazu zum Beispiel [67, 193].

Deshalb wurden die Ortsableitungen in der Enstrophie in der folgenden Analyse mit einer Genauigkeit fünfter Ordnung berechnet. Gemäß [67, 69, 173, 193] wurde die

Dissipationsrate $-dk/dt$ über finite Differenzen (mit einem Inkrement von $\tilde{\delta}_t = 1/6$) berechnet. Somit ergab sich eine Schätzung für die effektive Reynoldszahl als

$$Re_{\text{eff}} = -\frac{\mathcal{E}}{\frac{dk}{dt}}.$$

In den folgenden Simulationen wurde die Reynoldszahl von 100 bis 100 000 variiert, um die Stabilität und Genauigkeit der PES für Simulationen unteraufgelöster Turbulenz zu bewerten und mit dem BGK-Modell zu vergleichen. Alle Simulationen wurden bei einer konstanten Auflösung von $80 \times 80 \times 80$ Gitterpunkten durchgeführt.

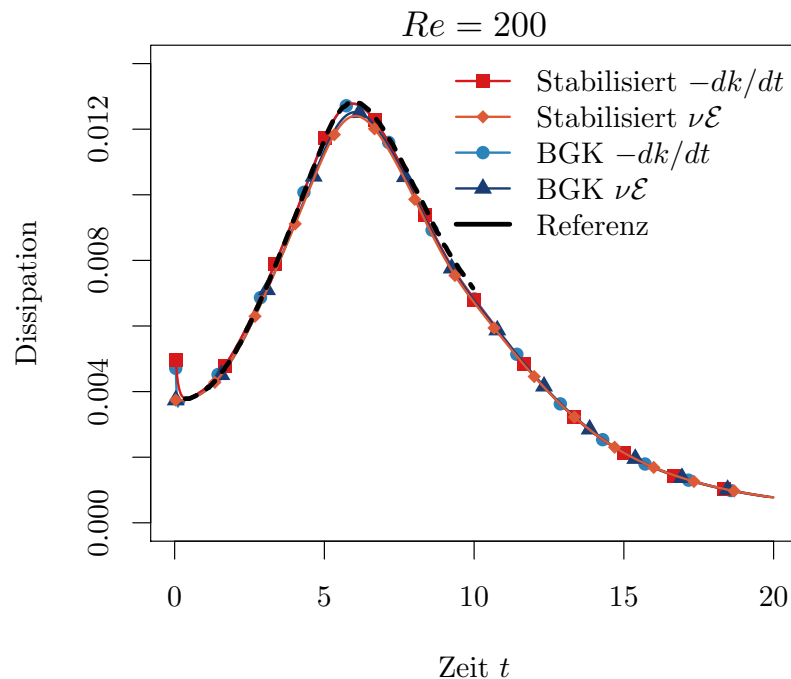


Abbildung 3.4: Dissipation der kinetischen Energie in Simulationen des dreidimensionalen Taylor-Green-Wirbels mit dem BGK-Modell und der PES auf einem Gitter mit 80^3 Punkten. Die Reynoldszahl war $Re = 200$. Die Dissipationsraten wurden über finite Differenzen ($-dk/dt$) und über die Enstrophie ($\nu\mathcal{E}$) berechnet. Zum Vergleich ist die Referenzlösung aus [21] eingetragen.

Abbildung 3.4 zeigt zunächst die Dissipation bei einer moderaten Reynoldszahl von $Re = 200$. Hier stimmten sowohl die numerisch als auch die direkt berechneten Dissipationsraten für beide Modelle nahezu exakt mit der Referenzlösung aus [21] überein. Demzufolge stimmte auch die effektive Reynoldszahl während beider Simulationsläufe gut mit der vorgeschriebenen Reynoldszahl überein.

Als nächstes wurden die gleichen Simulationen bei $Re = 800$ durchgeführt. Während die numerisch ermittelten Dissipationsraten qualitativ miteinander und mit der Referenzlösung übereinstimmten, zeigte sich bei den Enstrophie-basierten Dissipations-

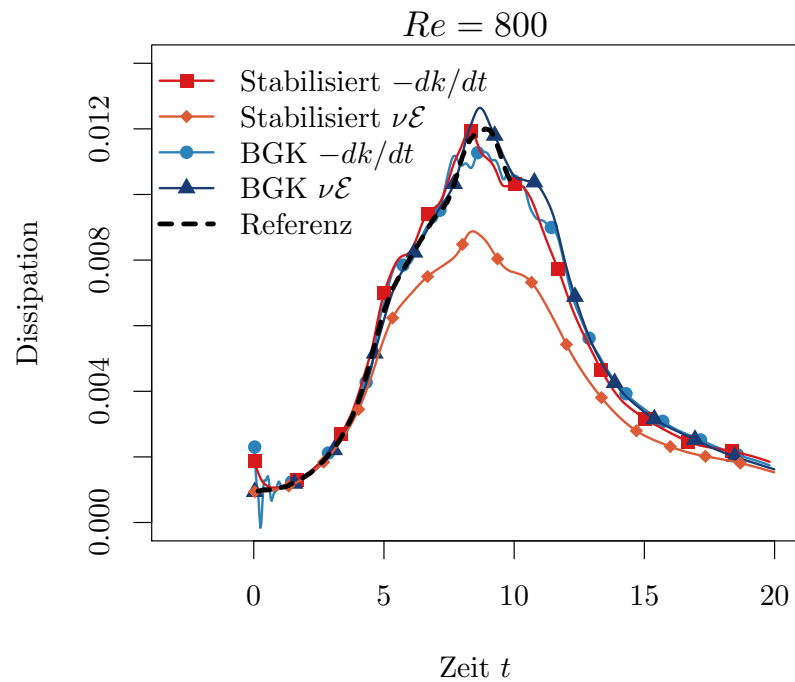
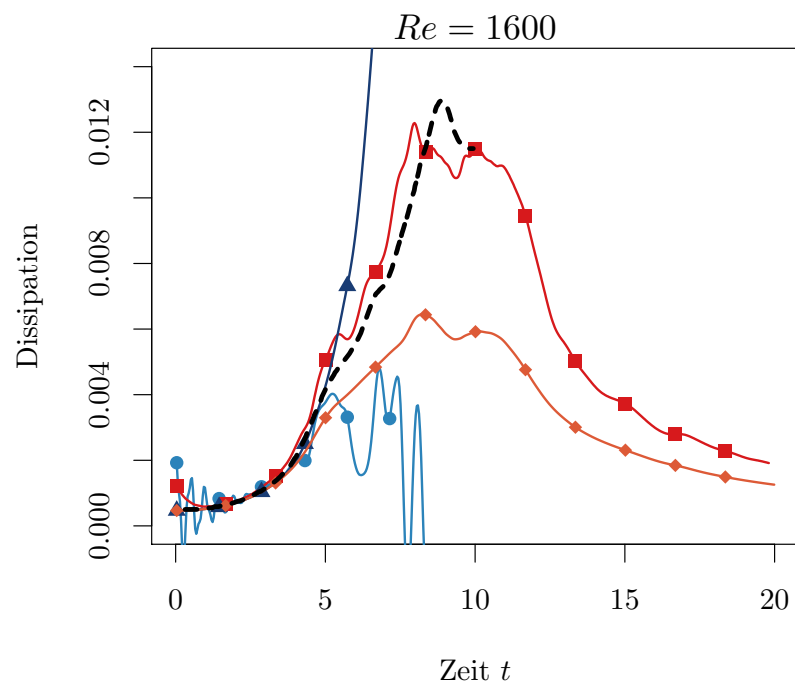
(a) Dissipation der kinetischen Energie für $Re = 800$.(b) Dissipation der kinetischen Energie für $Re = 1600$.

Abbildung 3.5: Dissipation der kinetischen Energie in Simulationen des dreidimensionalen Taylor-Green-Wirbels mit dem BGK-Modell und der PES auf einem Gitter mit 80^3 Punkten. Die Dissipationsraten wurden über finite Differenzen ($-dk/dt$) und über die Enstrophie ($\nu\mathcal{E}$) berechnet. Zum Vergleich ist die Referenzlösung aus [21] eingetragen.

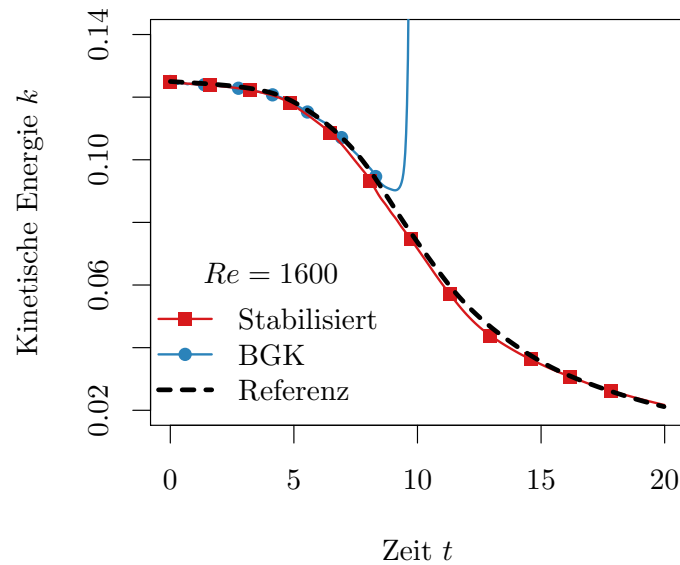


Abbildung 3.6: Bilanz der kinetischen Energie in Simulationen des dreidimensionalen Taylor-Green-Wirbels mit dem BGK-Modell und der PES auf einem Gitter mit 80^3 Punkten. Die Reynoldszahl war $Re = 1600$. Zum Vergleich ist die Referenzlösung aus [40] eingetragen.

raten ein Unterschied. Mit dem Anwachsen des turbulenten Spektrums unterschätzte die PES zunehmend die Enstrophie, am Hochpunkt der Dissipationsrate um circa 75%. Demzufolge verfehlte die effektive Reynoldszahl die vorgeschriebene Reynoldszahl in dieser Phase der Simulation um 25%. Hier führte die Optimierung der höheren Momente in unteraufgelöster Turbulenz also zu einer Glättung der Geschwindigkeitsgradienten. Diese Dämpfung beeinflusste aber die Genauigkeit der makroskopischen Dissipationsrate kaum.

Bei einer weiteren Erhöhung der Reynoldszahl wurden die Simulationen mit dem BGK-Modell instabil. Abbildung 3.5b zeigt die Dissipationsraten für $Re = 1600$. Ab $t = 5$ erzeugte das BGK-Modell Oszillationen, die sich vor $t = 10$ zu einer negativen Dissipationsrate aufschaukelten. In dieser Phase wuchs die kinetische Energie unkontrolliert an. Bei den Rechnungen mit der PES verfehlte die skalierte Enstrophie die Referenzlösung um circa 50%. Dafür lieferte die PES immer noch eine qualitativ gute Schätzung der makroskopischen Dissipationsrate. Abbildung 3.6 zeigt, dass die Bilanz der kinetischen Energie mit einer Referenzlösung aus [40] gut übereinstimmte.

Zum Schluss wurde die Reynoldszahl weiter erhöht. Vollaufgelöste Referenzlösungen existieren in der Literatur bis $Re = 5000$, jedoch war die PES auch für sehr viel größere Reynoldszahlen noch stabil (bis zu $Re = 100\,000$) Abbildung 3.7 zeigt die Dissipationsraten und den Verlauf der effektiven Reynoldszahl für $Re = 15\,000$. Der Simulationslauf wurde auf $t_{\max} = 40$ verlängert, um zu zeigen, dass die Rechnung bis zum Abklingen der Dissipationsrate stabil blieb. Im Bereich des Hochpunkts war

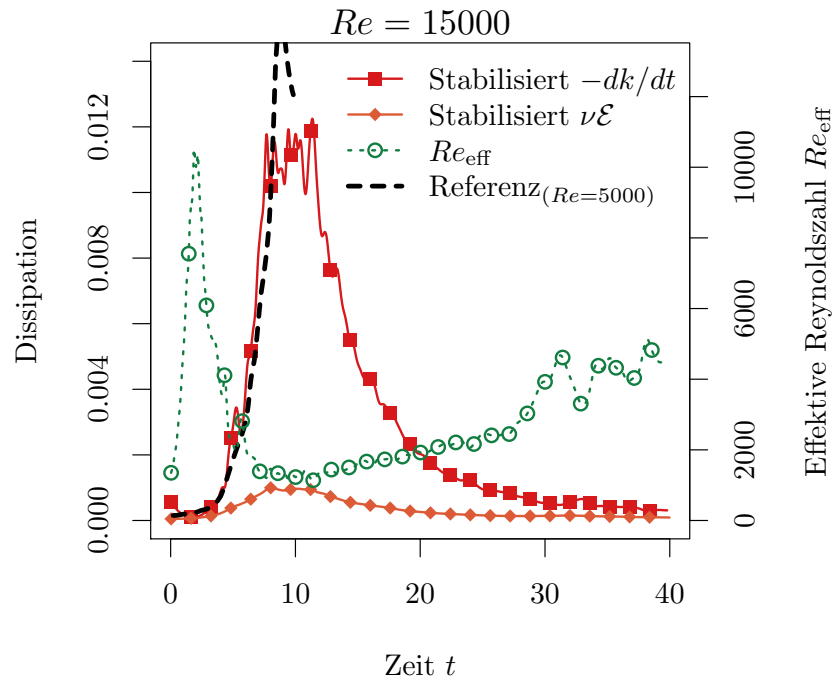


Abbildung 3.7: Dissipation der kinetischen Energie und effektive Reynoldszahl Re_{eff} in Simulationen des dreidimensionalen Taylor-Green-Wirbels mit der PES auf einem Gitter mit 80^3 Punkten. Die Reynoldszahl war $Re = 15\,000$. Die Dissipationsraten wurden über finite Differenzen ($-dk/dt$) und über die Enstrophie ($\nu\mathcal{E}$) berechnet. Zum Vergleich ist die Referenzlösung für $Re = 5000$ aus [20] eingetragen.

die effektive Reynoldszahl um eine Größenordnung kleiner als die vorgeschriebene. Dennoch lieferte die makroskopische Dissipationsrate insbesondere in der ersten Phase der Simulation noch eine akzeptable Approximation der Referenzlösung.

Abbildung 3.8 zeigt die gemittelten effektiven Reynoldszahlen und fasst damit die Ergebnisse für die dreidimensionalen Taylor-Green-Wirbel zusammen. Bei den Rechnungen mit dem BGK-Modell stimmten die effektiven Reynoldszahlen immer gut mit den vorgeschriebenen Reynoldszahlen überein, für $Re > 800$ wurden die Simulationen jedoch instabil. Die PES lieferte im Bereich kleiner Reynoldszahlen gleichwertige Ergebnisse. Bei größeren Reynoldszahlen glättete die PES die Oszillationen in den Momenten höherer Ordnung und somit die effektive Reynoldszahl. Dafür blieben auch stark unteraufgelöste Simulationen stabil.

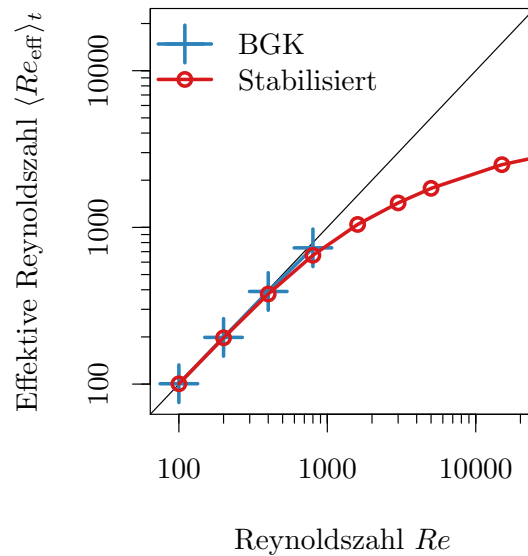


Abbildung 3.8: Gemittelte effektive Reynoldszahlen $\langle Re_{\text{eff}} \rangle_t$ für das BGK-Modell und die PES auf einem Gitter mit 80^3 Punkten. Alle Simulationen mit dem BGK-Modell und $Re > 800$ waren instabil.

3.4. Diskussion der Pseudo-Entropischen Stabilisierung

Dieser Abschnitt widmet sich der Diskussion der oben hergeleiteten PES. Das Modell wird in Unterabschnitt 3.4.1 theoretisch mit anderen Kollisionsmodellen verglichen. Dabei wird deutlich, dass die oben verwendeten Spezialisierungen der PES mit der Regularisierten LBM [139] übereinstimmen, obwohl die Herleitung und Formulierung sich grundsätzlich unterscheiden. Diese Äquivalenz hat interessante Konsequenzen für die Interpretation der Regularisierten LBM und für die Beziehung unterschiedlicher Kollisionsmodelle. Zum Abschluss werden die wesentlichen Ergebnisse dieses Kapitels in Abschnitt 3.4.2 zusammengefasst.

3.4.1. Vergleich mit anderen Kollisionsmodellen

In diesem Abschnitt wird die Pseudo-Entropische Stabilisierung mit den in Unterabschnitt 3.1 vorgestellten Kollisionsmodellen verglichen.

3.4.1.1. Vergleich mit der Regularisierten LBM

Die Regularisierte LBM [139] minimiert den Einfluss der nicht-hydrodynamischen Momente, indem die asymptotische Entwicklung der Verteilungen nach dem letzten physikalisch relevanten Glied abgeschnitten wird. In [139] wurde gezeigt, dass die Regularisierte LBM in einer bestimmten Momentenbasis als MRT-Modell zu interpretieren ist. Im Folgenden wird gezeigt, dass die Regularisierte LBM mit den betrachteten Spezialisierungen der PES äquivalent ist.

Dazu werden die Verteilungsfunktionen zunächst in eine Hermite-Reihe entwickelt.

$$f_i = w_{(i)} \sum_{k=0}^{Q-1} \mathcal{H}_{(ki)} m_{(k)}. \quad (3.16)$$

Hierbei bezeichnen $\mathcal{H}_{ki} := \mathcal{H}_k(\mathbf{e}_i)$ die Hermite-Basispolynome und m_k die dazu gehörigen Momente. Eine solche Hermite-Basis wird für die D2Q9- und D3Q27-Sterne in [41–43] definiert. Für die folgende Analyse spielt der Grad der höheren Hermite-Polynome keine Rolle, sodass durch eine geeignete Auswahl von Hermite-Polynomen eine Basis der Q -dimensionalen diskreten Verteilungsfunktionen erzeugt werden kann. Für den D2Q9-Stern beispielsweise ergeben sich die Hermite-Polynome der hydrodynamischen Momente durch die Tensoren 1 , e_{ij} , $e_{(i)j}e_{(i)k} - c_s^2\delta_{jk}$ und werden durch die Tensoren $|\mathbf{e}_i|^4 - (15/9)|\mathbf{e}_i|^2 + 2/9$ und $e_{(i)j}(|\mathbf{e}_{(i)}|^2 - 4/3)$ zu einer vollständigen Basis ergänzt [42].

Wegen der Konstruktion des Geschwindigkeitsraums (siehe Abschnitte 2.4.1 und 2.4.2) sind die Hermiteschen Basispolynome orthogonal bezüglich des gewichteten Skalarprodukts

$$\langle f, g \rangle_w := \sum_i w_{(i)} f_{(i)} g_{(i)},$$

denn

$$\langle \mathcal{H}_k, \mathcal{H}_l \rangle_w = \sum_i w_{(i)} \mathcal{H}_{k(i)} \mathcal{H}_{l(i)} = \int e^{-\xi^2} \mathcal{H}_k(\xi) \mathcal{H}_l(\xi) d\xi = 0 \quad \text{für } k \neq l.$$

Gleichung (3.16) macht klar, dass durch die Hermite-Basis eine inverse Momententransformation definiert ist als $t_{ij} = w_{(i)} \mathcal{H}_{j(i)}$. Die Matrix $T = (t_{ij})_{ij} = (T_{\mathcal{P}} \ T_{\mathcal{Q}})$ (siehe Gleichung (3.5)) und die dadurch induzierten Momente $m_{\mathcal{P}}$ und $m_{\mathcal{Q}}$ werden nun in die PES eingesetzt.

Dazu kann zunächst festgestellt werden, dass der lineare Term der quadratischen Pseudo-Entropie keinen Einfluss auf die Maximierung hat, da er lediglich die Dichte bezeichnet, die auf dem Unterraum der nicht-hydrodynamischen Momente konstant ist. Somit lässt sich die Maximierung der Pseudo-Entropie (unter den gleichen Nebenbedingungen wie oben) äquivalent formulieren als

$$\min_{m_{\mathcal{Q}}} \|f\|_H^2,$$

wobei $\|f\|_H$ die induzierte Norm des Skalarprodukts

$$\langle f, g \rangle_H := \sum_i \frac{f^{(i)}g^{(i)}}{w^{(i)}}$$

darstellt. Durch Einsetzen der Hermite-Basis ergibt sich

$$\|f\|_H^2 = \|T_{\mathcal{P}}m_{\mathcal{P}} + T_{\mathcal{Q}}m_{\mathcal{Q}}\|_H^2 = \|T_{\mathcal{P}}m_{\mathcal{P}}\|_H^2 + \|T_{\mathcal{Q}}m_{\mathcal{Q}}\|_H^2 + \langle T_{\mathcal{P}}m_{\mathcal{P}}, T_{\mathcal{Q}}m_{\mathcal{Q}} \rangle_H$$

und das Skalarprodukt verschwindet wegen

$$\begin{aligned} \langle T_{\mathcal{P}}m_{\mathcal{P}}, T_{\mathcal{Q}}m_{\mathcal{Q}} \rangle_H &= \sum_i \frac{t^{(i)\mathcal{P}}m_{\mathcal{P}}t^{(i)\mathcal{Q}}m_{\mathcal{Q}}}{w^{(i)}} = \sum_i \frac{w^{(i)}\mathcal{H}_{\mathcal{P}^{(i)}}m_{\mathcal{P}}w^{(i)}\mathcal{H}_{\mathcal{Q}^{(i)}}m_{\mathcal{Q}}}{w^{(i)}} \\ &= \langle \mathcal{H}_{\mathcal{P}^{(i)}}\mathcal{H}_{\mathcal{Q}^{(i)}} \rangle_w m_{\mathcal{P}}m_{\mathcal{Q}} = 0. \end{aligned}$$

(Hierbei werden die Indizes p und q wieder nur jeweils über \mathcal{P} und \mathcal{Q} summiert.)
Somit erhält man

$$\min_{m_{\mathcal{Q}}} \|f\|_H^2 \Leftrightarrow \min_{m_{\mathcal{Q}}} \|T_{\mathcal{P}}m_{\mathcal{P}}\|_H^2 + \|T_{\mathcal{Q}}m_{\mathcal{Q}}\|_H^2 \Leftrightarrow m_{\mathcal{Q}} = 0.$$

Also wird die Pseudo-Entropie in der Hermite-Basis optimal genau dann, wenn die höheren Momente verschwinden, was genau der Regularisierten LBM entspricht.

Nun muss überprüft werden, ob auch der Suchraum der höheren Momente in der Hermite-Basis und in der ursprünglichen Basis äquivalent sind. Dies hängt von der verwendeten Basis ab. Die Berechnung der Basiswechsellmatrix (siehe Tabelle C.2) von der D2Q9-Basis von Lallemand und Luo [135] in die Hermite-Basis [42] zeigt aber, dass die optimierten höheren Momente keinen Einfluss auf die hydrodynamischen Momente der Hermite-Basis haben. Also sind für den D2Q9-Stern auch die Suchräume gleich und die beiden Methoden sind äquivalent. Diese Äquivalenz gilt auch für die D3Q19-Basis von D’Humières [47], was aber in dieser Arbeit nur numerisch und nicht analytisch verifiziert wurde.

Dieser Beweis identifiziert die PES als eine alternative Formulierung der Regularisierten LBM. Die Äquivalenz wurde auch in praktischen Simulationen der zweidimensionalen Scherschicht und des dreidimensionalen Taylor-Green-Wirbels bestätigt, in denen die PES und die Regularisierte LBM exakt gleiche Ergebnisse lieferten. Im Gegensatz zur Regularisierten LBM ist die Formulierung der PES (Algorithmus 2) aber unabhängig von der Relaxation der Schermomente und von der verwendeten Gleichgewichtsverteilung. Damit ist sie flexibel mit anderen Modellen kombinierbar. Die Herleitung der PES bietet außerdem neue Einblicke in den Mechanismus der Regularisierten LBM: Die Regularisierung sorgt konstant für eine Randomisierung der lokalen Verteilungsfunktionen. Dadurch wird unter Berücksichtigung der hydrodynamischen Restriktionen eine maximal glatte Verteilung angenähert. Somit liefert die PES eine entropiegestützte Erklärung für die Stabilität der Regularisierten LBM.

3.4.1.2. Vergleich mit der Entropischen LBM

Während die Entropische LBM das Boltzmannsche H -Theorem erfüllt, garantiert die PES keinen Anstieg der Entropie. Dafür vermeidet die PES die iterative Lösung von nichtlinearen Gleichungen. Vielmehr ist das Verfahren durch die Verwendung einer quadratischen Pseudo-Entropie explizit und rechnerisch effizient.

Während die Entropische LBM (sowie auch andere Subgrid-Modelle in der LBM) den lokalen Relaxationsparameter und somit die Viskosität lokal erhöhen, erhält die PES die physikalische Relaxation der Schermomente. Allerdings begünstigt die Optimierung der nicht-hydrodynamischen Momente eine Erhöhung der Hyper-Viskosität. Die so induzierte Diffusion in den nicht-hydrodynamischen Momenten schlägt sich im Lauf der Simulationen auch auf die physikalischen Gradienten nieder. Dieser Effekt ist jedoch bei unteraufgelösten Simulationen durchaus erwünscht, da stabile Simulationen nur dann möglich sind, wenn die hochfrequenten Oszillationen lokal gedämpft werden [111].

3.4.1.3. Vergleich mit MRT-Modellen

MRT-Modelle relaxieren die verschiedenen Momente mit unterschiedlichen Relaxationsparametern, die vor der Simulation gewählt werden müssen. Diese Relaxationsparameter können zum Beispiel durch eine lineare Stabilitätsanalyse festgelegt werden – eine grobe Schätzung angesichts der Nichtlinearität der Strömung. Ob die so gewählten Relaxationsparameter sich für eine Simulation eignen, kann von Fall zu Fall variieren und ist zu Beginn der Simulation kaum abzusehen. Darüber hinaus hängen MRT-Modelle von der Wahl der Momente und Gleichgewichtsverteilungen ab.

Der Vergleich mit dem Regularisierten Modell macht klar, dass die PES als MRT-Modell mit voller Relaxation der nicht-hydrodynamischen Momente in einer Hermite-Basis formuliert werden kann. Im Idealfall sind MRT-Modelle so konstruiert, dass eine volle Relaxation der nicht-hydrodynamischen Momente die Stabilität verbessert. In diesem Kontext kann das MRT-Modell von Dellar [42] so verstanden werden, dass die höheren Momente der Gleichgewichtsverteilung die quadratische Pseudo-Entropie \tilde{H} maximieren. Da die Informationsentropie H ein aussagekräftiger Indikator für die Stabilität ist, kann auf dieser Grundlage ein theoretischer Vorteil gegenüber anderen MRT-Modellen argumentiert werden.

3.4.1.4. Vergleich mit dem entropischen MRT-Modell

Der Grundansatz der PES, nämlich die Maximierung einer Entropie-Funktion zur Festlegung der höheren Momente, wurde als erstes im entropischen MRT-Modell von Karlin, Bösch und Chikatamarla (KBC) formuliert. KBC ist ein Two-Relaxation-Time-Modell, welches die höheren Momente mit einem gemeinsamen Relaxations-

parameter relaxiert. Dieser Relaxationsparameter wird an jedem Lattice-Knoten so gewählt, dass eine Entropiefunktion maximiert wird.

KBC unterscheidet sich in drei Punkten von der PES. Zunächst hängt KBC von der Wahl der Momente ab, da die nicht-hydrodynamischen Momente *relaxiert* werden. Die Wahl der Momente beeinflusst die Stabilität des Verfahrens, was zum Beispiel in [18] demonstriert wird. Des Weiteren relaxiert KBC alle hydrodynamischen Momente mit der gleichen Rate, während die PES die nicht-physikalischen Momente ohne zusätzliche künstliche Restriktionen optimiert. Dies wird ermöglicht durch die Verwendung einer global definierten quadratischen Pseudo-Entropie. Im Gegensatz dazu verwendet KBC eine quadratische Taylorentwicklung der Entropie um das lokale Gleichgewicht, während die Pseudo-Entropiefunktion in der PES global konstant ist.

Einerseits weisen die optimierten Momente der PES darauf hin, dass eine unabhängige Optimierung der nicht-physikalischen Momente im Allgemeinen auf sehr unterschiedliche Relaxationsraten führen dürfte. Andererseits ist die lokal approximierte Entropie in KBC eine genauere Approximation der Entropiefunktion als die hier verwendete global quadratische Entropie. An dieser Stelle kann daher noch keine Aussage darüber gemacht werden, welcher Ansatz vorzuziehen ist, die Simulationen der Scherschichten zeigen im Vergleich mit [18] jedoch Vorteile der PES. Auch in Bezug auf ihre Implementierung ist die PES einfacher. KBC berechnet alle höheren Momente und deren Gleichgewichte sowie den optimierten Relaxationsparameter. Diese Schritte fallen bei der PES weg, da sich die nicht-hydrodynamischen Postkollisionsmomente direkt aus den hydrodynamischen ergeben. Hier reduziert sich der Stabilisierungsschritt auf eine einfache Matrix-Vektor-Multiplikation.

Entropische MRT-Modelle wurden bisher nur für die D2Q9- und D3Q27-Sterne definiert. Mit der Verwendung des D3Q19-Sterns bietet die PES hier eine effizientere Alternative. Die PES zeigt auch die enge Verwandtschaft der entropischen MRT-Modelle mit der Regularisierten LBM. Obwohl das Verfahren ähnlich wie KBC aus der Maximierung einer quadratischen Entropiefunktion gewonnen wurde, ist das Ergebnis äquivalent mit der Regularisierten LBM.

3.4.2. Zusammenfassung

In diesem Kapitel wurde eine neue Klasse von Kollisionsmodellen für die LBM entwickelt. Diese Modelle stabilisieren die Simulationen mit der LBM durch eine Optimierung der nicht-hydrodynamischen Momente. Während die Schermomente gemäß der vorgegebenen Viskosität relaxiert werden, wählt die PES die höheren Momente durch die Maximierung einer quadratischen Pseudo-Entropie.

Dieses Paradigma führte auf rechnerisch effiziente Stabilisierungsoperatoren, die im Gegensatz zur Entropischen LBM explizit und im Gegensatz zu MRT-Modellen und KBC unabhängig von der Basis der höheren Momente, der Gleichgewichtsverteilung

und der Relaxation der Schermomente sind. Somit lässt sich die PES flexibel mit anderen Kollisionsmodellen kombinieren. Für die unkritische Wahl $\tau = 1$ stimmt das Verfahren mit dem BGK-Modell überein.

Das Verfahren wurde mit zwei Testfällen konfrontiert. In allen Simulationen konnte die Stabilität im Vergleich mit dem BGK-Modell drastisch gesteigert werden. Besonders die Simulationen der zweidimensionalen Scherschicht sind für andere CFD-Methoden problematisch. In diesem Testfall zeigt der Vergleich mit anderen Publikationen die Überlegenheit der PES gegenüber KBC und der entropischen LBM. Insgesamt wies die PES in den hier gezeigten Simulationen alle wünschenswerten Eigenschaften eines effektiven Subgrid-Modells auf: Bei drastisch verringerter Auflösung blieben die Stabilität und die grobskaligen Strukturen erhalten, bei höherer Auflösung konvergierten die Simulationen zu vollaufgelösten Ergebnissen.

Besonders interessant ist die Äquivalenz der hier hergeleiteten Methode mit der Regularisierten LBM, welche dadurch um eine entropiegestützte Interpretation und um eine flexiblere Formulierung erweitert wird. Somit unterstützen die hier gerechneten Testfälle auch die guten Stabilitätseigenschaften der Regularisierten LBM und ihre Verwendung als effektives Subgrid-Modell.

Während Subgrid-Modelle in klassischen CFD-Methoden auf den makroskopischen Größen und deren Ableitungen aufbauen, verwendet der hier verfolgte Ansatz das kinetische Modell der LBM. In diesem Sinn liefern die lokalen Nichtgleichgewichtsanteile der Verteilungsfunktionen Informationen über die Subgrid-Skalen, die zur Stabilisierung der Simulationen herangezogen werden können. Konkret wird dazu das allgemeine thermodynamische Prinzip verwendet, dass physikalische Systeme zu ihrem wahrscheinlichsten Zustand streben, also zum Maximum der Entropie. Somit nutzt die PES ein im Kontext der CFD einzigartiges Merkmal des Lattice-Boltzmann-Ansatzes zur Konstruktion effizienter und stabiler Kollisionsmodelle.

4. Räumliche Diskretisierungen: Eulersche Methoden

Während im vorangehenden Kapitel verbesserte Kollisionsschritte untersucht wurden, befasst sich dieses Kapitel mit alternativen Formulierungen des Strömungsschritts. In Unterabschnitt 2.6.1 wurde gezeigt, dass die Ortsdiskretisierung in der Standard-LBM zweiter Ordnung ist. Im Gegensatz dazu werden besonders turbulente und aeroakustische Strömungen in klassischen Methoden oft mit Verfahren höherer Ordnung gerechnet, da dies eine drastische Vergrößerung des Gitters erlaubt [44, 69]. Deswegen soll untersucht werden, ob sich die LBM in effizienter Weise auf eine höhere Konvergenzordnung erweitern lässt.

Ein weiterer offensichtlicher Nachteil der Standard-LBM ist ihre Formulierung auf regulären kartesischen Gittern. Besonders in wandgebundenen Strömungen ist es von Vorteil, wenn das Gitter in wandnormaler Richtung feiner aufgelöst werden kann, da die wandnormalen Ableitungen der Strömung in wandnahen Bereichen um ein Vielfaches größer sind als die tangentialen Ableitungen. Die Verallgemeinerung auf allgemeine Gitter ist daher eine wünschenswerte Erweiterung der LBM.

Die Problematik bei dieser Art von LBMs, sogenannten Off-Lattice-Boltzmann-Methoden (OLBM), besteht in der effizienten Lösung des Strömungsschritts. Dieser ist in der Standard-LBM so formuliert, dass Verteilungsfunktionen von Gitterpunkt zu Gitterpunkt weitergeschoben werden. Das ist auf allgemeinen irregulären Gittern nicht mehr möglich.

Viele OLBMs lösen den advektiven Teil der diskreten BGK-Gleichung in einem Eulerschen Formalismus, was bedeutet, dass der Strömungsschritt auf einem fixierten Gitter durchgeführt wird. Der LBM-Strömungsschritt kann dagegen als Lagrange-sches Verfahren interpretiert werden, in dem das Gitter mit der Strömung bewegt wird. Wegen der Verwendung von regulären Gittern können die geströmten Gitter in der Standard-LBM jedoch mit dem Gitter vor der Strömung identifiziert werden.

Im Folgenden wird zunächst ein Überblick über die Entwicklungen im Bereich von OLBMs gegeben. Dabei wird ein Verfahren besonders betrachtet, die

Spektralelement-Discontinuous-Galerkin (SEDG)-LBM [161]. Diese wird um eine dreidimensionale Formulierung und um implizite Zeitintegrationsverfahren erweitert und auf drei verschiedene Probleme angewandt. Im Zentrum steht dabei der Vergleich mit der Standard-LBM in Bezug auf Genauigkeit und Rechenzeit.

Die Untersuchung der SEDG-LBM motiviert auch die Entwicklung einer neuartigen Semi-Lagrangeschen Lattice-Boltzmann-Methode (SLLBM) in Kapitel 5. Diese kombiniert Elemente der Eulerschen LBMs mit der (Lagrangeschen) Standard-LBM. Es wird gezeigt, weshalb dieser Ansatz eine effizientere Alternative zu aktuellen OLBMs darstellt.

4.1. Stand der Technik

Dieser Abschnitt gibt einen Überblick über die existierenden Lattice-Boltzmann-Methoden auf irregulären Gittern.

4.1.1. Native LBMs zur Gitterverfeinerung

Neben OLBMs, welche die Diskretisierung des Geschwindigkeitsraums vom Rechengitter entkoppeln, existieren auch sogenannte *native LBMs* mit einem exakten Strömungsschritt, die auf regulären, lokal verfeinerten Gittern operieren [53, 66, 134, 189]. Diese Verfahren teilen das Strömungsgebiet in Regionen unterschiedlicher Gitterauflösung ein. Dabei wird in verfeinerten Regionen mit einem entsprechend skalierten Zeitschritt gearbeitet. Beim Gittertransfer müssen die Nicht-Gleichgewichtsanteile der transformierten Verteilungsfunktionen skaliert werden [53, 66].

Statt hierarchisch verfeinerter Gitter kann auch eine Gebietszerlegung verwendet werden, wobei Überlapp-Bereiche zwischen verschiedenen aufgelösten Regionen definiert werden, in denen die relevanten Strömungsgrößen von einem auf das andere Gitter übersetzt werden. Diese Art der Gitterverfeinerung ist in vielen Lattice-Boltzmann-Codes implementiert [51, 88, 134] und kann zu einer wesentlichen Effizienzsteigerung gegenüber Simulationen auf äquidistanten Gittern führen [74, 148].

Auch solche hierarchischen Verfeinerungen erfordern jedoch quadratische oder kubische Zellen, während eine anisotrope Verfeinerung für die meisten Strömungen vorteilhaft ist [55, 123, 227]. Insbesondere in wandgebundenen Strömungen sind die Geschwindigkeitsgradienten in wandnormaler Richtung um ein Vielfaches steiler als in die tangentialen Richtungen [131], weshalb eine effiziente Diskretisierung durch eine graduelle anisotrope Verfeinerung der Zellen in Wandnähe erreicht werden kann. Hierarchischen Verfeinerungen auf quadratischen und kubischen Zellen schließen eine solche effiziente Vernetzung von vornherein aus.

Einige Arbeiten betrachten auch die Möglichkeit, auf allgemeineren Gittern nati-

ve LBMs zu definieren [19, 121, 180]. Dies ist auf regulären Rechteckgittern in Form spezieller MRT-Modelle möglich [19, 180, 221]. Die ersten Modelle dieser Art [19, 238, 239] verletzen die Isotropie der Viskosität [243] und waren nicht konsistent mit den Navier-Stokes-Gleichungen [33]. Während in [94] zusätzliche diskrete Partikelgeschwindigkeiten eingesetzt wurden, um diese Anisotropie der Viskosität zu korrigieren, konnte dies in [180] erreicht werden, indem Momente zweiter Ordnung in der Gleichgewichtsverteilung verwendet wurden. In [221] wurde dieser Ansatz auf dreidimensionale Gitter übertragen und erfolgreich zur Simulation einer turbulenten Kanalströmung verwendet. Diese neuen Entwicklungen von anisotropen LBMs sind sehr vielversprechend, lassen sich aber nicht ohne Weiteres auf allgemeinere Gitter übertragen.

Der Versuch einer allgemeineren Formulierung [121] erfordert einige Kompromisse, verwendet einen größeren Satz diskreter Geschwindigkeiten, verlangt die Lösung von Gleichungssystemen an jedem Gitterpunkt und erhält den Impuls nur im Limit kleiner Abweichungen vom Gleichgewicht. Die praktische Relevanz solcher allgemeinen Ansätze ist daher im Moment nicht absehbar.

4.1.2. Off-Lattice-Randbedingungen

Bei allen LBMs auf regulären Gittern stellt sich darüber hinaus das Problem, Randbedingungen zu formulieren, die auch dann genaue Lösungen liefern, wenn Ränder nicht entlang der Gitterachsen verlaufen. Dazu werden bei nativen LBMs zum Beispiel *Immersed-Boundary-Methoden* [183] verwendet [50, 64, 172, 230, 236]. Ein Gitter, das sich der jeweiligen Geometrie anpasst, liefert jedoch im Allgemeinen bessere Lösungen, da die relevanten Strömungsgrößen direkt auf den Randpunkten vorgegeben werden können.

4.1.3. Finite-Volumen-LBMs

Wegen der offensichtlichen Nachteile regulärer Gitter wurde schon in den ersten Jahren der LBM ein erstes Finite-Volumen-Verfahren für die diskrete Boltzmann-Gleichung (2.5) entwickelt [171]. Dieses Verfahren und seine frühen Erweiterungen [181, 182, 231, 232, 237] übergehen jedoch einen zentralen Aspekt der LBM, nämlich den impliziten Charakter des Kollisionsschritts. Dieser geht auf [186] zurück und wurde in dieser Arbeit in Unterabschnitt 2.4.4 beschrieben. Da der Kollisionsschritt in den ersten Finite-Volumen-Verfahren explizit behandelt wurde, ergibt sich für diese Verfahren eine viskositätsabhängige Einschränkung des Zeitschritts $\delta_t < 2\tau$ [216], was die Methoden für die effiziente Simulation zum Beispiel turbulenter Strömungen disqualifiziert.

Die implizite Diskretisierung des Kollisionsschritts wurde erst später für Eulersche OLBMs formuliert [10, 81]. Wie in der Standard-LBM führt dies zu einer Aufteilung in

Kollisions- und Strömungsschritt. Der Strömungsschritt ergibt sich in diesem Kontext als Zeitschritt eines Finite-Volumen-Verfahrens für die lineare Advektionsgleichung.

$$\frac{\partial f_i}{\partial t} + e_{(i)j} \frac{\partial f_{(i)}}{\partial x_j} = 0. \quad (4.1)$$

Die verschiedenen Finite-Volumen-LBMs unterscheiden sich vor allem in der Wahl der konkreten Finite-Volumen-Verfahren zur Lösung des advektiven Teils [10, 144, 178, 179, 216]. Während viele Finite-Volumen-LBMs nur für einfache zweidimensionale Strömungen validiert wurden, wurde in [164] eine dreidimensionale Strömung durch ein poröses Medium simuliert. Hierbei wurden insbesondere die Vorteile unstrukturierter Gitter in komplexen Geometrien hervorgehoben. In [83] wurde eine Finite-Volumen-LBM zur Simulation einer zweidimensionalen turbulenten Kanalströmung genutzt, wobei die Simulation mit der Finite-Volumen-LBM trotz einer verbesserten Auflösung des Randes langsamer war als eine Simulation mit der Standard-LBM.

Eine neuere Entwicklung im Bereich der Finite-Volumen-LBMs ist das *Discrete Unified Gas-Kinetic Scheme* (DUGKS) [80]. Im Gegensatz zu anderen Verfahren führt DUGKS zusätzlich zur Transformation (2.7) eine zweite Transformation der Verteilungsfunktionen ein [222], um auch implizite zentrale Flüsse über Zellgrenzen explizit zu formulieren. Darüber hinaus wurde demonstriert, dass das Verfahren mit einem größeren Satz diskreter Teilchengeschwindigkeiten eine realitätsnahe Approximation von Nicht-Kontinuums-Strömungen erlaubt. In [241] wurde DUGKS mit der Finite-Volumen-LBM von Bardow et al. [10] verglichen und gezeigt, dass die zusätzliche Transformation der Verteilungsfunktionen die Genauigkeit und Stabilität gegenüber [10] verbessert, aber den Rechenaufwand pro Zelle verdoppelt. In [223] wird DUGKS für zweidimensionale Strömungen mit der Standard-LBM verglichen. Dabei überstiegen die Kosten pro Zeitschritt die der Standard-LBM um ein Fünffaches, wobei aber keine Details zur Implementierung der Standard-LBM genannt wurden. In [14] wurde DUGKS zur Simulation eines dreidimensionalen Taylor-Green-Vortex bei einer Reynoldszahl von 300 verwendet. Dabei war das Verfahren deutlich dissipativer als die Standard-LBM. Auf diese Simulationen wird in Abschnitt 5.4.6 noch näher eingegangen. Bei der Simulation einer turbulenten Kanalströmung [14] wurden die Statistiken zwar ähnlich gut wiedergegeben wie in einer feiner aufgelösten Standard-LBM-Simulation, allerdings war der Zeitschritt in der DUGKS-Simulation um eine Größenordnung kleiner, um die numerische Dissipation in Grenzen zu halten.

4.1.4. Finite-Differenzen LBMs

Zum ersten Mal wurde eine Anwendung von Finite-Differenzen-Verfahren auf die LBM in [26] diskutiert und getestet. Hier wurde der Kollisionsterm, wie auch in den ersten Finite-Volumen-Verfahren, explizit diskretisiert, was schon bei moderaten Reynoldszahlen leicht zu einem instabilen Verfahren führt. Darüber hinaus wurde

auch eine implizite Diskretisierung des Kollisionsterms diskutiert, aber erst in [160] umgesetzt. Eine Chapman-Enskog-Analyse zeigt, dass dabei die Art der Diskretisierung die Viskosität verändert [203]. In [81] wurden in diesem Verfahren schließlich die transformierten Verteilungsfunktionen genutzt, um die implizite Diskretisierung des Kollisionsoperators in explizite Form zu bringen.

In [96, 215, 233, 234] wurden Finite-Differenzen-LBMs auf Simulationen kompressibler Strömungen erweitert und sogar für Simulationen aeroakustischer Probleme und kompressibler Mehrphasenströmungen verwendet. In kompressiblen Strömungen haben OLBMs eine besondere Relevanz, da sie allgemeinere Diskretisierungen des Geschwindigkeitsraums erlauben. Die dadurch ermöglichte Entkopplung der Momente erlaubt die unabhängige Wahl der Prandtl-Zahl und des Isentropenexponenten [78, 122], was auf regulären Gittern nur mit großen Geschwindigkeitssternen möglich ist [68].

In [95, 99] wurden kompakte Finite-Differenzen-Verfahren hoher Ordnung in der LBM genutzt, um Strömungen um zweidimensionale Tragflächenprofile zu simulieren. Dabei wurde gezeigt, dass die räumlichen Diskretisierungsfehler mit der Ordnung des Finite-Differenzen-Verfahrens konvergieren. Ähnliche Ergebnisse wurden auch mit einer *Chebyshev-Collocation*-Methode erzielt [97, 98], wobei hier die Anzahl der Gitterpunkte und damit der Rechenaufwand gegenüber [95] verringert werden konnte. All diese Verfahren hoher Ordnung verwendeten dabei explizite Runge-Kutta-Verfahren zur Zeitintegration, was kleine Zeitschritte erfordert.

In [169] wurde mithilfe eines gitterlosen Petrov-Galerkin-Ansatzes eine Finite-Differenzen-LBM mit einer unstrukturierten Verteilung von Gitterpunkten definiert. Es wurde gezeigt, dass diese Art der Diskretisierung bei der Simulation von Strömungen durch poröse Medien vorteilhaft ist. Da jedoch an jedem Punkt eine Interpolation mit radialen Basisfunktionen erforderlich ist, kann ein solcher Ansatz nur in speziellen Strömungen mit sehr komplexen Geometrien effizient sein.

In [59] wurde schließlich eine Finite-Differenzen-LBM mit einer hierarchisch verfeinerten Standard-LBM zur Umsetzung von adaptiven Gitterverfeinerungen verglichen. Das hierarchisch verfeinerte Rechengitter und das dort verwendete Lax-Wendroff-Verfahren waren so definiert, dass die Finite-Differenzen-LBM auf dem feinsten Gitter der Standard-LBM entsprach. Während die nativ verfeinerte LBM für eine zweidimensionale Zylinderumströmung 10% schneller war, hatte die Finite-Differenzen-LBM Vorteile in der Stabilität. Dieses Ergebnis zeigt, dass die Finite-Differenzen-LBM mit der Standard-LBM effizient zu einem hybriden Verfahren kombiniert werden kann, wie es für die Finite-Volumen-LBM auch in [48] gezeigt wird.

4.1.5. Finite-Elemente-LBMs

Statt mit Finite-Volumen- oder Finite-Differenzen-Verfahren lässt sich die diskrete Boltzmann-Gleichung auch mit Finite-Elemente-Verfahren lösen. In [141, 142] wur-

den dabei jeweils implizite und explizite Diskretisierungen des Kollisionsschritts und des Advektionsschritts untersucht. Dabei wurde die Transformation der Verteilungsfunktionen (2.7) nicht berücksichtigt. In [224] wurde diese Finite-Elemente-LBM zur Simulation einer dreidimensionalen Wirbelströmung mit freien Oberflächen in einem konzentrischen Zylinder verwendet. In [54] wurde gezeigt, dass sich mit Finite-Elemente-Methoden exponentielle Konvergenz der räumlichen Fehler erzielen lässt und dass dazu sehr kleine Zeitschritte notwendig sind. In den Weiterentwicklungen der Finite-Elemente-LBM wurde die diskrete Boltzmann-Gleichung mit einem Kleinste-Quadrate-Ansatz gelöst, wodurch die Anzahl an Gitterpunkten zur Simulation eines porösen Mediums gegenüber der Standard-LBM um Größenordnungen verkleinert werden konnte [145–147].

Neben den betrachteten OLBMs, die auf Finite-Volumen-, Finite-Differenzen- und Finite-Elemente-Verfahren aufbauen, gibt es auch OLBMs, die die Advektion mit Discontinuous-Galerkin-Verfahren durchführen. Da diese im weiteren Verlauf der Arbeit eine größere Rolle spielen, werden sie im Folgenden in einem separaten Abschnitt behandelt.

4.2. Discontinuous-Galerkin-Lattice-Boltzmann-Verfahren

Discontinuous-Galerkin-Verfahren kombinieren Ideen der Finite-Elemente- und Finite-Volumen-Methoden [101]. Das Strömungsgebiet wird durch ein Gitter diskretisiert. Innerhalb der Gitterzellen wird die Lösung durch einen Finite-Elemente-Ansatz approximiert, die Kommunikation zwischen den Zellen wird jedoch im Stile von Finite-Volumen-Methoden über Flüsse realisiert. Dadurch erlauben Discontinuous-Galerkin-Verfahren unstetige Lösungen und eine hohe Konvergenzordnung. Beides ermöglicht eine Reduktion dissipativer und dispersiver numerischer Effekte, sowie eine effiziente Parallelisierung. Im Folgenden werden Discontinuous-Galerkin-Methoden für die Advektionsgleichung auf quadrilateralen Gittern formuliert, um den Strömungsschritt der LBM auf irregulären Gitter durchzuführen. Die Ortsdiskretisierung findet sich in gleicher Form im Zweidimensionalen in [161, 217]. Diese SEDG-LBM wurde auf schwach kompressible Strömungen [161, 217], natürliche Konvektion [177], Wärmetransfer [175] und Mehrphasenströmungen [176] angewandt und auf GPUs implementiert [158]. Ähnliche Ansätze wurden auch in [195, 202, 235] verfolgt, wobei dort andere Discontinuous-Galerkin-Verfahren zur Advektion verwendet wurden.

In dieser Arbeit wird die SEDG-LBM von Min und Lee [161] zunächst mit der Standard-LBM verglichen, wobei für die Simulationen mit der Standard-LBM der Open-Source-Code Palabos [2] verwendet wird. In diesem Rahmen wird die SEDG-Methode um implizite Zeitintegration erweitert, um die CFL-Bedingung des Advektionslösers zu umgehen. Dieses wurde für Finite-Elemente-LBMs und Finite-Volumen-

LBM in ähnlicher Weise zum Beispiel in [109, 110, 141, 144] verfolgt, ist aber im Rahmen von Discontinuous-Galerkin-LBMs neu. Zum Abschluss des Kapitels wird die SEDG-LBM auf dreidimensionale Strömungen erweitert und zur Simulation einer turbulenten Kanalströmung verwendet.

Die Untersuchung der Vorteile und Nachteile der SEDG-LBM motiviert in dieser Arbeit vor allem die Verwendung Semi-Lagrangescher statt Eulerscher Verfahren. Im weiteren Verlauf der Arbeit wird gezeigt, dass mit einer Semi-Lagrangeschen LBM die wesentlichen Nachteile der SEDG-LBM behoben werden können.

4.2.1. Ortsdiskretisierung

Die folgende Formulierung der Ortsdiskretisierung folgt [101, 161]. Die Advektionsgleichung (4.1) hat auf jeder Zelle D_c eine äquivalente schwache Formulierung

$$\left\langle \psi, \frac{\partial f_i}{\partial t} + e_{(i)j} \frac{\partial f_{(i)}}{\partial x_j} \right\rangle_{D_c} = 0.$$

Hierbei bezeichnet ψ eine Testfunktion und

$$\langle \psi_1, \psi_2 \rangle_{D_c} := \int_{D_c} \psi_1 \psi_2 \, d\mathbb{R}^{\dim}$$

das Skalarprodukt. Die Flüsse über die Zellgrenzen werden im Stil eines Finite-Volumen-Verfahrens realisiert. Dazu sei \mathbf{n} der nach außen gerichtete Normaleneinheitsvektor am jeweiligen Randpunkt. Durch eine zweimalige Anwendung des Satzes von Gauß erhält man

$$\frac{\partial}{\partial t} \langle \psi, f_i \rangle_{D_c} = -e_{(i)j} \left\langle \psi, \frac{\partial f_{(i)}}{\partial x_j} \right\rangle_{D_c} + \langle \psi, n_j e_{(i)j} f_{(i)} \rangle_{\partial D_c} - \langle \psi, n_j e_{(i)j} f_{(i)} \rangle_{\partial D_c},$$

wobei nun die Flüsse über die Verteilungsfunktionen der Nachbarzellen berechnet werden können.

In [161] wurde gezeigt, dass die resultierende gewöhnliche Differentialgleichung für zentrale Flüsse instabil ist und daher *Upwind*-Flüsse verwendet werden sollten.

Dazu wird der Rand der Zelle in zwei Teile aufgeteilt. Über $\partial_i^+ D_c := \{x \in \partial D_c : e_{ij} n_j < 0\}$ fließt die Verteilungsfunktion f_i in die Zelle und über $\partial_i^- D_c := \{x \in \partial D_c : e_{ij} n_j \geq 0\}$ fließt f_i aus der Zelle hinaus. Die Upwind-Flüsse werden über die einfließenden Verteilungsfunktionen berechnet:

$$\frac{\partial}{\partial t} \langle \psi, f_i \rangle_{D_c} = -e_{(i)j} \left\langle \psi, \frac{\partial f_{(i)}}{\partial x_j} \right\rangle_{D_c} + \langle \psi, n_j e_{(i)j} f_{(i)} \rangle_{\partial_i^+ D_c} - \langle \psi, n_j e_{(i)j} f_{(i)} \rangle_{\partial_i^- D_c}.$$

Im letzten Term wird nun f_i durch die Verteilungsfunktionen f_i^+ der Nachbarzellen ersetzt.

Dies ergibt

$$\frac{\partial}{\partial t} \langle \psi, f_i \rangle_{D_c} = -e_{(i)j} \left\langle \psi, \frac{\partial f^{(i)}}{\partial x_j} \right\rangle_{D_c} + \left\langle \psi, n_j e_{(i)j} (f^{(i)} - f_{(i)}^+) \right\rangle_{\partial_{(i)}^+ D_c}.$$

Nun werden die Verteilungsfunktionen auf jeder Zelle D_c durch einen Finite-Elemente-Ansatz diskretisiert

$$f_i = \hat{f}_{i(c)k} \psi_{(c)k},$$

wobei $\psi_{(c)k}$ die Ansatzfunktionen und $\hat{f}_{i(c)k}$ die zugehörigen Freiheitsgrade bezeichnet.

Setzt man die Ansatzfunktionen als Testfunktionen ein, so ergibt sich

$$\begin{aligned} \langle \psi_{(c)l}, \psi_{(c)k} \rangle_{D_c} \frac{\partial \hat{f}_{i(c)k}}{\partial t} &= - \left\langle \psi_{(c)l}, \frac{\partial \psi_{(c)k}}{\partial x_j} \right\rangle_{D_c} e_{(i)j} \hat{f}_{i(c)k} \\ &+ n_j e_{(i)j} \left(\hat{f}_{i(c)k} \langle \psi_{(c)l}, \psi_{(c)k} \rangle_{\partial_i^+ D_c} - \hat{f}_{i(c+)k} \langle \psi_{(c)l}, \psi_{(c+)k} \rangle_{\partial_i^+ D_c} \right). \end{aligned} \quad (4.2)$$

Hierbei bezeichnet $c+$ den Index der jeweiligen Nachbarzellen.

Nach der Wahl einer geeigneten Basis von Finiten Elementen bildet Gleichung (4.2) ein System linearer gewöhnlicher Differentialgleichungen. Dazu werden hier Lagrange-sche Polynome verwendet, da sich dadurch die Freiheitsgrade mit den Funktionswerten an den Stützstellen identifizieren lassen. Diese Identifikation bietet einen Vorteil im Hinblick auf den Kollisionsschritt: Dieser lässt sich lokal auf den Stützstellen durchführen. Die Stützstellen können außerdem so gewählt werden, dass die Ansatzfunktionen eine Orthonormalbasis bilden. Dadurch wird die Massenmatrix $\langle \psi_{(c)l}, \psi_{(c)k} \rangle_{D_c}$ diagonal und es ergibt sich ein explizites Differentialgleichungssystem.

Konkret wird dazu für jede Zelle eine bijektive, multilineare Mapping-Funktion auf die Einheitszelle $\mathcal{M}_c : D_c \rightarrow [0, 1]^{\dim}$ definiert. Auf der Einheitszelle wird eine Basis von Lagrange-Polynomen $\tilde{\psi}_k$ definiert, deren Stützstellen die Gauß-Lobatto-Legendre-Punkte \tilde{x}_k sind [101]. Die so definierten Ansatzfunktionen haben folgende besondere Eigenschaften:

Lemma 1 *Seien \tilde{x}_k , $k = 1, \dots, N + 1$, die Gauß-Lobatto-Legendre-Punkte N -ter Ordnung auf $[0, 1]$. Dann gilt für die Lagrange-Polynome*

$$\tilde{\psi}_k(\tilde{x}) := \prod_{i \neq k} \frac{\tilde{x} - \tilde{x}_i}{\tilde{x}_k - \tilde{x}_i}$$

1. $\tilde{\psi}_k(\tilde{x}_l) = \delta_{kl}$,
2. $\langle \tilde{\psi}_k, \tilde{\psi}_l \rangle_{[0,1]} = \delta_{kl}$.

BEWEIS siehe [101].

□

Diese Eigenschaften übertragen sich über Produktbildung auch auf höherdimensionale Räume. Mit der oben definierten Mapping-Funktion können darüber die Ansatzfunktionen und Stützstellen auf beliebigen quadrilateralen Zellen als $\psi_{ck} := \tilde{\psi}_k \circ \mathcal{M}_c$ und $x_{ck} := \mathcal{M}_c^{-1}(\tilde{x}_k)$ definiert werden. Auf der Einheitszelle können die Integrale mit einer numerischen Quadratur berechnet werden. Die Ansatzfunktionen haben dabei den Grad N . Die Gauß-Quadratur gleicher Ordnung auf den Gauß-Lobatto-Legendre-Punkten mit Gewichten \tilde{w}_i ist exakt für die Ansatzfunktionen. Mit der Transformationsregel lassen sich die Elemente der Massenmatrix und Steifigkeitsmatrix berechnen als

$$\begin{aligned} \hat{m}_{clk} &= \langle \psi_{(c)l}, \psi_{(c)k} \rangle_{D_c} = \int_{[0,1]^{\dim}} \tilde{\psi}_l \tilde{\psi}_k \det \left(\frac{\partial \mathcal{M}_c^{-1}}{\partial \mathbf{x}} \right) d\mathbb{R}^{\dim} \\ &= \sum_{i=1}^{(N+1)^{\dim}} \tilde{w}_{(i)} \tilde{\psi}_l(\tilde{\mathbf{x}}_{(i)}) \tilde{\psi}_k(\tilde{\mathbf{x}}_{(i)}) \det \left(\frac{\partial \mathcal{M}_c^{-1}}{\partial \mathbf{x}}(\tilde{\mathbf{x}}_{(i)}) \right) \\ &= \tilde{w}_{(l)} \det \left(\frac{\partial \mathcal{M}_c^{-1}}{\partial \mathbf{x}}(\tilde{\mathbf{x}}_{(l)}) \right) \delta_{(l)k}, \\ \hat{s}_{jclk} &= \left\langle \psi_{(c)l}, \frac{\partial \psi_{(c)k}}{\partial x_j} \right\rangle_{D_c} = \int_{[0,1]^{\dim}} \tilde{\psi}_l \frac{\partial \tilde{\psi}_k}{\partial x_j} \det \left(\frac{\partial \mathcal{M}_c^{-1}}{\partial \mathbf{x}} \right) d\mathbb{R}^{\dim} \\ &= \sum_{i=1}^{(N+1)^{\dim}} \tilde{w}_{(i)} \tilde{\psi}_l(\tilde{\mathbf{x}}_{(i)}) \frac{\partial \tilde{\psi}_k}{\partial x_j}(\tilde{\mathbf{x}}_{(i)}) \det \left(\frac{\partial \mathcal{M}_c^{-1}}{\partial \mathbf{x}}(\tilde{\mathbf{x}}_{(i)}) \right) \\ &= \tilde{w}_{(l)} \frac{\partial \tilde{\psi}_k}{\partial x_j}(\tilde{\mathbf{x}}_{(l)}) \det \left(\frac{\partial \mathcal{M}_c^{-1}}{\partial \mathbf{x}}(\tilde{\mathbf{x}}_{(l)}) \right). \end{aligned}$$

Für die Randintegrale wird in gleicher Weise eine Quadratur der Dimension $\dim - 1$ verwendet. Da die Mapping-Funktionen so definiert sind, dass die Stützpunkte benachbarter Zellen aufeinanderfallen, gilt

$$\begin{aligned} \hat{r}_{(i)clk} &= \langle \psi_{(c)l}, \psi_{(c)k} \rangle_{\partial_i^+ D_c} = \langle \psi_{(c)l}, \psi_{(c+)k} \rangle_{\partial_i^+ D_c} \\ &= \tilde{w}_{(l)} \det \left(\frac{\partial \mathcal{M}_c^{-1}}{\partial \mathbf{x}}(\tilde{\mathbf{x}}_{(l)}) \right) \delta_{(l)k} \cdot \chi(n_{(i)j} e_{(i)j} < 0). \end{aligned}$$

Hierbei bezeichnet χ die Indikatorfunktion. Mit den so berechneten Matrizen kann Gleichung (4.2) umformuliert werden zu

$$\hat{m}_{(c)lk} \frac{d\hat{f}_{i(c)k}}{dt} = -e_{(i)j} \hat{s}_{j(c)lk} \hat{f}_{i(c)k} + n_j e_{(i)j} \hat{r}_{(c)lk} \left(\hat{f}_{(i)(c)k} - \hat{f}_{(i)(c+)k} \right),$$

oder in Matrixform

$$\frac{d}{dt}\hat{f}_i = -e_{(i)j}\hat{M}^{-1}\hat{S}_j\hat{f}_{(i)} + n_j e_{(i)j}\hat{M}^{-1}\hat{R}_{(i)}\left(\hat{f}_{(i)} - \hat{f}_{(i)}^+\right). \quad (4.3)$$

Insgesamt bezeichnet Gleichung (4.3) eine lineare gewöhnliche Differentialgleichung der Form

$$\hat{f} = L\hat{f} + \hat{b},$$

wobei der konstante Term \hat{b} durch die Randbedingungen zustande kommt (siehe Unterabschnitt 4.2.3).

4.2.2. Zeitdiskretisierung

Die gewöhnliche explizite Differentialgleichung (4.3) kann mit Zeitintegratoren, beispielsweise Runge-Kutta-Verfahren, gelöst werden. Der Lattice-Boltzmann-Strömungsschritt entspricht dann einem Schritt des Zeitintegrators. In [161] und [217] wurden dazu jeweils ein klassisches explizites Runge-Kutta-Verfahren vierter Ordnung und ein Exponentialintegrator verwendet. Da es sich bei beiden Verfahren um explizite Integratoren handelt, bleiben beide nur bei kleinen CFL-Zahlen stabil.

Die CFL-Zahl lässt sich für das hier betrachtete Discontinuous-Galerkin-Verfahren definieren als

$$CFL = \frac{\delta_t N^2 \max |e_i|}{\min \delta_x},$$

wobei $\min \delta_x$ den minimalen Abstand zwischen zwei Gitterknoten bezeichnet. Die quadratische Abhängigkeit von der Ordnung der Ansatzfunktionen ist für moderate Ordnungen ($N \ll 100$) damit zu begründen, dass die Kondition der Steifigkeitsmatrix in diesem Bereich quadratisch mit der Ordnung ansteigt [101]. Aus diesem Grund wird in dieser Arbeit diese Definition der CFL-Zahl herangezogen, statt der Standarddefinition, die auf den Faktor N^2 verzichtet. Explizite Integratoren besitzen eine kritische CFL-Zahl, jenseits der die Simulationen instabil werden.

In dieser Arbeit wird die Discontinuous-Galerkin-Diskretisierung des Strömungsschritts um weitere Zeitintegratoren erweitert. Insbesondere werden implizite Integratoren verwendet, um die kritische CFL-Bedingung zu umgehen. Dazu wurden verschiedene implizite Integratoren getestet und implementiert. In den hier beschriebenen Simulationen wird nur die Theta-Methode mit $\theta = 0.5$ verwendet. Diese löst in jedem Zeitschritt das Gleichungssystem

$$(I - \theta\delta_t L)\hat{f}(t + \delta_t) = (I + \theta\delta_t L)\hat{f}(t) + \hat{b},$$

wozu in dieser Arbeit ein BiCGSTAB-Löser mit einer Toleranz von $10^{-6} \cdot \|\hat{f}\|_2$ verwendet wird.

4.2.3. Randbedingung

Die Randbedingungen können in der SEDG-LBM über die Flüsse definiert werden [161]. An Festkörperändern wird dazu die *No-Slip*-Randbedingung aus [133] verwendet. Hierbei werden die Nichtgleichgewichtsverteilungen an der Wand in entgegengesetzter Richtung reflektiert: $f_i^{\text{neq}} = f_j^{\text{neq}}$, wobei j die Richtung mit $\mathbf{e}_i = -\mathbf{e}_j$ bezeichnet.

$$\begin{aligned} f_i(t, \mathbf{x}) &= f_i^{\text{eq}}(\rho, \mathbf{u}) + f_i^{\text{neq}}(t, \mathbf{x}) = f_i^{\text{eq}}(\rho, \mathbf{u}) + f_j^{\text{neq}}(t, \mathbf{x}) \\ &= f_j(t, \mathbf{x}) + f_i^{\text{eq}}(\rho, \mathbf{u}) - f_j^{\text{eq}}(\rho, \mathbf{u}). \end{aligned}$$

Bei einer vorgegebenen Wandgeschwindigkeit \mathbf{u}_w heben sich die meisten Terme der beiden Gleichgewichtsverteilungen auf und die Flüsse ergeben sich als

$$f_i^+(t, \mathbf{x}) = f_j(t, \mathbf{x}) + 2\omega_{(i)}\rho_0(\mathbf{e}_{(i)} \cdot \mathbf{u}_w)/c_s^2. \quad (4.4)$$

Die Dichte am Rand wird wie in [161] durch die konstante Dichte ρ_0 ersetzt. Diese Approximation vermeidet eine Kopplung aller Verteilungsfunktionen in der Randbedingung. So bleiben nur entgegengesetzte Verteilungsfunktionen f_i und f_j mit $\mathbf{e}_i = -\mathbf{e}_j$ gekoppelt.

4.3. Simulationen mit Discontinuous-Galerkin-Lattice-Boltzmann-Verfahren

In diesem Abschnitt werden die SEDG-LBM [161] und die oben beschriebenen Erweiterungen getestet. Gegenüber der Standard-LBM bieten diese Verfahren verbesserte Konvergenzeigenschaften und ermöglichen zudem die Simulation auf unstrukturierten Gittern. Diese Stärken werden zunächst an zwei Beispielströmungen untersucht. Vergleichsrechnungen mit der Standard-LBM werden dazu in Palabos [2], einem offenen LBM-Code, durchgeführt. Zum Abschluss wird eine minimale turbulente Kanalströmung simuliert, um die Nachteile des Verfahrens bei der Simulation von turbulenten Strömungen deutlich zu machen und damit die Semi-Lagrangische Methode zu motivieren.

4.3.1. Instationäre Couette-Strömung

Um die Konvergenzeigenschaften der SEDG-LBM herauszuheben, wurde eine zweidimensionale instationäre Couette-Strömung simuliert. Dazu wurden die Verteilungen im gesamten Gebiet $[0, 1] \times [0, 1]$ durch eine Gleichgewichtsverteilung mit Geschwindigkeit $\mathbf{u} = 0$ (und Dichte $\rho = 1$) initialisiert. In x -Richtung wurden periodische Randbedingungen gesetzt. An der oberen und unteren Wand wurden mit der Randbedingung (4.4) jeweils die Geschwindigkeiten $\mathbf{u}(y = 0) = 0$ und $u_1(y = 1) = u_0 = 1$,

$u_2(y = 1) = 0$ gesetzt. Die Reynoldszahl und Machzahl wurden jeweils auf 2000 und 0.05 gesetzt.

Für diese Strömung existiert eine zeitabhängige analytische Lösung [161]

$$u_1(y, t) = u_0 \frac{y}{l_0} + \sum_{m=1}^{\infty} \frac{2u_0(-1)^m}{\lambda_m l_0} \exp(-\nu \lambda_m^2 t) \sin(\lambda_m y), \quad u_2 = 0$$

mit $l_0 = 1$ und $\lambda_m = \frac{m\pi}{l_0}$, $m \in \mathbb{N}$. Die numerisch ermittelten Geschwindigkeiten wurden zum Zeitpunkt $t_{\max} = 40$ mit der analytischen Lösung verglichen.

Die Ordnung der Ansatzfunktionen $N = 1, \dots, 9$ und die (äquidistante) Gitterverfeinerung $\ell = 1, 2, 3$ wurden variiert, wodurch sich jeweils 4^ℓ Zellen à $(N + 1)^2$ Gitterpunkte ergaben. Die Gesamtzahl an Gitterpunkten wurde dabei als $4^\ell \cdot (N + 1)^2$ definiert, sodass die Randpunkte der Zellen mehrfach gezählt werden. Dies ist im SEDG-Kontext sinnvoll, da jeder dieser Gitterpunkte auch eigene Freiheitsgrade besitzt.

Wie in der Originalpublikation [161] wurde als Zeitintegrator ein klassisches Runge-Kutta-Verfahren vierter Ordnung genutzt. Die CFL-Zahl wurde auf $CFL = 0.4$ gesetzt. In beiden Codes wurde das BGK-Kollisionsmodell mit D2Q9-Stern verwendet.

Die Vergleichsrechnungen in Palabos wurden auf Gittern mit 4^ℓ Punkten durchgeführt, wobei hier $\ell = 2, \dots, 9$ war. Hierbei wurden drei verschiedene Randbedingungen getestet: die lokale Randbedingung von Zou und He [244], die Interpolationsrandbedingung von Skordos [201] und die regularisierte Randbedingung von Latt et al. [140]. Die Simulationen liefen unparallelisiert auf einem Kern eines AMD Opteron (Shanghai, Familie 10h; Modell 8378) mit 2.4 GHz, wobei 16 GB DDR2 RAM zur Verfügung standen.

Abbildung 4.1 zeigt die numerischen Fehler in Abhängigkeit der Anzahl an Gitterpunkten und in Abhängigkeit der Laufzeit. Der Vergleich zeigt deutlich die höhere Konvergenzordnung der SEDG-LBM. Unter einem Fehler von ca. 10^{-2} fiel dieser exponentiell ab. Im Gegensatz dazu erzielte Palabos für die Randbedingung von Skordos [201] und die regularisierte Randbedingung [140] jeweils eine Konvergenzordnung von ungefähr 1.25. Für die Randbedingung von Zou und He [244] ist das Konvergenzverhalten sogar etwas schlechter.

Die defizitäre Konvergenz der Standard-LBM ist zum einen dadurch zu erklären, dass sich der zeitliche Fehler in der Standard-LBM nicht unabhängig steuern lässt, da die Zeitschrittweite durch Machzahl und Gitterauflösung vorgegeben wird. Zum anderen erzeugt die Unstetigkeit der initialen Lösung an $y = 1$ nicht-physikalische Oszillationen, die durch den exakten Strömungsschritt nur langsam dissipiert werden. Die asymptotische Analyse der Standard-LBM setzt hier eine differenzierbare Lösung voraus. Im Gegensatz dazu lässt das Discontinuous-Galerkin-Verfahren Unstetigkeiten an den Zellgrenzen zu und ermöglicht somit schon auf groben Gittern eine genaue Simulation der Couette-Strömung.

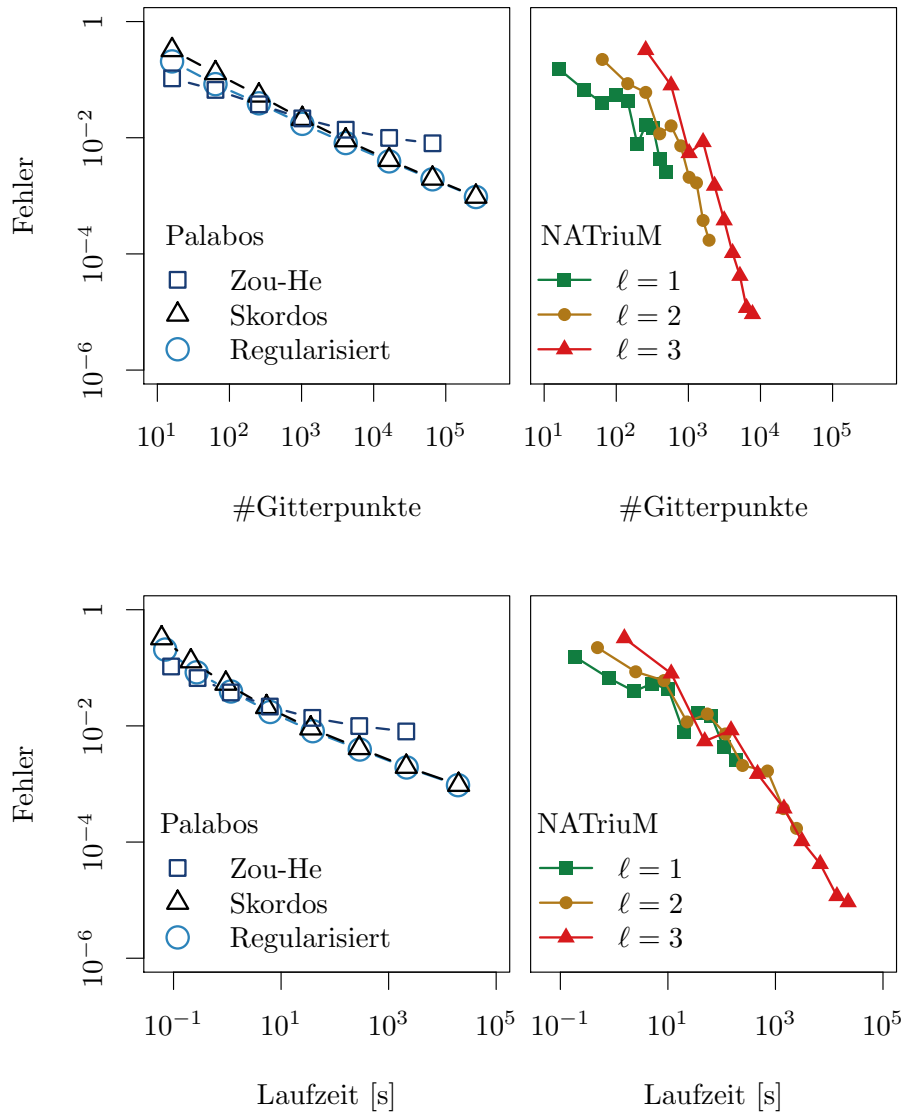
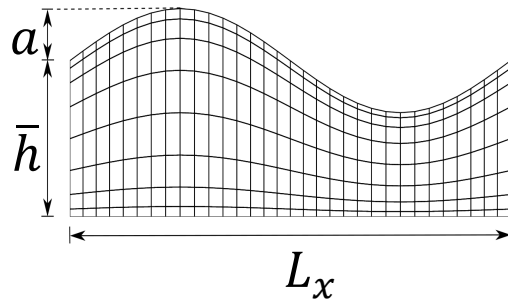


Abbildung 4.1: Vergleich von NATriuM (SEDG-LBM) und Palabos (Standard-LBM) für die Simulation einer instationären Couette-Strömung. In Palabos wurde mit drei verschiedenen Randbedingungen gerechnet [140, 201, 244]. In NATriuM wurde die Verfeinerungsstufe ℓ des Gitters variiert. Die Fehler $\|\mathbf{u} - \mathbf{u}^{\text{ref}}\|_{\infty}/u_0$ sind aufgetragen gegen die Anzahl an Gitterpunkten und gegen die Rechenzeit.

Trotz des Mehraufwands pro Iteration und Gitterpunkt zahlt sich in diesem Fall die Verwendung der SEDG-LBM auch in Bezug auf die Laufzeit aus, insbesondere wenn eine sehr genaue Lösung angestrebt wird.

4.3.2. Simulationen auf irregulären Gittern

Ein weiterer Vorteil der SEDG-LBM besteht in der Möglichkeit, irreguläre Gitter zu verwenden. Um dies zu demonstrieren, wurden Scherströmungen in einer sinusförmigen Geometrie simuliert. Diese stationären Testfälle wurden vorher in [4] mit der Standard-LBM gerechnet, um tribologische Kennzahlen beim Schmiermittelfluss an rauhen Oberflächen zu bestimmen [22]. Da die Reynoldszahlen in der tribologischen Anwendung typischerweise sehr klein sind, wurde eine Reynoldszahl von $Re = 1$ verwendet, wobei die charakteristische Länge als $l_0 = 1$ und die charakteristische Geschwindigkeit als $u_0 = 0.1$ definiert war. Die Machzahl wurde auf $Ma = 0.01$ gesetzt, um potenziell große Druckdifferenzen vor und nach der Verengung abbilden zu können.



Konfiguration	L_x	a	\bar{h}
Konf. 1	5	0.05	0.3
Konf. 2	1	0.05	0.3
Konf. 3	5	0.1	0.3
Konf. 4	1	0.1	0.3

Abbildung 4.2: Schematische Darstellung des Rechengitters für Scherströmungen in sinusförmigen Geometrien. Das Gebiet $\{(x, y) : 0 < x \leq 2\pi, 0 \leq y \leq \bar{h} + a \sin(2\pi x/L_x)\}$ ist parameterisiert über die Länge L_x , die mittlere Höhe \bar{h} und die Amplitude a . Die getesteten Konfigurationen sind in der nebenstehenden Tabelle aufgelistet und stammen aus [4].

Abbildung 4.2 zeigt eine schematische Darstellung des Rechengitters in NATrium und listet die getesteten Konfigurationen auf. Jede der vier Konfigurationen wurde in Palabos auf einem regulären Gitter simuliert. An der unteren und oberen Wand wurde jeweils eine Geschwindigkeit von $u_1(y = 0) = u_0 = 0.1$ und $u_1(y = 1) = 0$ vorgegeben. In x -Richtung wurden periodische Randbedingungen gesetzt. Die Strömung wurde initialisiert mit Gleichgewichtsverteilungen der Geschwindigkeit $\mathbf{u} = 0$ und Dichte $\rho_0 = 1$.

In jedem hundertsten Schritt wurde der Scherflussfaktor

$$\Psi_x := \frac{\sqrt{2} \cdot \bar{h}}{a} \left(2 \cdot \frac{u_w - \langle u_1 \rangle}{u_w} - 1 \right)$$

ausgewertet. In Palabos wurde der Mittelwert $\langle u_1 \rangle$ über alle Fluidknoten gebildet. In NATrium wurde dazu die Quadratur der Discontinuous-Galerkin-Diskretisierung verwendet. Die Simulationen wurden abgebrochen, wenn die Änderung der Scherflussfaktoren über 100 Iterationen kleiner als 10^{-6} war.

Da sich bei den langen Geometrien (Konf. 1 und Konf. 3) keine Wirbel bilden, sind dort die Trägheitskräfte des Fluids sehr klein. In diesem Fall erhält man eine analytische Approximation für die Scherflussfaktoren [143]

$$\Psi_x = \frac{3\sqrt{2}\frac{a}{h}}{2 + \left(\frac{a}{h}\right)^2}. \quad (4.5)$$

Zur Simulation in Palabos wurden die Festkörperknoten oberhalb der oberen Wand durch *Bounce-Back*-Knoten dargestellt. An der unteren Wand wurde eine regulierte Randbedingung gesetzt. In NATrium wurden die Randbedingungen der SEDG-LBM über Gleichung (4.4) gesetzt.

Alle Simulationen wurden mit der in Unterabschnitt 4.3.1 beschriebenen Hardware durchgeführt. Die Gitterauflösung in Palabos wurde variiert, wobei für die maximale Höhe der Geometrie, $(\bar{h} + a)$, insgesamt 2^ℓ , $\ell = 5, \dots, 10$ Gitterpunkte verwendet wurden. In NATrium wurden die Ordnung der Ansatzfunktionen $N = 2, 4, 6, 8$ und die Gitterauflösung $\ell = 2, 3, 4$ variiert, wobei das Gitter aus $2 \cdot 4^\ell$ Zellen bestand.

Die Zeitintegration in der SEDG-LBM wurde zunächst mit einem klassischen Runge-Kutta-Verfahren vierter Ordnung und $CFL = 0.4$ durchgeführt. In beiden Codes wurde das BGK-Kollisionsmodell mit einem D2Q9-Stern verwendet.

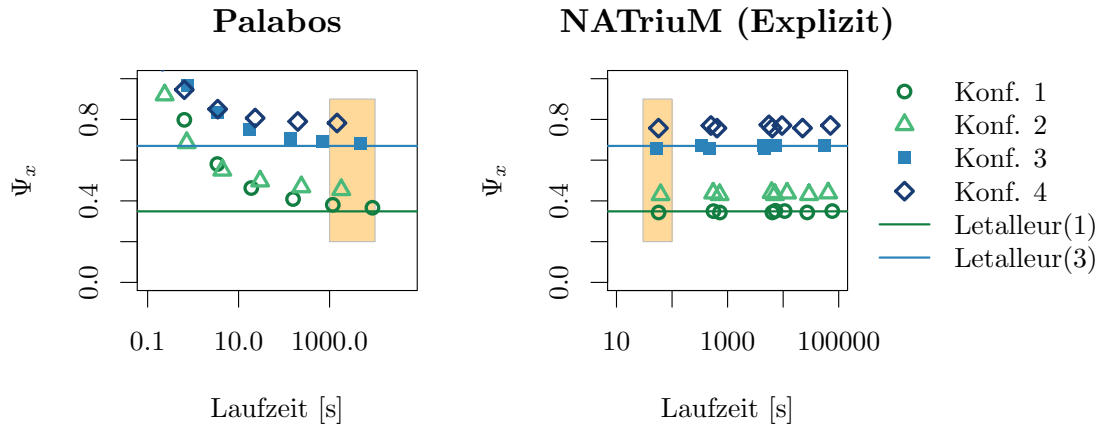


Abbildung 4.3: Numerisch ermittelte Scherflussfaktoren Ψ_x in Simulationen mit Palabos und NATrium. Die Simulationen in NATrium wurden mit einem expliziten Runge-Kutta-Verfahren vierter Ordnung mit CFL-Zahl 0.4 durchgeführt. In beiden Programmen wurde die Gitterauflösung und in NATrium zusätzlich die Ordnung der Ansatzfunktionen variiert. Die Konfigurationen Konf. 1–4 sind in Abbildung 4.2 beschrieben. Die Referenzlösungen von Letalleur [143] wurden mit Gleichung (4.5) berechnet.

Abbildung 4.3 zeigt die numerisch ermittelten Scherflussfaktoren in Palabos und NATrium für die verschiedenen Auflösungen und Ordnungen. In Palabos wurden erst

bei der höchsten Auflösung ausreichend genaue Flussfaktoren ermittelt. Die Flussfaktoren der langen Geometrien (Konf. 1 und Konf. 3) konvergierten dabei gegen den analytischen Ausdruck von Letalleur et al. [143]. Mit der SEDG-LBM konnte dagegen durch die Verwendung geometrieangepasster Gitter schon für die niedrigste Auflösung eine gute Lösung erzielt werden. Die benötigte Laufzeit wurde dadurch um fast zwei Größenordnungen auf weniger als eine Minute reduziert.

In dieser Arbeit wurde die SEDG-LBM um mehrere Zeitintegratoren erweitert. Da der zeitabhängige Verlauf der Lösung für dieses stationäre Problem nicht interessant ist, wurde diese Strömung auch mit einem impliziten Integrator und einer CFL-Zahl von $CFL = 100$ simuliert. Dazu wurde eine Theta-Methode mit $\theta = 0.5$ verwendet. Der wesentliche Vorteil des impliziten Integrators besteht in diesem Zusammenhang darin, dass der Strömungsschritt für beliebig große Zeitschritte stabil bleibt.

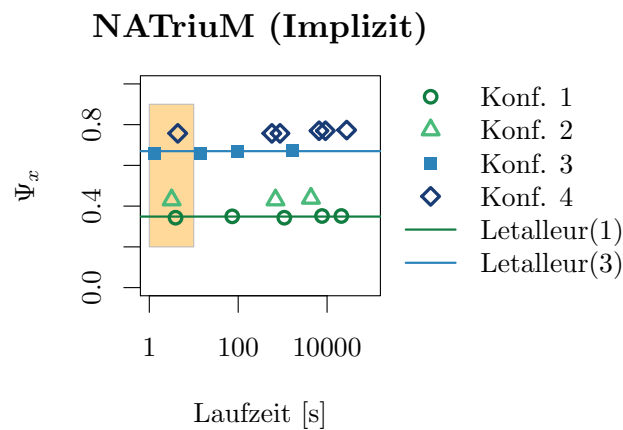


Abbildung 4.4: Numerisch ermittelte Scherflussfaktoren Ψ_x in NATrium bei $CFL = 100$. Die große CFL-Zahl wurde durch die Verwendung einer impliziten Theta-Methode mit $\theta = 0.5$ ermöglicht. Die Konfigurationen Konf. 1–4 sind in Abbildung 4.2 beschrieben. Die Referenzlösungen von Letalleur [143] wurden mit Gleichung (4.5) berechnet.

Abbildung 4.4 zeigt, dass die niedrigste Auflösung von 32 Zellen auch hier wieder genaue Ergebnisse lieferte. Mit dem drastisch vergrößerten Zeitschritt mit $CFL = 100$ wurde der Aufwand der Simulation um eine weitere Größenordnung auf wenige Sekunden verringert.

Dabei wurden die Simulationen in Palabos bewusst einfach gehalten. Auf Seiten der Standard-LBM könnten die Ergebnisse durch die Verwendung von *Off-Lattice*-Randbedingungen, wie zum Beispiel *Immersed-Boundary*-Methoden [19], verbessert werden. Zusätzlich könnte das Gitter an den Rändern hierarchisch verfeinert werden [66].

Insgesamt zeigen die Simulationen jedoch, dass OLBMs wie die SEDG-LBM bekannte Schwachpunkte der Standard-LBM kurieren können: Erstens ermöglichen sie eine genauere Darstellung komplexerer Strömungsgeometrien. Zweitens erlauben Zellen, de-

ren Form den Geschwindigkeitsgradienten angepasst sind, eine Reduktion der Gitterpunkte durch eine Verfeinerung des Gitters am Rand. Drittens ermöglichen OLBMs die Verwendung impliziter Integratoren, was die Simulation stationärer Strömungen beschleunigen kann.

4.3.3. Turbulente Kanalströmung

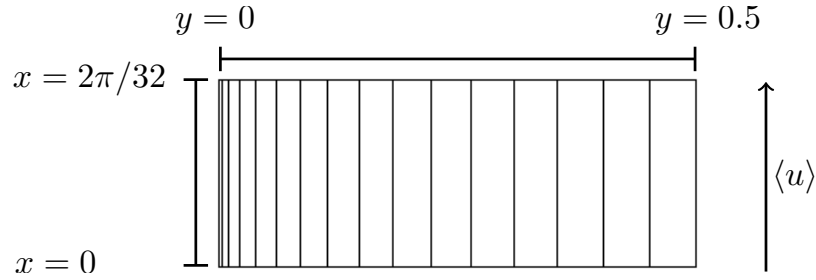


Abbildung 4.5: Ausschnitt des irregulären Gitters zur Simulation einer turbulenten Kanalströmung. Der Vektor $\langle u \rangle$ zeigt in Stromabrichtung.

Als letzter Testfall für die SEDG-LBM wurde eine dreidimensionale turbulente Kanalströmung als implizite LES simuliert. Das Strömungsgebiet $[0, 2\pi] \times [0, 1] \times [0, 2\pi/3]$ wurde durch ein Gitter mit $32 \times 32 \times 32$ Zellen diskretisiert. In x - und z -Richtung waren diese Zellen äquidistant angeordnet. An den Rändern wurden dort jeweils periodische Randbedingungen gesetzt. In y -Richtung wurde das Gitter zu den Festkörperperrändern hin verfeinert, an denen die Randbedingung (4.4) mit $\mathbf{u}_w = 0$ verwendet wurde. Abbildung 4.5 zeigt die Verfeinerung dieses Gitters entlang der wandnormalen Koordinate.

In x -Richtung wurde eine Volumenkraft F_1^{ext} definiert, um eine Reynoldszahl $Re_\tau = \frac{u_\tau \delta}{\nu}$ vorzugeben. Dabei bezeichnet δ die halbe Kanalhöhe und $u_\tau := \sqrt{\frac{\tau_w}{\rho}}$ die Wandschubspannungsgeschwindigkeit. Da die Volumenkraft F_x der Schubspannung der beiden Wände entgegenwirken muss, ergibt sich

$$\begin{aligned} \int_{\mathbb{R}^3} F_1^{\text{ext}} dx dy dz &= 2 \int_{\mathbb{R}^2} \tau_w dx dz \quad \Leftrightarrow \quad 2\delta F_1^{\text{ext}} = 2\tau_w \\ &\Leftrightarrow \quad F_1^{\text{ext}} = \frac{\rho u_\tau^2}{\delta} = \frac{\rho Re_\tau^2 \nu^2}{\delta^3}. \end{aligned}$$

Die Reynoldszahl wurde auf $Re_\tau = 140$ gesetzt.

Die Advektion wurde mit einem SEDG-LBM-Verfahren mit Exponentialintegrator und $CFL = 2$ durchgeführt, da die Verwendung verschiedener Zellgrößen die Kondition des Advektionsoperators erhöht. Der Exponentialintegrator erlaubt im Vergleich zum Runge-Kutta-Verfahren der Originalpublikation größere Zeitschritte, was für diese rechenaufwändigere Strömung einen entscheidenden Unterschied macht.

Exponentialintegratoren eignen sich besonders für steife Probleme [39]. Die Implementierung des Integrators ist an [217] angelehnt, im Gegensatz zu [217] wird das Matrix-Exponential allerdings durch eine Padé-Approximation statt durch eine Taylor-Entwicklung genähert, was im Allgemeinen die Eigenschaften des Zeitintegrators für steife Probleme mit oszillierenden Lösungen verbessert [106].

Die Ordnung des SEDG-Verfahrens war $N = 2$, sodass das Gitter aus $96 \times 96 \times 96$ Gitterpunkten bestand. Hierbei wurden Gitterpunkte auf den Zellgrenzen wieder doppelt gezählt. Diese minimale Konfiguration wurde zum einen gewählt, um den Rechenaufwand zu verkleinern und zum anderen, um die Stabilität des Verfahrens zu demonstrieren. Es wurde ein BGK-Kollisionsmodell mit einem D3Q19-Stern verwendet. Die Machzahl wurde auf $Ma = 0.1$ gesetzt. Diese Wahl rechtfertigt sich durch vorherige Simulationen von turbulenten Kanalströmungen mit der LBM [34, 137], wo die Machzahlen in der gleichen Größenordnung waren.

Die Strömung wurde zunächst mit einem empirischen mittleren Geschwindigkeitsprofil initialisiert, auf das inkompressible Fluktuationen addiert wurden, ähnlich wie in [14, 137]. Dann wurde die Strömung zunächst auf einem groben Gitter mit $8 \times 8 \times 8$ Zellen simuliert, bis eine Transition zu einer turbulenten Strömung eintrat. Das turbulente Profil wurde dann auf ein feineres Gitter mit $16 \times 16 \times 16$ Zellen interpoliert und simuliert, bis die Statistiken der RMS-Werte an allen y -Werten konvergiert waren. Nach einer weiteren Interpolation auf das finale Gitter wurde zunächst eine Äquilibriumierung über 400 000 Zeitschritte durchgeführt. Danach wurden über weitere 430 000 Zeitschritte die turbulenten Statistiken gemittelt. Die Mittelung wurde über die Zeit, die homogenen Richtungen und die beiden Hälften des Kanals durchgeführt. An der Wand ergab sich mit

$$u_\tau = \sqrt{\nu \left\langle \frac{\partial u_1}{\partial y} \Big|_{y=0} \right\rangle}$$

eine tatsächliche Reynoldszahl von 137.22. Der Ausdruck $\langle \cdot \rangle$ bezeichnet dabei einen Reynolds-Mittelwert.

Abbildung 4.6 zeigt das gemittelte Geschwindigkeitsprofil über der wandnormalen Koordinate. Dabei wurden die Geschwindigkeiten und Koordinaten jeweils mit den tatsächlichen u_τ und $l^+ = \frac{\nu}{u_\tau}$ in sogenannte „+-Einheiten“ normiert. Das mit der SEDG-LBM simulierte Geschwindigkeitsprofil stimmte gut mit der Referenzlösung [214] überein und reproduzierte das Logarithmische Wandgesetz und die viskose Unterschicht. In der Referenz [214] wurde ein Finite-Differenzen-Verfahren mit einem feineren $256 \times 128 \times 256$ -Gitter verwendet, beginnend bei $y^+ = 0.17$. Der erste Gitterpunkt im hier verwendeten Gitter lag bei $y^+ = 0.33$.

Abbildung 4.7 zeigt das Geschwindigkeitsprofil und die Standardabweichungen der Geschwindigkeitskomponenten im Vergleich zu [214]. Alle Kurven stimmen gut mit der Referenzlösung überein. Die Geschwindigkeit in der Kanalmitte wird leicht unterschätzt, was auf die etwas größere Reynoldszahl der Referenzlösung zurückzuführen

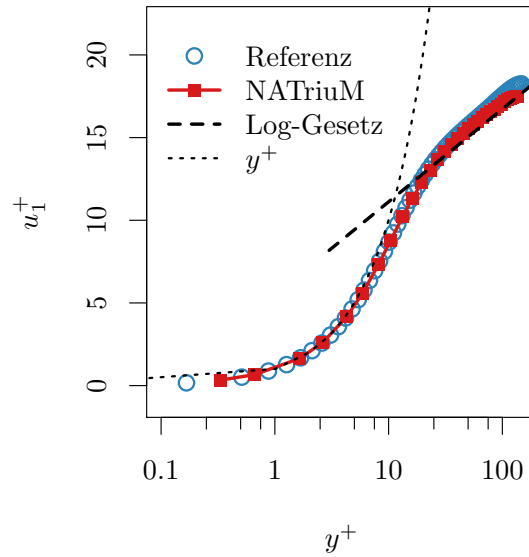


Abbildung 4.6: Gemitteltetes normiertes Geschwindigkeitsprofil $u_1^+ = u_1/u_\tau$ der turbulenten Kanalströmung im Vergleich zur Referenzlösung [214]. In der Referenzlösung war die Reynoldszahl $Re_\tau = 147.53$. Zur Einordnung sind außerdem das Logarithmische Wandgesetz $0.41 \cdot \ln(y^+) + 5.5$ (Log-Gesetz) und das lineare Profil der viskosen Unterschicht $u_1^+ = y^+$ eingezeichnet.

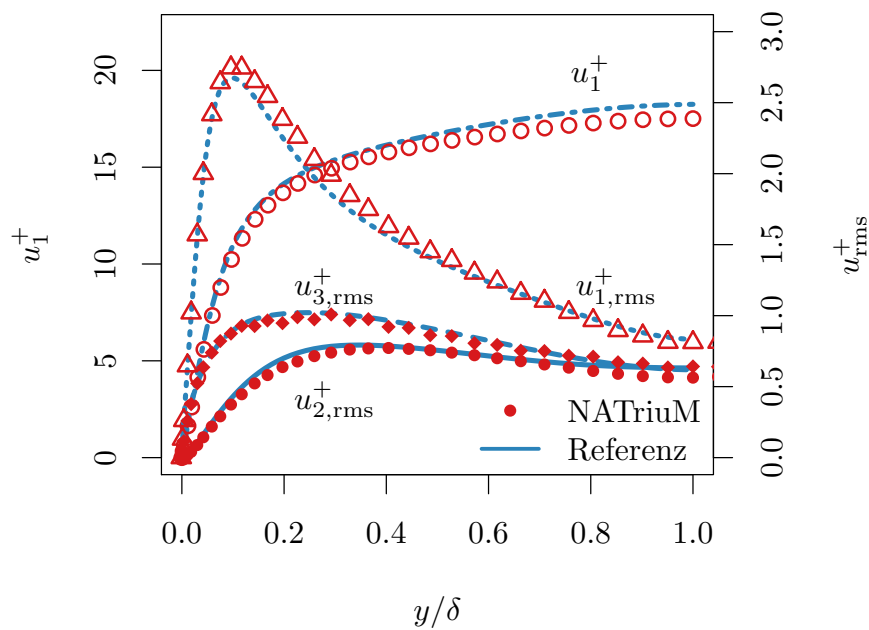


Abbildung 4.7: Mittleres Geschwindigkeitsprofil u_1^+ und Standardabweichungen $u_{i,rms}^+$ im Vergleich zur Referenzlösung [214].

ist. Im Bereich von $y \approx 0.3 \cdot \delta$ weichen die Standardabweichungen leicht von der Referenzlösung ab. Dass diese Abweichung bei der Simulation auf den gröberen Gittern noch stärker ausgeprägt war, weist darauf hin, dass es sich hierbei um ein Artefakt der niedrigen Auflösung handelt.

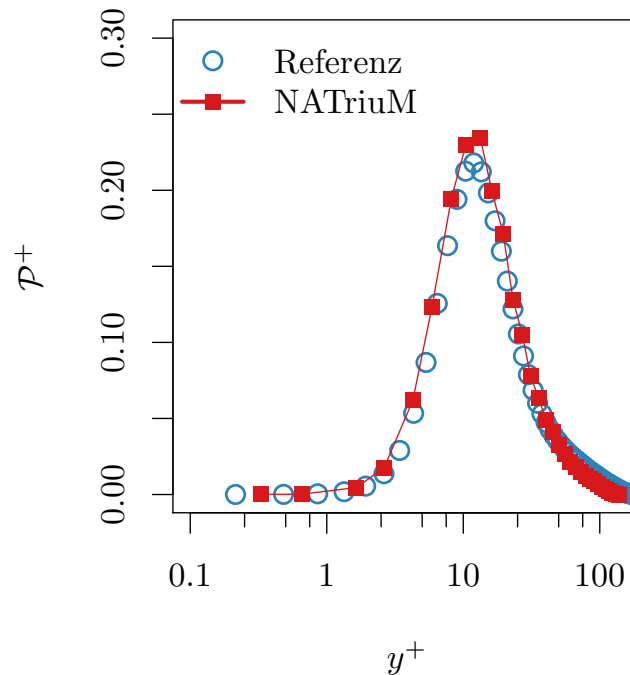


Abbildung 4.8: Produktion der turbulenten kinetischen Energie im Vergleich zur Referenzlösung von Moser et al. [168]. Die Produktion ist definiert als $\mathcal{P} = -\langle u_1' u_2' \rangle \langle \partial u_1 / \partial y \rangle$. Die Reynoldszahl in der Referenzlösung war $Re_\tau = 178.12$.

Abbildung 4.8 zeigt die Produktion der turbulenten kinetischen Energie. Unabhängig von der Reynoldszahl hat die Produktion in turbulenten Kanalströmungen ihr Maximum bei $y^+ \approx 12$ [154]. Die simulierten Ergebnisse stimmen darin gut mit der Referenzlösung von Moser et al. [168] überein.

Die Auflösung und die simulierte Zeit waren jedoch in dieser Simulation nicht hoch genug, um Statistiken höherer Ordnung akkurat wiederzugeben. Die Reynoldsspannungen wurden um 10% überschätzt und auch die Flachheit und Schiefe waren verfälscht. Um diese Statistiken höherer Ordnung zu reproduzieren, wäre eine längere und höher aufgelöste Simulation notwendig gewesen. Diese wurde aber wegen des hohen Rechenaufwands nicht durchgeführt, der im Folgenden diskutiert wird.

4.3.3.1. Rechenzeit

Obwohl die Geschwindigkeitsprofile und Standardabweichungen gut mit den Referenzlösungen übereinstimmen, ist die Rechenzeit der turbulenten Kanalströmung

mit der SEDG-LBM kritisch zu bewerten. Die Simulation lief für eine Woche auf 8 Knoten, wobei jeder Knoten mit zwei Intel Xeon X5650 Sechskern-Prozessoren und 48 GB RAM bestückt war. Dieser hohe Rechenbedarf kommt durch drei Faktoren zustande.

Der erste Faktor ist der kritische Speicheraufwand für die Advektionsoperatoren. Für den D3Q19-Stern müssen 18 solcher dünnbesetzten Matrizen gespeichert werden. Da die Freiheitsgrade jeder Zelle im Strömungsschritt miteinander gekoppelt sind, besitzt jede dieser Matrizen Diagonalelemente der Größe $(N + 1)^3 \cdot (N + 1)^3$. Zusätzlich müssen für jede Zelle $O(N)$ Nicht-Diagonalelemente zur Berechnung der Flüsse abgespeichert werden. Durch die Fluss-Randbedingung (4.4) sind die Matrizen darüber hinaus miteinander gekoppelt. Der Speicheraufwand ist dadurch im Vergleich sowohl zu klassischen Methoden der CFD als auch zur Standard-LBM enorm.

Dieser hohe Speicheraufwand erhöht besonders im Dreidimensionalen den rechnerischen Aufwand, da in jedem Strömungsschritt auf alle Elemente zugegriffen wird. Da der Anteil der Cache-Misses in NATrium sich hierbei auf 2% aller Speicherzugriffe beschränkte, ist der limitierende Faktor hier nicht die Cache-Effizienz, sondern die Speicherbandbreite. Um dieses Problem zu umgehen, könnte der Advektionsschritt matrixfrei implementiert werden. Hierbei werden die Matrix-Elemente in jedem Schritt neu berechnet, was effizienter sein kann als auf abgespeicherte Einträge zuzugreifen [128]. In [128] konnte die Auswertung eines Laplace-Operators durch die Matrix-freie Implementierung um einen Faktor zwei bis fünf (je nach Problemgröße und Hardware) beschleunigt werden.

Der zweite Faktor ist, dass pro Iteration mehrere solcher Matrix-Vektor-Multiplikationen durchgeführt werden müssen. Für das klassische Runge-Kutta-Verfahren werden dazu 5 Multiplikationen benötigt. Aus diesem Grund war der Advektionsschritt in der vorliegenden Implementierung etwa zehnmal teurer als der Kollisionsschritt.

Der dritte limitierende Faktor besteht in der CFL-Bedingung, die den Zeitschritt durch die kleinsten Zellen, die Ordnung des Verfahrens und die Schallgeschwindigkeit vorgibt. Da die Schallgeschwindigkeit in der LBM die Fluidgeschwindigkeit üblicherweise um eine Größenordnung übersteigt, ist der Zeitschritt im Kontext aller Lattice-Boltzmann-Methoden generell schon kleiner zu wählen als in klassischen CFD-Verfahren. Die quadratische Abhängigkeit der CFL-Bedingung von der Ordnung der Ansatzfunktionen erzwingt bei der SEDG-LBM eine weitere Verkleinerung des Zeitschritts. Da bei der Kanalströmung das Gitter an der Wand zusätzlich fein aufgelöst war, lag der Zeitschritt in dieser Simulation bei $\delta_t^+ = \delta_t / (\nu / u_\tau^2) \approx 0.0083$. Dieser Zeitschritt ist für ein Verfahren zweiter Ordnung bei einem vergleichsweise groben Gitter sehr klein, siehe unten für eine ausführlichere Diskussion.

Insgesamt summieren sich diese Probleme zu einem Rechenaufwand, der weder mit klassischen CFD-Verfahren noch mit der Standard-LBM konkurrieren kann.

4.4. Diskussion der Discontinuous-Galerkin-Lattice-Boltzmann-Verfahren

In diesem Abschnitt werden die Vor- und Nachteile der getesteten Varianten der SEDG-LBM abschließend diskutiert. Diese Diskussion dient unter anderem zur Motivation eines neuartigen Semi-Lagrangeschen Strömungsschritts, der im nächsten Kapitel beschrieben wird.

Die Simulation der instationären Couette-Strömung demonstriert die besseren Konvergenzeigenschaften der SEDG-LBM gegenüber der Standard-LBM. Die exponentielle Konvergenz ist in Übereinstimmung mit anderen Publikationen zu OLBMs hoher Ordnung [54, 95, 97, 161]. Als Konsequenz lassen sich selbst auf groben Gittern sehr genaue Lösungen erzielen, was im Gesamten zu einer besseren Effizienz der SEDG-LBM führt, wenn eine genaue Lösung erzielt werden soll.

Auch die Simulation von Scherströmungen in sinusförmigen Geometrien verdeutlicht die Vorteile der SEDG-LBM. Während die Standard-LBM mit Bounce-Back-Randbedingung erst mit einer sehr hohen Auflösung eine genaue Lösung lieferte, konnte mit der SEDG-LBM die Auflösung und damit der Rechenaufwand drastisch reduziert werden. Die gesteigerte Effizienz von OLBMs gegenüber der Standard-LBM in komplexeren Geometrien wurde von anderen Autoren auch für Simulationen poröser Medien gezeigt [146, 164, 235].

Die Verwendung eines impliziten Integrators führte zu einer weiteren Reduktion der Rechenzeit. Im Rahmen von Finite-Volumen und Finite-Elemente-LBMs wurde bereits gezeigt, dass die Verwendung impliziter Verfahren den Rechenaufwand für stationäre Probleme reduzieren kann [109, 110, 141, 144]. In dieser Arbeit wurde aber zum ersten Mal eine Discontinuous-Galerkin-LBM mit einem impliziten Integrator kombiniert.

Die Simulation einer turbulenten Kanalströmung stellte für die SEDG-LBM eine größere Herausforderung dar. Wie oben diskutiert, sind der rechenintensive Strömungsschritt und dessen speicherintensive Implementierung sowie die kritische CFL-Bedingung hier problematisch. Durch die CFL-Bedingung wurde ein Zeitschritt von $\delta_t^+ \approx 0.0083$ erzwungen.

Andere Verfahren konnten bei einer Scher-Reynoldszahl von $Re_\tau = 180$ mit einem signifikant größeren Zeitschritt arbeiten. Trotz eines feineren Gitters wurde bei Kim et al. [123] ein Zeitschritt von $\delta_t^+ = 0.0676$ verwendet. In [35] konnte der Zeitschritt mithilfe voll-impliziter Zeitintegratoren sogar auf $\delta_t^+ = 0.4$ erhöht werden, ohne die turbulenten Statistiken wesentlich zu verfälschen. Auch bei der Simulation einer turbulenten Kanalströmung mit der Standard-LBM [137] konnte zumindest ein Zeitschritt von $\delta_t^+ \approx 0.014$ verwendet werden, wobei das Gitter 128 Punkte in wandnormaler Richtung hatte. Dennoch war die LBM in dieser Publikation bei gleicher Ergebnisqualität etwa zweimal schneller als eine dort durchgeführte Vergleichsrechnung mit

einem Pseudo-Spektral-Code. In einer weiteren Simulation mit der Standard-LBM mit 200 Punkten in wandnormaler Richtung wurde der Zeitschritt auf $\delta_t^+ = 0.0116$ gesetzt.

Bei der einzigen publizierten turbulenten Kanalströmung mit einer OLBM [14] war der Zeitschritt mit $\delta_t^+ = 0.001893$ um mehr als zwei Größenordnungen kleiner als bei [35]. Dies zeigt, dass die CFL-Bedingung auch für andere Eulersche OLBMs (in diesem Fall DUGKS) eine kritische Einschränkung darstellt.

Ein möglicher Ausweg ist die Verwendung eines Semi-Lagrangeschen statt eines Eulerschen Ansatzes zur Lösung des Advektionsterms. Das nächste Kapitel widmet sich daher der Entwicklung einer Semi-Lagrangeschen LBM, die neben großen Zeitschritten auch die Verwendung irregulärer Gitter und eine hohe räumliche Konvergenzordnung erlaubt.

5. Räumliche Diskretisierungen: Semi-Lagrangesche Methode

Die Diskussion der Discontinuous-Galerkin-LBM demonstriert ein grundsätzliches Dilemma der Eulerschen Off-Lattice-Boltzmann-Verfahren: Eine Eulersche Diskretisierung der Advektion erfordert entweder explizite oder implizite numerische Integration. Während explizite Integratoren kleine Zeitschritte erzwingen, erfordern implizite Integratoren die Lösung großer linearer Gleichungssysteme in jedem Strömungsschritt. In beiden Fällen dominiert die Numerik des Strömungsschritts bei weitem den rechnerischen Aufwand der Simulationen.

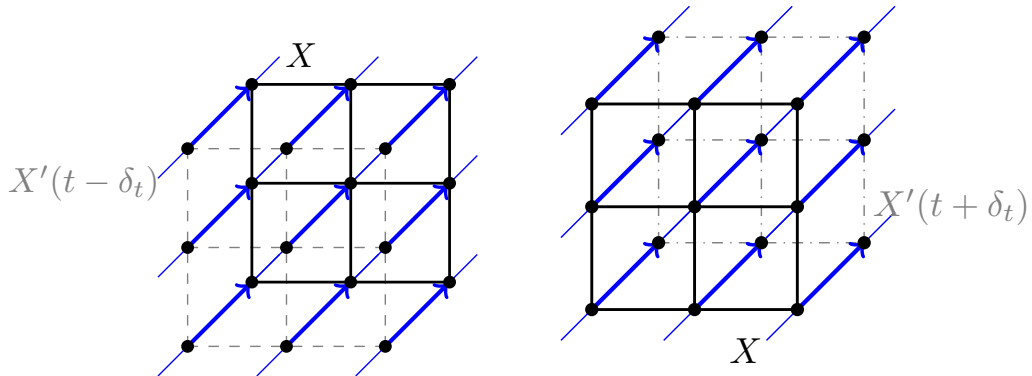
Im Rahmen dieser Arbeit wurde ein Verfahren entwickelt, das diesen grundsätzlichen Nachteil von Eulerschen Methoden adressiert. Diese Weiterentwicklung löst den Strömungsschritt mit einem Semi-Lagrangeschen statt einem Eulerschen Ansatz und wurde in [125] veröffentlicht.

Dieses Kapitel gibt zunächst eine kurze Einführung in Semi-Lagrange-Verfahren (Abschnitt 5.1) und einen Überblick über verwandte Methoden in der LBM (Abschnitt 5.2). Danach wird die neue Semi-Lagrangesche LBM (SLLBM) vorgestellt (Abschnitt 5.3) und in Simulationen von Wirbelströmungen getestet (Abschnitt 5.4).

5.1. Semi-Lagrange-Verfahren

Semi-Lagrange-Verfahren kombinieren Aspekte von Eulerschen und Lagrangeschen Verfahren. Wie Eulersche Verfahren operieren sie auf einem festen Gitter. Wie Lagrangesche Verfahren (und insbesondere die LBM) arbeiten sie mit Gleichungen, die bereits entlang der Charakteristiken integriert sind. Damit eignen sie sich besonders gut für advektionsdominierte Probleme, in denen das Verfolgen der Trajektorien effizienter ist als die numerische Integration [60, 61]. Ein bekanntes Beispiel dafür ist die numerische Wettervorhersage [112, 204]. Semi-Lagrangesche Verfahren für die lineare Advektionsgleichung lassen beliebig große Zeitschritte zu und erfordern pro

Zeitschritt nur eine Auswertung des Interpolationsoperators [61]. Sie vermeiden die kritischen Aspekte der Eulerschen LBM und bilden zudem eine natürliche Weiterentwicklung der inhärent Lagrangeschen LBM.



(a) Rückwärts-Verfolgen der Trajektorie (b) Vorwärts-Verfolgen der Trajektorie

Abbildung 5.1: Gitter X und temporäres Gitter X' im Semi-Lagrangeschen Strömungsschritt. Die blauen Linien und Pfeile symbolisieren die Trajektorien.

Semi-Lagrangesche Methoden verfolgen die Trajektorien der Strömung entweder vorwärts oder rückwärts in der Zeit [210], siehe Abbildung 5.1. Diese beiden Ansätze unterscheiden sich darin, dass eine skalare Größe Υ entweder von einem entlang der Charakteristiken geströmten Gitter $X'(t + \delta_t)$ auf das ursprüngliche Gitter X interpoliert werden muss (Abbildung 5.1b)

$$\begin{aligned} \Psi' : B(X', \mathbb{R}) &\rightarrow B(X, \mathbb{R}), \\ \Upsilon|_{X'} &\mapsto \Upsilon|_X, \end{aligned}$$

oder umgekehrt vor der Advektion vom Gitter X auf das Gitter $X'(t - \delta_t)$ interpoliert wird (Abbildung 5.1a)

$$\begin{aligned} \Psi : B(X, \mathbb{R}) &\rightarrow B(X', \mathbb{R}), \\ \Upsilon|_X &\mapsto \Upsilon|_{X'}. \end{aligned}$$

Dabei bezeichnen Ψ' und Ψ die jeweiligen Interpolationsoperatoren und $B(X, \mathbb{R})$ die Menge der beschränkten skalaren Funktionen auf X . Beim Vorwärts-Verfolgen ergibt sich der Advektionsschritt somit als

$$\Upsilon|_X(t + \delta_t) = \Psi'(\Upsilon|_{X'(t + \delta_t)}(t + \delta_t)) = \Psi'(\Upsilon|_X(t)),$$

beim Rückwärts-Verfolgen dagegen als

$$\Upsilon|_X(t) = \Upsilon|_{X'(t - \delta_t)}(t - \delta_t) = \Psi(\Upsilon|_X(t - \delta_t)).$$

Da im Strömungsschritt der LBM nur eine lineare Advektionsgleichung zu lösen ist, ist das Verfolgen der Trajektorien trivial. Beim Vorwärts-Verfolgen der Trajektorie ist

die Interpolation jedoch schwieriger zu realisieren. Dort muss entweder das temporäre Gitter X' explizit abgespeichert werden oder es müssen zumindest die zur Interpolation benötigten Punkte im Vorfeld identifiziert werden, was auf einem irregulären Gitter algorithmisch ungünstiger ist. Beim Rückwärts-Verfolgen sind dagegen alle zur Interpolation benötigten Punkte bereits im Gitter X abgespeichert. Daher ist dieser Ansatz in der Regel einfacher und effizienter zu realisieren.

5.2. Stand der Technik: Interpolationsbasierte Lattice-Boltzmann-Methoden

Im Kontext der LBM wurden Semi-Lagrangesche Ideen in den sogenannten interpolationsbasierten LBMs aufgegriffen. Trotz der oben beschriebenen Vorteile hat dieser Ansatz in der LBM wesentlich weniger Beachtung gefunden als die Eulerschen Methoden. Allerdings wurde der Semi-Lagrangesche Ansatz in den bestehenden OLBMs auch nicht effizient umgesetzt.

In [93] wurde ein erstes interpolationsbasiertes Verfahren für die diskrete BGK-Gleichung formuliert, das in [90, 91] auf nicht-kartesische Gitter erweitert wurde. Dieses Verfahren strömt die Verteilungsfunktionen f_i am Punkt \mathbf{x}_k zunächst zum Punkt $\mathbf{x}_k + \delta_t \mathbf{e}_i$, der im Allgemeinen nicht auf dem Gitter liegt. Aus diesen geströmten Verteilungsfunktionen $f_{(i)}(t + \delta_t, \mathbf{x} + \delta_t \mathbf{e}_{(i)})$ werden die Verteilungsfunktionen auf den ursprünglichen Gitterpunkten $f_i(t + \delta_t, \mathbf{x}_k)$ durch eine polynomielle Interpolation rekonstruiert. Diese Prozedur erfordert, dass das geströmte Gitter zumindest in Teilen explizit abgespeichert wird. Die Interpolation wird dann für jeden Gitterpunkt einzeln durchgeführt. Der Ansatz wurde in [163] in Kombination mit geeigneteren *Mapping*-Funktionen verwendet und in [31] auf Lagrange-Interpolationen dritter Ordnung erweitert. In [89] wurde gezeigt, dass Interpolationen mindestens zweiter Ordnung die Viskosität der Strömung nicht verfälschen. Allerdings können sie die Hyper-Viskositäten und Galilei-Invarianz beeinflussen [135].

In [32, 196, 197] sowie [85] wurde die interpolationsbasierte LBM weiterentwickelt. Dabei wurden die Verteilungsfunktionen $f_i(t + \delta_t, \mathbf{x}_k)$ an den Gitterpunkten \mathbf{x}_k in eine Taylor-Reihe zweiter Ordnung entwickelt. Die Ableitungen in dieser Taylor-Entwicklung sowie die Funktionswerte selbst sind unbekannt. Allerdings sind die Verteilungsfunktionen auf dem geströmten Gitter $f_{(i)}(\mathbf{x}_l + \delta_t \mathbf{e}_{(i)}, t + \delta_t \mathbf{e}_{(i)})$ bekannt, wobei \mathbf{x}_l benachbarte Gitterpunkte bezeichnet. Diese bekannten Funktionswerte werden in die Taylor-Entwicklung eingesetzt, sodass sich ein überbestimmtes Gleichungssystem ergibt, welches mit einer linearen Regression näherungsweise gelöst wurde. Da das Gitter zeitlich konstant ist, musste die Regression nur einmal zu Beginn der Simulation gelöst werden. Dieser Ansatz wurde anhand einiger zweidimensionaler Strömungen auf nicht-kartesischen Gittern validiert.

In [187] wurden die Trajektorien schließlich zum ersten Mal rückwärts verfolgt. Wäh-

rend in allen zuvor beschriebenen interpolationsbasierten LBMs die geströmten Verteilungsfunktionen $f_{(i)}(t + \delta_t, \mathbf{x}_k + \delta_t \mathbf{e}_{(i)})$ zurück auf das Gitter interpoliert wurden, wurde hier $f_{(i)}(t, \mathbf{x}_k - \delta_t \mathbf{e}_{(i)})$ durch eine Interpolation rekonstruiert. Wie oben beschrieben, ist die Rückwärts-Verfolgung für die lineare Advektionsgleichung effizienter. Allerdings wurden die Basisfunktionen der Interpolation in [187] so definiert, dass ein nichtlineares Gleichungssystem gelöst werden musste, um die Ausgangspunkte $\mathbf{x}_k - \delta_t \mathbf{e}_i$ zu finden. Des Weiteren wurde die Interpolation nur über die direkten Nachbarpunkte berechnet, wodurch der Zeitschritt auf $CFL < 1$ beschränkt wird. Dadurch relativiert sich ein wesentlicher Vorteil des Lagrangeschen Ansatzes gegenüber den Eulerschen OLBMs.

5.3. Semi-Lagrangesches Lattice-Boltzmann-Verfahren

Im Folgenden wird die in dieser Arbeit entwickelte SLLBM [125] formuliert und theoretisch untersucht.

5.3.1. Formulierung

Die rückwärts integrierte lineare Advektionsgleichung

$$f_i(t, \mathbf{x}) = f_{(i)}(t - \delta_t, \mathbf{x} - \delta_t \mathbf{e}_{(i)}) \quad (5.1)$$

bildet den Ausgangspunkt für die Semi-Lagrangesche Formulierung des Strömungsschritts. Standardmäßig operiert die Lattice-Boltzmann-Methode auf regulären Gittern mit einer Maschenweite $\delta_x = \delta_t \xi_0$, sodass $\mathbf{x} - \delta_t \mathbf{e}_i$ einen Nachbarknoten auf dem Gitter bezeichnet. Um die Ortsdiskretisierung von der Zeitschrittweite und der Diskretisierung des Geschwindigkeitsraums zu entkoppeln, wird $f_{(i)}(t - \delta_t, \mathbf{x} - \delta_t \mathbf{e}_{(i)})$ im Folgenden polynomiell approximiert.

Zur Definition der Polynome wird wieder für jede Zelle des Gitters eine bilineare Mapping-Funktion \mathcal{M}_c definiert. Wie bei der Discontinuous-Galerkin-Methode wird auf der Einheitszelle eine Menge von Stützpunkten $\tilde{\mathbf{x}}_j$ und eine Basis von Lagrange-Polynomen $\tilde{\psi}_j$, $j = 1, \dots, (N + 1)^{\dim}$, definiert. Diese Ansatzfunktionen definieren auf jeder Gitterzelle D_c eine polynomielle Basis

$$\psi_{cj} = \begin{cases} \tilde{\psi}_j \circ \mathcal{M}_c, & \mathbf{x} \in D_c \\ 0, & \text{sonst} \end{cases}, \quad j = 1, \dots, (N + 1)^{\dim},$$

woraus sich folgende Approximation der Verteilungsfunktionen ergibt:

$$f_i(t, \mathbf{x}) \approx \hat{f}_{icj}(t) \psi_{cj}(\mathbf{x}). \quad (5.2)$$

Die Verwendung von Lagrange-Polynomen ist zum einen effizienter als beispielsweise Spline-Interpolation [153]. Zum anderen identifizieren Lagrange-Interpolationen die Freiheitsgrade \hat{f}_{icj} mit den Werten der Verteilungsfunktionen an den Stützstellen:

$$\hat{f}_{icj}(t) = f_i(t, \mathbf{x}_{cj}), \quad \mathbf{x}_{cj} = \mathcal{M}_c^{-1}(\tilde{\mathbf{x}}_j).$$

Aus diesem Grund kann der Kollisionsschritt direkt auf die Freiheitsgrade angewandt werden. Setzt man die Approximation (5.2) in den Strömungsschritt (5.1) ein, ergibt sich

$$\hat{f}_{icj}(t) = f_i(t, \mathbf{x}_{cj}) = \hat{f}_{(i)\zeta k}(t - \delta_t)\psi_{\zeta k}(\mathbf{x}_{cj} - \delta_t \mathbf{e}_{(i)}). \quad (5.3)$$

Zusammengenommen erhält man die SLLBM als

$$\begin{aligned} \hat{f}_{icj}^{\text{pc}} &= \Omega_i(\hat{f}_{cj}) && \text{(Kollision),} \\ \hat{f}_{icj}(t, \mathbf{x}_{cj}) &= \hat{f}_{(i)\zeta k}^{\text{pc}}(t - \delta_t)\psi_{\zeta k}(\mathbf{x}_{cj} - \delta_t \mathbf{e}_{(i)}) && \text{(Advektion).} \end{aligned}$$

In der Praxis lässt sich der gesamte Semi-Lagrangesche Strömungsschritt wegen der effizienten Rückwärts-Verfolgung der Trajektorien als dünnbesetzte Matrix-Operation $\hat{f}(t) = \Psi \hat{f}^{\text{pc}}(t - \delta_t)$ definieren. Bei der Verwendung eines zeitlich konstanten Gitters ist auch die Matrix Ψ im Verlauf der Simulation konstant.

5.3.2. Standard-Lattice-Boltzmann-Methode als Spezialfall

Wählt man ein reguläres Gitter, $N = 1$ und $\delta_x = \delta_t \xi_0$, so ergibt sich die Standard-LBM als Spezialfall der SLLBM. Im Gegensatz zur Standard-LBM ermöglicht die SLLBM allerdings die Verwendung irregulärer Gitter, größerer und kleinerer Zeitschritte sowie Diskretisierungen höherer Ordnung.

5.3.3. Stabilitätsuntersuchung

Obwohl die SLLBM keine CFL-Bedingung besitzt, erfordert die Wahl des Zeitschritts eine gewisse Vorsicht, da der Spektralradius des Advektionsoperators von der Zeitschrittweite abhängt. In diesem Abschnitt wird daher die Stabilität des Semi-Lagrangeschen Strömungsschritts untersucht.

5.3.3.1. Stabilität auf regulären Gittern

Der Strömungsschritt ist dann stabil, wenn die Lösung bei einer iterativen Anwendung beschränkt ist. Um Bedingungen für diese Voraussetzung herzuleiten, werden in der Folge die Eigenwerte des Strömungsoperators auf einem regulären Gitter untersucht. Dabei reicht es aus, den Operator $\Phi(\delta_t \xi_0) = \left(\tilde{\psi}_j(x_i - \delta_t \xi_0) \right)_{ij}$ auf einer

eindimensionalen Einheitszelle $[0, 1]$ mit periodischen Rändern $x = x + n$, $n \in \mathbb{Z}$ zu betrachten. Auf regulären Gittern lassen sich die Ergebnisse sofort durch ein Ähnlichkeitsargument auf Gitter mit periodischen Randbedingungen in höheren Dimensionen übertragen.

Die wesentlichen Parameter des Verfahrens sind der Grad der Lagrangeschen Basispolynome N , die Wahl der Stützstellen \tilde{x}_k und die Schrittweite $\delta_t \xi_0$. In der Folge werden drei Mengen von Stützstellen untersucht: Äquidistante Stützstellen, Gauß-Lobatto-Legendre-Punkte und Gauß-Lobatto-Tschebyscheff-Punkte.

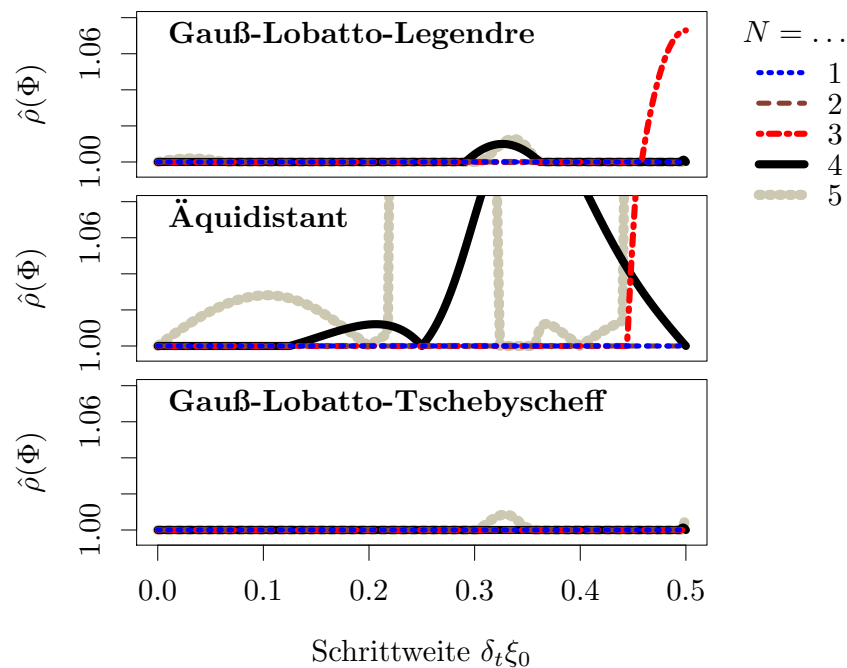


Abbildung 5.2: Spektralradius des Advektionsoperators $\hat{\rho}(\Phi)$ auf einem regulären Gitter in Abhängigkeit der Schrittweite $\delta_t \xi_0$, der Ordnung der Ansatzfunktionen N und der Verteilung der Stützstellen (Gauß-Lobatto-Legendre-Punkte, äquidistante Stützstellen und Gauß-Lobatto-Tschebyscheff-Punkte).

Abbildung 5.2 zeigt den Spektralradius des Advektionsoperators $\hat{\rho}(\Phi)$ in Abhängigkeit der Schrittweite $\delta_t \xi_0$. Für quadratische und lineare Ansatzfunktionen sind die drei Mengen von Stützstellen äquivalent und das Verfahren ist bedingungslos stabil. Für Ansatzfunktionen dritter Ordnung bilden nur die Gauß-Lobatto-Tschebyscheff-Punkte ein bedingungslos stabiles Verfahren. Die äquidistante Verteilung von Stützstellen ist für die Stabilitätseigenschaften generell ungünstig.

Dabei sind die Spektralradien nur für Schrittweiten $\delta_t \xi_0 \in [0, 0.5]$ dargestellt. Da die Zellen periodisch und die Stützstellen innerhalb einer Zelle symmetrisch verteilt sind, ist der Spektralradius symmetrisch in $[0, 1]$ und periodisch in $[0, \infty)$. Es ist zu

beachten, dass der Semi-Lagrangesche Advektionsschritt auf periodischen, regulären Gittern somit theoretisch beliebig große stabile Zeitschritte zulässt.

5.3.3.2. Stabilität auf irregulären Gittern

Während die bedingten Stabilitätsaussagen für Semi-Lagrange-Verfahren höherer Ordnung nur auf regulären Gittern gelten, überträgt sich die unbedingte Stabilität der Verfahren niedriger Ordnung direkt auf irreguläre Gitter (siehe [61]). Die Anwendung der Mapping-Funktionen verändert nichts am maximalen Spektralradius, weshalb sich die SLLBM bis zur Ordnung 2 stabil auf irregulären Gittern anwenden lässt.

Verwendet man Gauß-Lobatto-Tschebyscheff-Punkte, so ist das Verfahren mit $N = 3$ auf irregulären Gittern bedingungslos stabil. Auch für $N = 4$ ist der Spektralradius für fast alle Schrittweiten $\hat{\rho} = 1$. Nur im Intervall $\delta_t \xi_0 \in [0.49, 0.5]$ findet sich ein minimaler Anstieg auf 1.0015. Simulationen mit $N = 4$ und Gauß-Lobatto-Tschebyscheff-Punkten werden also auch in den allermeisten Fällen stabil sein, insbesondere da der Kollisionsschritt im Allgemeinen Dissipation erzeugt und somit zur Stabilisierung beiträgt.

Die Anwendung von Verfahren höherer Ordnung auf irregulären Gittern erfordert eine geschickte Wahl von Gitter und Zeitschritt, da hier theoretisch der Spektralradius des gesamten Advektionsoperators Ψ untersucht werden muss, um Stabilität auf allgemeinen Gittern zu gewährleisten.

5.3.3.3. Stabilität des Gesamtverfahrens

Die oben diskutierten Bedingungen betreffen nur die Stabilität des Strömungsschritts. Darüber hinaus beeinflusst natürlich auch das Zusammenspiel von Strömung und Kollision die Stabilität (siehe Kapitel 3). Während letztere Bedingung aber die Physik der Strömung reflektiert, ist die CFL-Bedingung Eulerscher OLBMs eine rein numerische Voraussetzung, welche nicht von der Reynoldszahl der Strömung oder der Strömungsgeschwindigkeit abhängt.

Der Übergang von einem Eulerschen zu einem Semi-Lagrangeschen Strömungsschritt löst dieses Problem und ermöglicht damit in vielen Situationen größere Zeitschritte. Eine praktische Auswertung der SLLBM findet sich im nächsten Abschnitt.

5.4. Simulationen mit der Semi-Lagrangeschen Lattice-Boltzmann-Methode

Im Folgenden werden Simulationsergebnisse der SLLBM in der Anwendung auf Wirbelströmungen gezeigt. Wenn nicht anders spezifiziert, wurden dazu die Gauß-Lobatto-Legendre-Stützstellen, reguläre Gitter und das BGK-Kollisionsmodell verwendet. Bei zweidimensionalen Strömungen wurde ein D2Q9 und bei dreidimensionalen Strömungen ein D3Q19-Stern genutzt. Die Ergebnisse wurden zum großen Teil in [125, 126] publiziert.

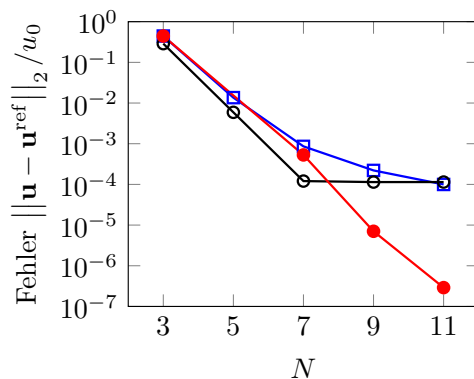
Die Simulationen in dieser Arbeit beschränken sich auf periodische Strömungen, um ein sauberes Fundament für den Semi-Lagrangeschen Strömungsschritt zu legen. Die Einbindung von Randbedingungen wird in Anhang D kurz beschrieben.

5.4.1. Konvergenz: Zweidimensionaler Taylor-Green-Wirbel

Um die Konvergenz der SLLBM zu testen, wurde zunächst der zweidimensionale Taylor-Green-Wirbel aus Abschnitt 2.6.1 mit der gleichen Reynoldszahl und Simulationszeit gerechnet.

5.4.1.1. Räumliche Konvergenz

—■— CFL größer —○— Ma größer —●— CFL, Ma klein



Simulation	CFL	Ma
CFL, Ma klein	0.1	0.008
Ma größer	0.1	0.1
CFL größer	2	0.008

Abbildung 5.3: Konvergenz der SLLBM bei der Simulation eines zweidimensionalen Taylor-Green-Wirbels mit verschiedener Ordnung der Ansatzfunktionen N auf einem Gitter mit 2×2 Zellen. Die konkreten CFL - und Machzahlen für die einzelnen Simulationsläufe sind in der nebenstehenden Tabelle aufgeführt.

Zunächst wurde die Ordnung der Ansatzfunktionen variiert. Abbildung 5.3 zeigt, dass der Fehler mit der Ordnung exponentiell abfiel, solange der zeitliche Fehler

und der Kompressibilitätsfehler durch genügend kleine CFL- und Machzahlen beschränkt wurden. Beim Fixieren der Machzahl auf $Ma = 0.1$ statt 0.008 konvergierte der Gesamtfehler gegen einen konstanten Kompressibilitätsfehler. Beim Fixieren der CFL-Zahl auf $CFL = 1$ statt 0.1 wurde der zeitliche Fehler dominant.

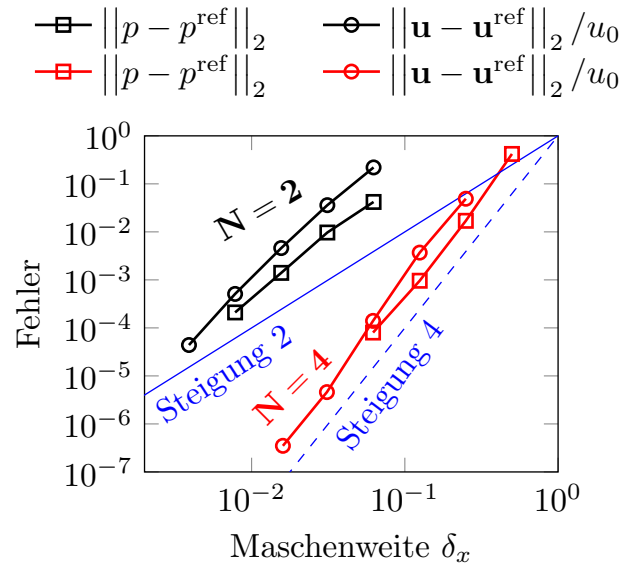


Abbildung 5.4: Konvergenz der SLLBM in Abhängigkeit der Maschenweite des Gitters δ_x bei der Simulation eines zweidimensionalen Taylor-Green-Wirbels. Die Ordnung der Ansatzfunktionen war $N = 2$ und $N = 4$. Die Graphen zeigen die Konvergenz des Drucks p und der Geschwindigkeit u gegen die analytische Referenzlösung bei $CFL = 0.1$ und $Ma = 0.008$.

Die höhere Konvergenzordnung des räumlichen Fehlers wurde bestätigt durch Simulationen mit unterschiedlicher Gitterauflösung. Hierbei waren die CFL- und Machzahl 0.1 und 0.008. Abbildung 5.4 zeigt, dass sowohl der Druck als auch die Geschwindigkeit mit der Ordnung der Ansatzfunktionen konvergierten.

5.4.1.2. Konvergenz des zeitlichen Fehlers

In der zeitlichen Konvergenzordnung ist keine Verbesserung gegenüber der Standard-LBM zu erwarten, da der zeitliche Fehler zweiter Ordnung im Operator-Splitting in Advektion und Kollision verankert ist (siehe Abschnitt 2.4.4 und Anhang A). Es ist lediglich zu bestätigen, dass die zeitliche Konvergenzordnung nicht – wie beispielsweise in [171] oder [165] – durch die OLBM degradiert wird. (Ähnliches gilt im nächsten Unterabschnitt für den quadratischen Fehler in der Machzahl.)

Abbildung 5.5 zeigt im oberen Bereich die Konvergenz des Fehlers in Abhängigkeit der Zeitschrittweite. Dabei wurden die räumliche Auflösung und die Machzahl so gewählt, dass der zeitliche Fehler der dominante Fehlerterm war ($N = 13$, $\delta_x = \pi$, $Ma =$

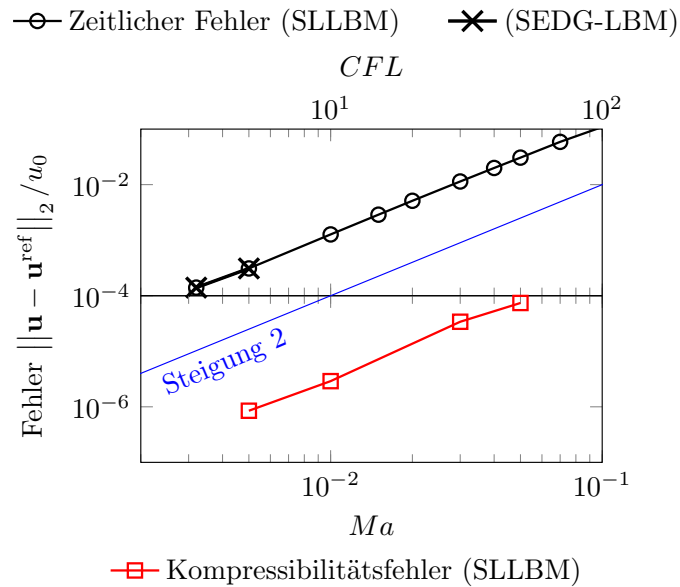


Abbildung 5.5: Oben: Konvergenz der SLLBM in Abhängigkeit von der CFL-Zahl bei der Simulation eines zweidimensionalen Taylor-Green-Wirbels bei konstanter Machzahl $Ma = 0.1$ und Gitterauflösung ($N = 13$, $\delta_x = \pi$). Zum Vergleich sind die zeitlichen Fehler der SEDG-LBM eingetragen. Für $CFL > 5$ waren die Simulationen mit der SEDG-LBM instabil. Unten: Konvergenz in Abhängigkeit von der Machzahl Ma bei konstanter Gitterauflösung ($N = 7$, $\delta_x = \pi/8$) und $CFL = 0.4$.

0.1). Dieser konvergierte über zwei Größenordnungen von CFL-Zahlen quadratisch. Für kleine CFL-Zahlen war die Übereinstimmung mit den Fehlern der SEDG-LBM exzellent. Im Vergleich zu Simulationen mit der SEDG-LBM blieb die SLLBM aber auch bei sehr großen CFL-Zahlen stabil.

Die theoretische Konvergenzordnung von Semi-Lagrange-Verfahren ist $O(\delta_x^N + \delta_x^N / \delta_t)$ (siehe zum Beispiel [57]), da jede Anwendung des Interpolationsoperators einen numerischen Fehler induziert. In Abbildung 5.5 ist wegen der hohen räumlichen Ordnung kein Anstieg des Fehlers bei kleinen CFL-Zahlen zu erkennen. Dieser Fehler wird aber in Unterabschnitt 5.4.6 noch eine Rolle spielen.

5.4.1.3. Konvergenz des Kompressibilitätsfehlers

Abbildung 5.5 zeigt im unteren Bereich die Konvergenz des Fehlers in Abhängigkeit der Machzahl. Dabei waren die Gitterauflösung und CFL-Zahl so gewählt, dass der Kompressibilitätsfehler dominant war ($N = 7$, $\delta_x = \pi/8$, $CFL = 0.4$). Wie in der Standard-LBM konvergierte der Fehler quadratisch in Abhängigkeit von der Machzahl. Dies war zu erwarten, da dieser Fehler durch die Diskretisierung des Geschwindigkeitsraums bedingt ist, die in der SLLBM identisch umgesetzt ist wie in der

Standard-LBM.

5.4.2. Performanz

Die Entwicklung der SLLBM wurde in dieser Arbeit durch die mangelhafte Effizienz anderer OLBMs und insbesondere der SEDG-LBM motiviert. Im vorigen Unterabschnitt wurde bereits gezeigt, dass die SLLBM wesentlich größere Zeitschritte erlaubt. In diesem Unterabschnitt wird die Performanz pro Zeitschritt untersucht und mit der SEDG-LBM verglichen.

Für den Vergleich wurde die SEDG-LBM mit verschiedenen Zeitintegratoren kombiniert. In [161] wurde die SEDG-LBM mit einem klassischen Runge-Kutta-Verfahren vierter Ordnung verwendet und in [217] mit einem Exponentialintegrator, der im Rahmen dieser Arbeit in ähnlicher Weise implementiert wurde [126]. Durch den modularen Aufbau von NATrium und die Einbindung verschiedener Runge-Kutta-Verfahren konnte die SEDG-LBM hier um verschiedene Zeitintegratoren erweitert und im Hinblick auf ihre Effizienz getestet werden.

Folgende Zeitintegratoren wurden mit der SEDG-LBM kombiniert: drei explizite Runge-Kutta-Methoden (das explizite Euler-Verfahren, ein explizites Runge-Kutta-Verfahren dritter Ordnung und das klassische *low-storage*-Runge-Kutta-Verfahren vierter Ordnung), vier implizite Runge-Kutta-Verfahren (das implizite Euler-Verfahren, die Mittelpunktmethode, das Crank-Nicolson-Verfahren und ein diagonal-implizites Runge-Kutta-Verfahren zweiter Ordnung (“singly diagonally implicit Runge-Kutta”, SDIRK) [5]), ein eingebettetes Runge-Kutta-Verfahren mit adaptiver Zeitschrittweitensteuerung nach Cash und Karp [27] und ein Exponentialintegrator [126, 199, 217]. Die SEDG-LBM mit diesen Zeitintegratoren wurde in Bezug auf die Effizienz pro Iteration mit der SLLBM verglichen.

Dabei wurde eine Strömung auf einem dreidimensionalen Gitter mit periodischen Randbedingungen simuliert. Die Problemgröße wurde auf ein Gitter mit insgesamt 110 592 Punkten festgesetzt und die Ordnung der Ansatzfunktionen war $N = 2$. Mit $CFL = 0.4$ und $Ma = 0.1$ wurde diese Simulation auf einem Cluster-Knoten gerechnet, wobei auf jedem Kern ein MPI-Prozess lief. Der Knoten war ausgestattet mit zwei CPUs des Typs Intel Xeon E5-2680v3 und 128 GB DDR4 RAM.

Die Performanz von LBM-Codes wird üblicherweise gemessen in *million lattice updates per second* (MLUPS). Obwohl die SEDG-LBM und die SLLBM nicht auf regulären Gittern (*Lattices*) operieren, wurde für diese Untersuchung das gleiche Maß verwendet, wobei die Anzahl an Lattice-Knoten durch die Anzahl an Gitterpunkten ersetzt wurde.

Abbildung 5.6 zeigt die Performanz der SLLBM im Vergleich mit der SEDG-LBM. Die SLLBM ist mit fast 10 MLUPS performanter als alle SEDG-LBMs und etwa fünf Mal schneller als die SEDG-LBM mit dem klassischen Runge-Kutta-Verfahren. Na-

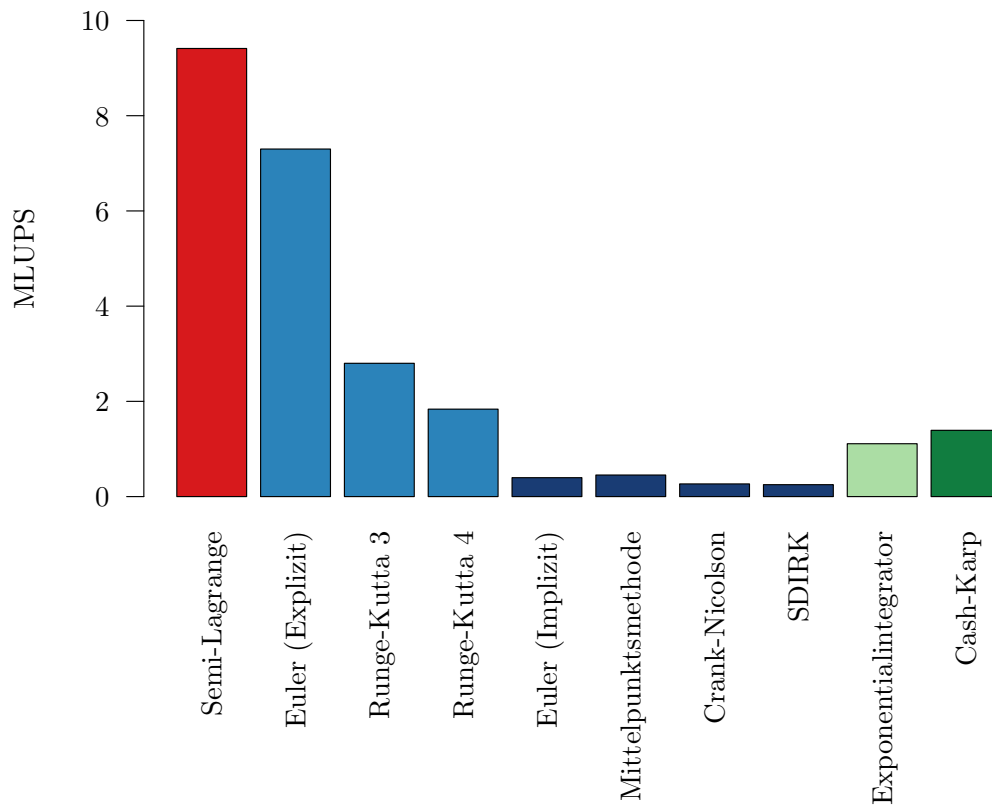


Abbildung 5.6: Laufzeitvergleich verschiedener OLBMs auf einem Clusterknoten und MPI-Parallelisierung. Die SLLBM (Semi-Lagrange) wurde verglichen mit der SEDG-LBM, die für diesen Vergleich mit verschiedenen Zeitintegratoren ausgestattet wurde. Die Performanz wird angegeben in MLUPS, wobei die Anzahl an Lattice-Knoten hier mit der Anzahl an Gitterpunkten identifiziert wird.

türlicherweise skalierte die Performanz der verschiedenen Integratoren invers mit der Anzahl an benötigten Matrix-Vektor-Multiplikationen. Das explizite Euler-Verfahren benötigt nur eine Multiplikation und kommt dabei der Performanz pro Zeitschritt der SLLBM mit 7 MLUPS sehr nahe. Allerdings sollte es im Kontext von OLBMs nicht verwendet werden, da es die zeitliche Konvergenz von OLBMs degradiert. (Für alle anderen Integratoren ist die zeitliche Konvergenz der SEDG-LBM von der Ordnung $O(\delta_t^2)$.) Der Exponentialintegrator ist ähnlich performant wie das klassische Runge-Kutta-Verfahren. Die impliziten Verfahren waren im Vergleich wesentlich langsamer, lassen dafür aber große Zeitschritte zu. Dabei ist allerdings zu beachten, dass die Performanz der impliziten Integratoren von vielen Faktoren abhängt (Zeitschrittweite, Strömung, Abbruchkriterium des linearen Löser, ...) und hier nur exemplarisch dargestellt werden kann.

Wie erwartet war die Anzahl an Gitterpunkt-Updates wesentlich kleiner als bei der Standard-LBM. So erreichte Palabos auf der gleichen Hardware etwa 140 MLUPS.

Die schlechtere Performanz von NATrium kommt durch die erhöhte algorithmische Komplexität und die damit verbundenen Speicherzugriffsmuster zustande. Wie in den Abschnitten 4.3.1 und 4.3.2 gezeigt wurde, kann aber auch eine um zwei Größenordnungen verschlechterte Performanz pro Gitterpunkt und Iteration noch akzeptabel sein, wenn durch das Verfahren der Zeitschritt vergrößert und das Gitter vergrößert werden kann.

In einem Vergleich zwischen DUKGS und der Standard-LBM findet sich ein Faktor von fünf pro Iteration für zweidimensionale Strömungen [223]. Dieser Vergleich macht jedoch keine Angaben zur Hardware oder zum verwendeten Standard-LBM-Code. Somit kann aus dieser Publikation kein objektives Maß für die Performanz von DUGKS extrahiert werden. Eine detaillierte Performanzstudie einer OLBM findet man dagegen in [158], wo die SEDG-LBM hocheffizient auf GPUs implementiert und getestet wurde. Diese Implementierung funktioniert allerdings nur für zweidimensionale Probleme mit quadratischen Zellen und kann daher nicht mit anderen OLBMs verglichen werden.

Dennoch ist klar, dass die Anzahl an Speicherzugriffen in der gegenwärtigen matrixbasierten NATrium-Implementierung für die Simulation fein-aufgelöster dreidimensionaler Strömungen problematisch ist, siehe dazu auch die Diskussion der Rechenzeit in Abschnitt 4.3.3. Es ist zu erwarten, dass eine speichereffizientere (matrixfreie) Implementierung der Strömungsschritte in NATrium die Performanz beider Advektionsoperatoren verbessern kann.

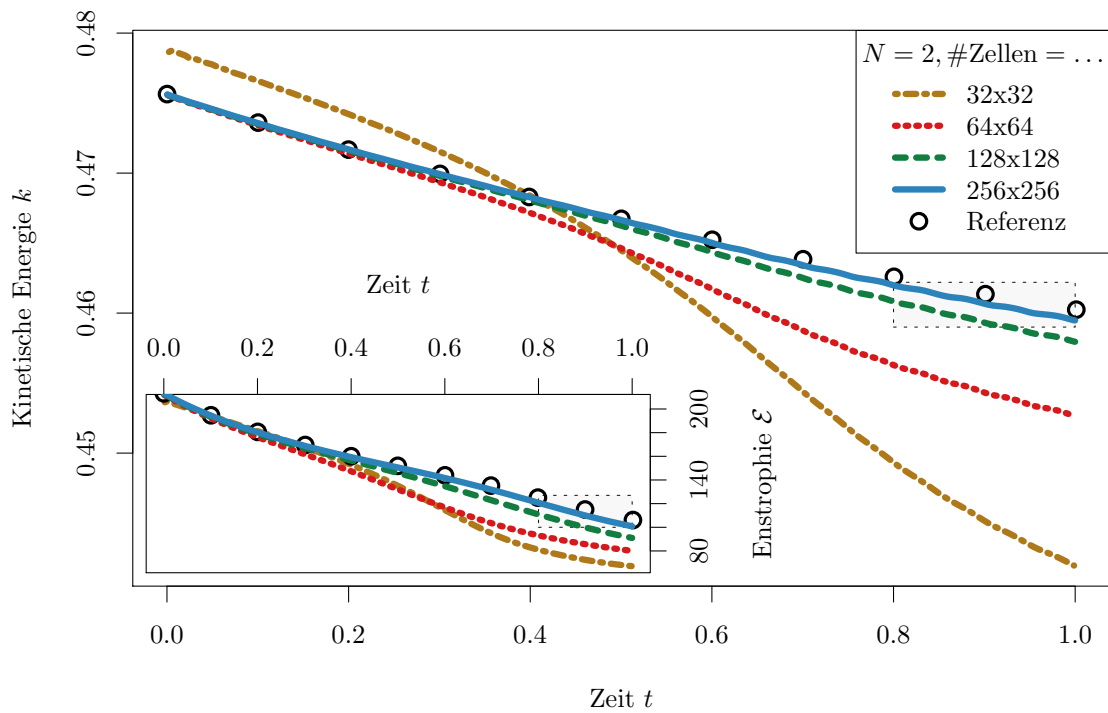
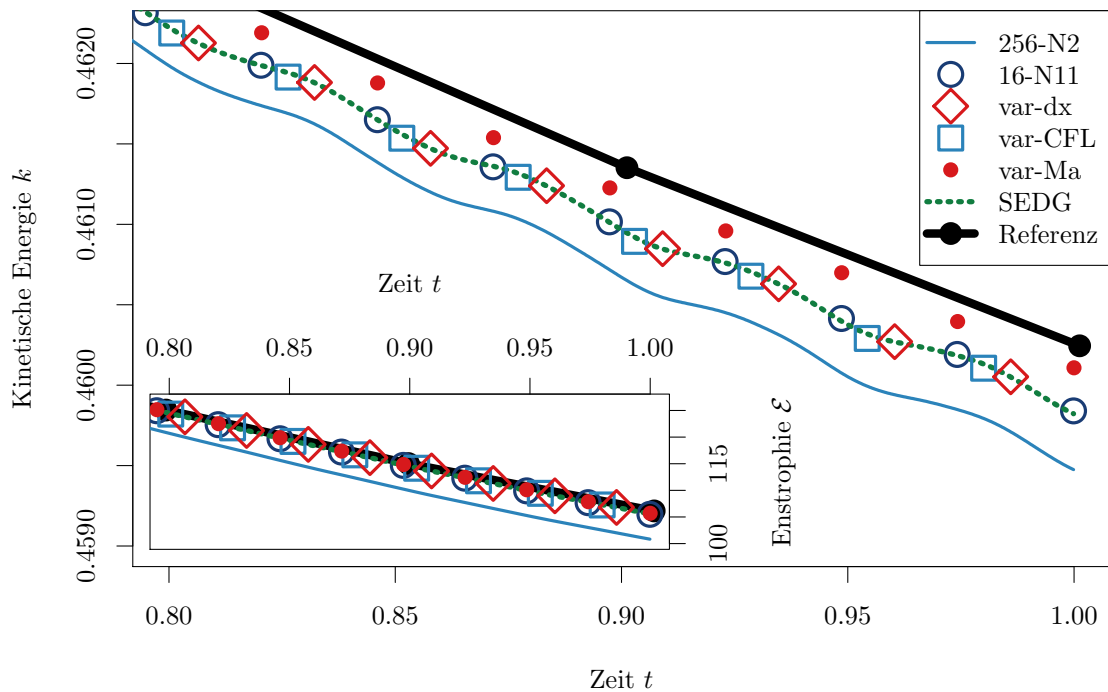
5.4.3. Zweidimensionale Scherschichten bei $Re=10\,000$

Um die SLLBM für komplexere Wirbelströmungen zu testen, wurde zunächst die zweidimensionale periodische Kelvin-Helmholtz-Instabilität aus Abschnitt 2.6.2 mit den dort beschriebenen Parametern simuliert. Dieser Fall wurde zuerst in [162] mit inkompressiblen Lösern gerechnet und analysiert. Die Literatur hierzu wurde bereits in Abschnitt 3.3.1 diskutiert, wo die Pseudo-Entropische Stabilisierung auf diesen Testfall angewandt wurde. Hier werden die kinetische Energie k und Enstrophie \mathcal{E} mit Referenzdaten verglichen (siehe Gleichung (5.4)). Diese sind definiert als

$$k = \int_0^1 \int_0^1 \frac{1}{2} u_i u_i \, dx \, dy \quad \text{und} \quad \mathcal{E} = \int_0^1 \int_0^1 \boldsymbol{\omega} \cdot \boldsymbol{\omega} \, dx \, dy \quad (5.4)$$

mit der Wirbelstärke $\boldsymbol{\omega} = \nabla \times \mathbf{u}$.

Zunächst wurden auf einem regulären Gitter Simulationen mit $N = 2$ und $CFL = 1$ durchgeführt. Abbildung 5.7a zeigt den Abfall der kinetischen Energie und Enstrophie für verschieden feine Gitter. Der Verlauf beider Größen konvergierte mit der Verfeinerung zur Referenzlösung von Minion und Brown [162].

(a) Simulationen mit Ordnung $N = 2$ auf verschieden feinen Gittern.

(b) Simulationen mit höherer Ordnung. Die Parameter und Konfigurationen sind in Tabelle 5.1 aufgelistet. 256-N2 bezeichnet die genaueste Simulation aus (a).

Abbildung 5.7: Abfall der kinetischen Energie k und der Enstrophie \mathcal{E} in Simulationen der zweidimensionalen Scherschicht bei $Re = 10\,000$ mit der SLLBM. Die Referenzlösung stammt aus [162]. Die Grenzen der unteren Abbildung sind in der oberen Abbildung gekennzeichnet.

Tabelle 5.1: Parameter und Konfigurationen für die Simulationen der zweidimensionalen Scherschichten mit Reynoldszahl $Re = 10\,000$. Die Ergebnisse zu diesen Simulationen werden in Abbildung 5.7b gezeigt.

Name	Methode	#Zellen	N	CFL	Ma
256-N2	SLLBM	256x256	2	1	$\sqrt{3}/25$
16-N11	SLLBM	16x16	11	1	$\sqrt{3}/25$
var-dx	SLLBM	32x32	11	1	$\sqrt{3}/25$
var-CFL	SLLBM	16x16	11	4	$\sqrt{3}/25$
var-Ma	SLLBM	16x16	11	1	$\sqrt{3}/37.5$
SEDG	SEDG-LBM [161]	16x16	11	1	$\sqrt{3}/25$
Referenz	Pseudo-Spektral	768x768	–	adaptiv	0

Mit einer Erhöhung der Ordnung konnte die Genauigkeit verbessert werden. Hierzu wurden verschiedene Simulationen durchgeführt, deren Parameter und Konfigurationen in Tabelle 5.1 aufgeführt sind. Abbildung 5.7b zeigt den Verlauf der kinetischen Energie und Enstrophie in der finalen Phase der Simulation. Bei einer Ordnung von $N = 11$ und einem Gitter von 16×16 wich die kinetische Energie noch leicht von der Referenzlösung ab. Durch eine Verdopplung der Gitterauflösung konnte hier keine Verbesserung mehr erzielt werden. Auch bei einer Variation der CFL-Zahl und bei der Verwendung der SEDG-LBM blieben die Energie und Enstrophie unverändert. Eine weitere Annäherung an die Referenzlösung konnte jedoch durch eine kleinere Machzahl erzielt werden. Dieses Ergebnis demonstriert, dass die räumlichen und zeitlichen Fehlerterme bei $N = 11$ und 16×16 Zellen konvergiert sind.

Abbildungen 5.8a und 5.8b zeigen die finalen Wirbelstärkefelder bei der niedrigen Ordnung $N = 2$. Hier bildeten sich bis zu einer Auflösung von bis zu 64×64 Zellen (also 128×128 Punkten) unphysikalische sekundäre Wirbel. Ab einer Auflösung von 128×128 Zellen (also 256×256 Punkten) traten keine sekundären Wirbel mehr auf. Es ist bemerkenswert, dass in [162] mit verschiedenen inkompressiblen Lösern die gleiche Grenze für das Entstehen sekundärer Wirbel gezeigt wurde.

Abbildungen 5.8c und 5.8d zeigen, dass eine Erhöhung der Ordnung auch auf groben Gittern die Entstehung sekundärer Wirbel verhindern konnte. Eine Simulation mit $N = 13$ und 8×8 Zellen reduzierte die Anzahl an Gitterpunkten (104×104) auf ein Sechstel gegenüber [162].

Um die räumliche Konvergenz zu betonen, wurde der Verlauf der Standardabweichungen von kinetischer Energie und Enstrophie untersucht. Abbildung 5.9 zeigt, dass die Standardabweichungen der Simulationen mit niedriger räumlicher Ordnung gegen die Lösungen höherer Ordnung konvergierten. Abbildung 5.10 zeigt, dass die Konvergenz in Übereinstimmung mit der Ordnung der Ansatzfunktionen quadratisch war.

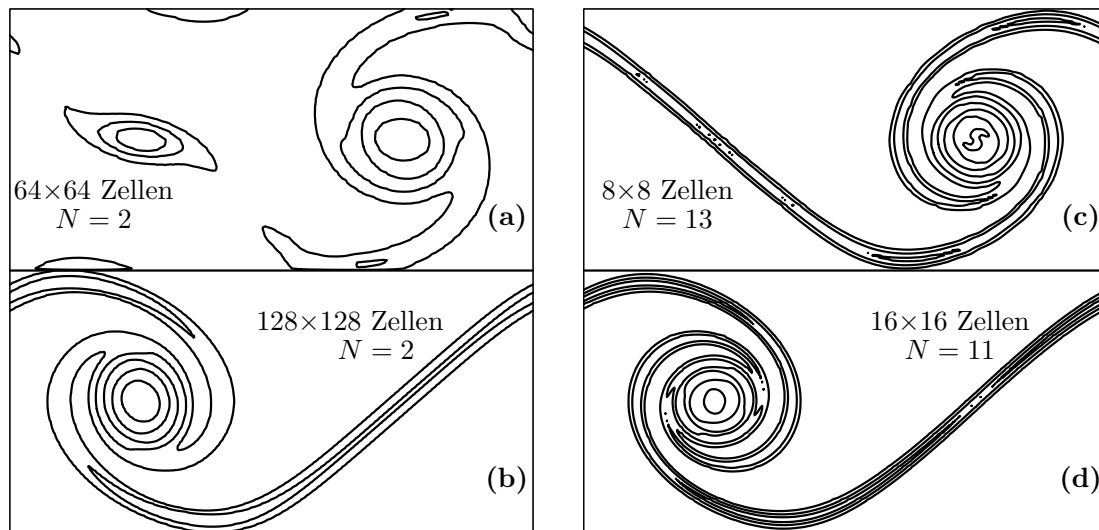


Abbildung 5.8: SLLBM-Simulationen der zweidimensionalen Scherschichten mit Reynoldszahl $Re = 10\,000$. Die Konturdiagramme zeigen die Wirbelstärke von $\{-75, -65, \dots, 65, 75\}$ zum finalen Zeitpunkt $t = 1$. Wegen der Symmetrie der Strömung wird in (a) und (c) nur die Wirbelstärke in der oberen Hälfte und in (b) und (d) nur die Wirbelstärke in der unteren Hälfte gezeigt. Die Simulationen unterschieden sich nur in der Gitterauflösung und Ordnung der Ansatzfunktionen, die in den einzelnen Diagrammen angegeben ist.

5.4.4. Zweidimensionale Scherschichten bei $Re=30\,000$

Die Simulationen von zweidimensionalen Scherschichten werden bei Erhöhung der Reynoldszahl noch herausfordernder. Im Folgenden wird daher die Scherströmung mit den Parametern aus Abschnitt 3.3.1 simuliert und mit der Referenz aus [18] verglichen. Die Referenzlösung wurde mit einer Standard-LBM mit BGK-Modell auf einem Gitter mit 256×256 Punkten erzeugt.

Zunächst wurden zwei Simulationen mit einer hohen Ordnung $N = 11$ auf Gittern mit 16×16 und 32×32 Zellen durchgeführt. Hierbei war $CFL = 10$. Um zu demonstrieren, dass die SLLBM auch auf irreguläre Gitter und dort mit großen CFL-Zahlen anwendbar ist, wurde eine Simulation mit $CFL = 30$ auf dem Gitter aus Abbildung 5.11 und $N = 2$ durchgeführt.

Das Gitter wurde mit der folgenden Transformation erzeugt

$$x = \frac{0.2 \cdot \sin(4\pi x_r) + 4\pi x_r}{4\pi}$$

$$y = \frac{0.8 \cdot \sin(4\pi y_r) + 4\pi y_r}{4\pi},$$

wobei (x_r, y_r) die Punkte eines regulären Gitters in $[0, 1]^2$ bezeichnen. Der minimale

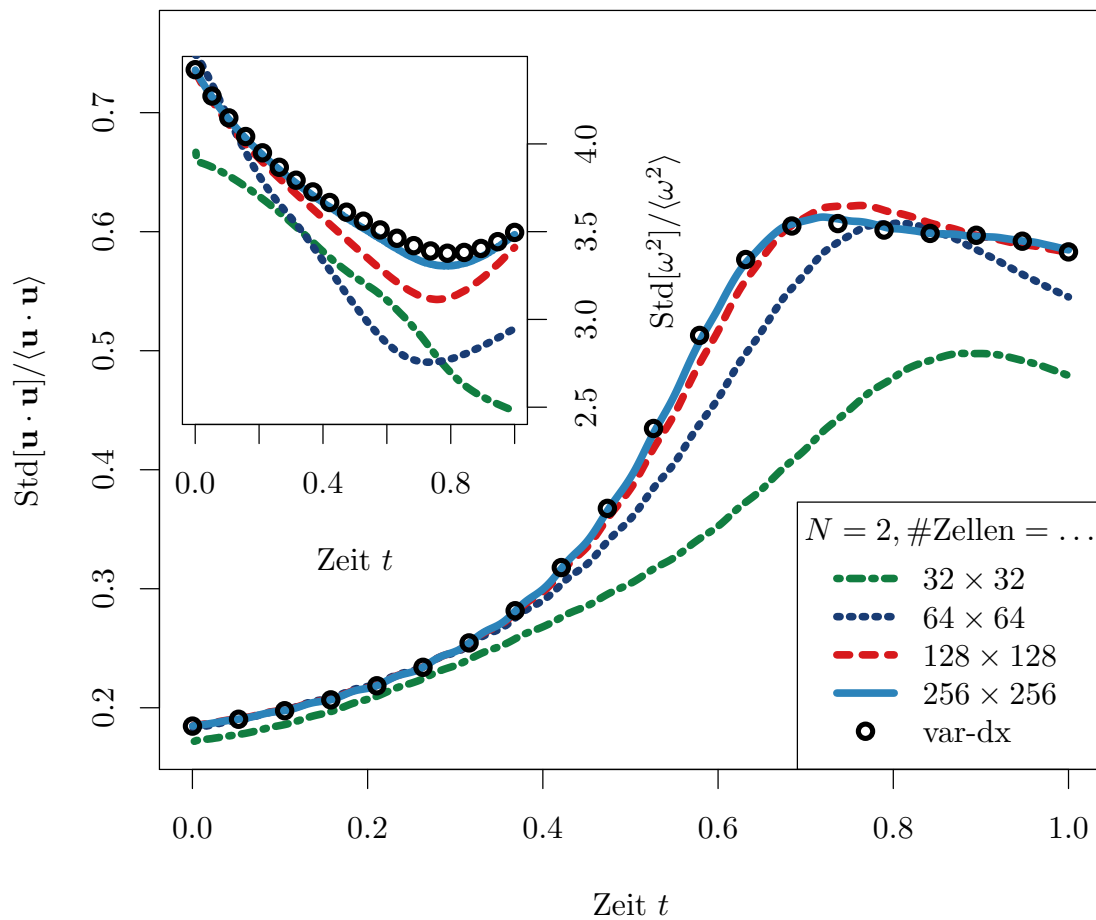


Abbildung 5.9: Standardabweichungen von kinetischer Energie und Enstrophie in Simulationen einer zweidimensionalen Scherschicht bei $Re = 10000$. “var-dx” bezeichnet die Simulation mit hoher räumlicher Auflösung aus Tabelle 5.1.

Abstand zwischen zwei Gitterpunkten war somit etwa $7.8 \cdot 10^{-4}$ und der Zeitschritt mit $CFL = 30$ mehr als fünf mal größer. Ein so großer Zeitschritt ist bei Eulerschen Verfahren nur mit impliziten Verfahren stabil.

Abbildung 5.12 zeigt den zeitlichen Verlauf von kinetischer Energie und Enstrophie sowie deren Standardabweichungen. Die Simulationen mit der hohen Ordnung waren etwas weniger dissipativ als die Referenzlösung. Dahingegen war die Simulation mit der niedrigen Ordnung auf dem irregulären Gitter etwas dissipativer. Insgesamt stimmten aber alle Ergebnisse gut mit der Referenzlösung [18] überein.

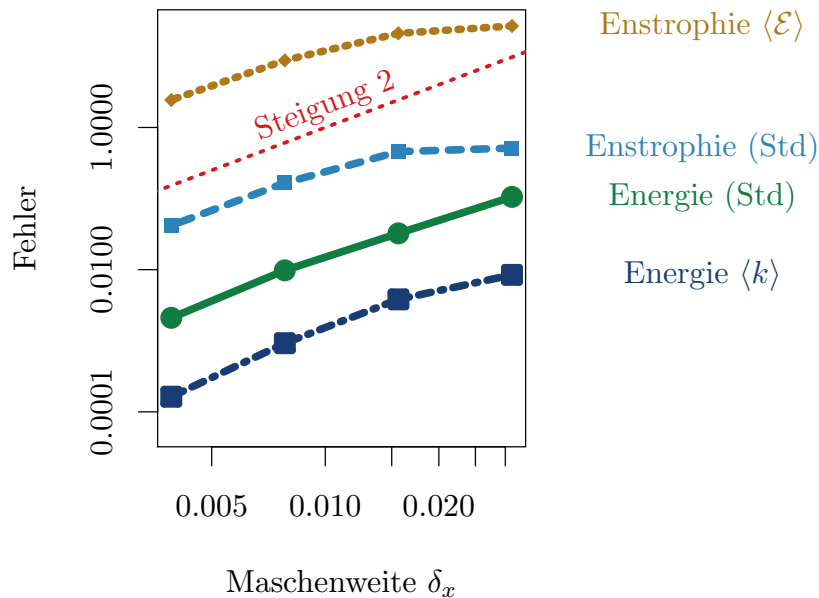


Abbildung 5.10: L_2 -Fehler der Lösungen zweiter Ordnung gegenüber “var-dx”. Die Fehler wurden berechnet als $\sqrt{\langle (\Gamma - \Gamma_{\text{var-dx}})^2 \rangle_t}$, wobei $\langle \cdot \rangle_t$ einen zeitlichen Mittelwert bezeichnet und Γ die folgenden Größen repräsentiert: k (Kinetische Energie), \mathcal{E} (Enstrophie), $\text{Std}[\mathbf{u} \cdot \mathbf{u}] / \langle \mathbf{u} \cdot \mathbf{u} \rangle$ (Energie (Std)) und $\text{Std}[\omega^2] / \langle \omega^2 \rangle$ (Enstrophie (Std)).

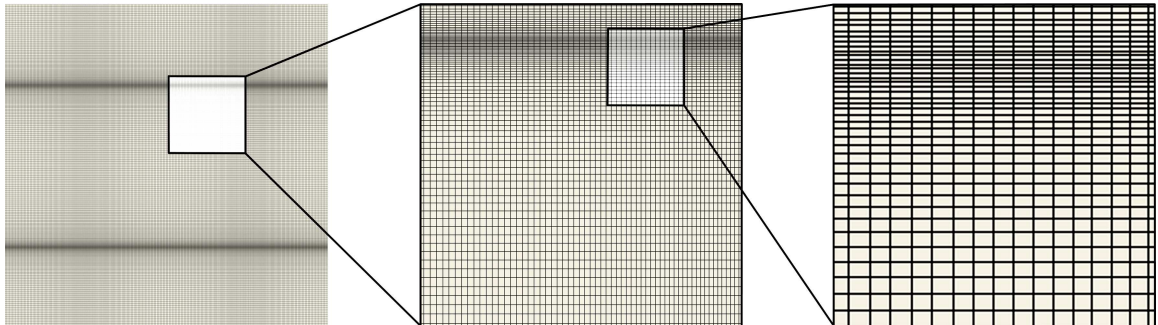
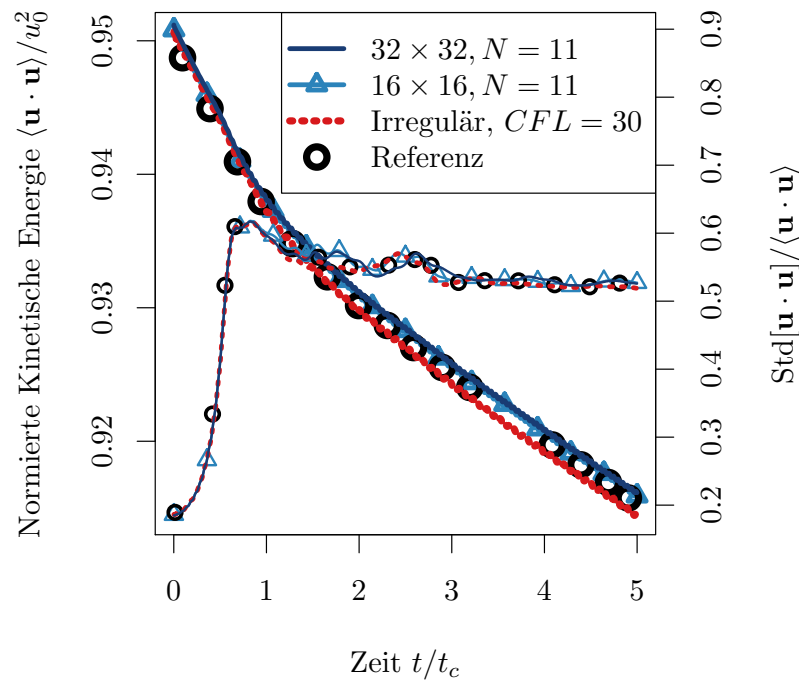


Abbildung 5.11: Irreguläres Gitter mit 256×256 Zellen. Dieses Gitter wurde in einer Simulation der zweidimensionalen Scherschicht bei $Re = 30\,000$ verwendet, um zu demonstrieren, dass die SLLBM nicht nur für reguläre Gitter verwendet werden kann.

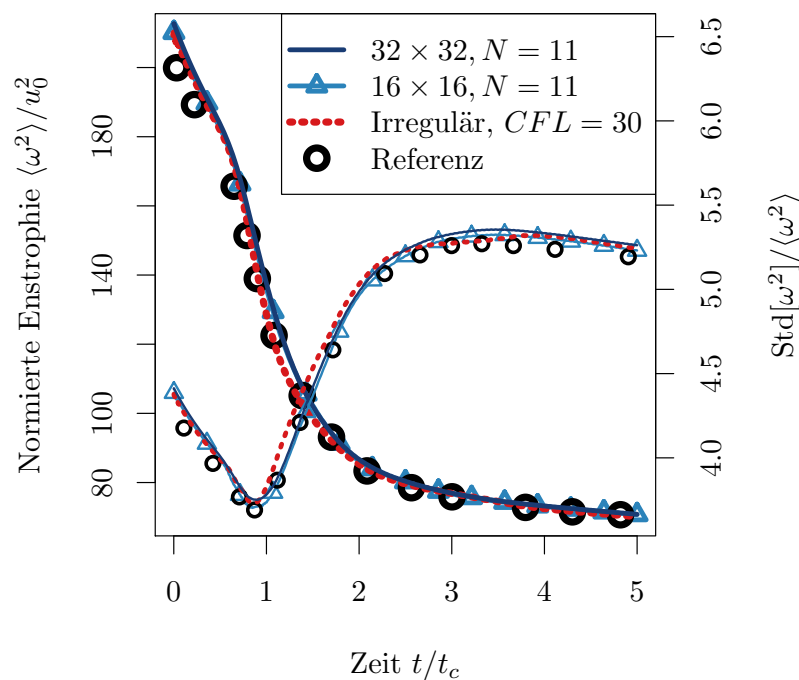
5.4.5. Dreidimensionale Taylor-Green-Wirbel bei $Re=1600$

Um das Verfahren auch für die Simulation dreidimensionaler Wirbelströmungen zu validieren, wurde der dreidimensionale Taylor-Green-Wirbel aus Abschnitt 3.3.2 mit der SLLBM gerechnet. Dabei wurde die Reynoldszahl $Re = 1\,600$ verwendet.

Abbildung 5.13 zeigt die effektive Viskosität $\nu_{\text{eff}} = -(dk/dt) / (\nu \mathcal{E})$ für Simulationen



(a) Normierte kinetische Energie $\langle \mathbf{u} \cdot \mathbf{u} \rangle / u_0^2$ und deren normierte Standardabweichung $\text{Std}[\mathbf{u} \cdot \mathbf{u}] / \langle \mathbf{u} \cdot \mathbf{u} \rangle$.



(b) Normierte Enstrophie $\langle \omega^2 \rangle / u_0^2$ und deren normierte Standardabweichung $\text{Std}[\omega^2] / \langle \omega^2 \rangle$.

Abbildung 5.12: Normierte kinetische Energie und Enstrophie in Simulationen einer zwei-dimensionalen Scherschicht mit $Re = 30\,000$. Die dicken Linien und Symbole repräsentieren die Mittelwerte. Die dünnen Linien und Symbole repräsentieren die Standardabweichungen. Das irreguläre Gitter für die Simulation bei $CFL = 30$ ist in Abbildung 5.11 gezeigt. Die Referenzlösung stammt aus [18].

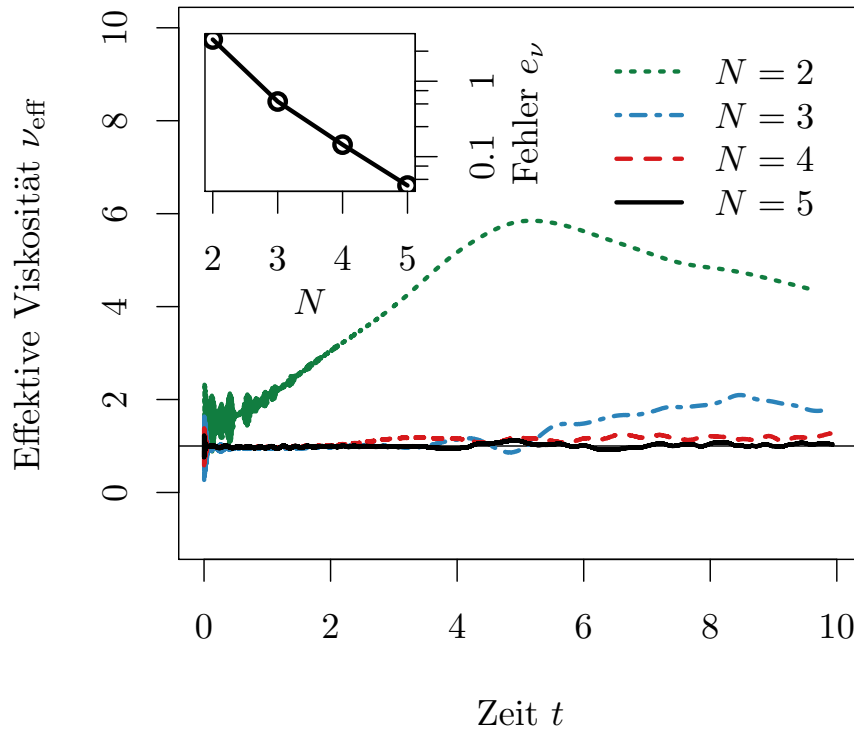


Abbildung 5.13: Effektive Viskosität $\nu_{\text{eff}} = -(dk/dt)/(\nu\mathcal{E})$ bei Simulationen eines dreidimensionalen Taylor-Green-Wirbels mit der SLLBM. Die Simulationen wurden bei einer Gitterauflösung von $32 \times 32 \times 32$ mit verschiedener Ordnung N durchgeführt. Die kleine Abbildung zeigt den gemittelten Fehler $e_\nu := \sqrt{\langle (\nu_{\text{eff}} - 1)^2 \rangle_t}$.

mit $N = 2, 3, 4, 5$ auf einem Gitter mit 32×32 Punkten. Bei einer effektiven Viskosität von $\nu_{\text{eff}} = 1$ stimmt die turbulente Dissipation mit der Ableitung der kinetischen Energie überein. Mit der Erhöhung der Ordnung konvergierte die effektive Viskosität gegen 1. Abbildung 5.13 zeigt, dass der Fehler mit der Ordnung der Ansatzfunktionen exponentiell konvergierte.

Abbildung 5.14 zeigt die Dissipation für die höchste Ordnung $N = 5$. Die Dissipation wurde zunächst über finite Differenzen der kinetischen Energie gemessen, wobei ein Inkrement von 100 Zeitschritten verwendet wurde. Dieser Ausdruck stimmte über den gesamten Simulationslauf gut mit der turbulenten Dissipation ϵ und der skalierten Enstrophie $\nu\mathcal{E}$ überein. Die nahezu exakte Übereinstimmung der beiden letzteren Größen lässt sich analytisch aus der inkompressiblen Kontinuitätsgleichung herleiten und zeigt damit, dass die künstliche Kompressibilität nahezu keinen Einfluss auf die Ergebnisse hatte. Die numerische Dissipation, also die Abweichung von $-dk/dt$ und ϵ , schwankte um Null, was zeigt, dass alle relevanten Skalen der Strömung in der Lösung repräsentiert wurden. Die berechneten Dissipationsraten stimmten insgesamt gut mit der Referenzlösung [21] überein.

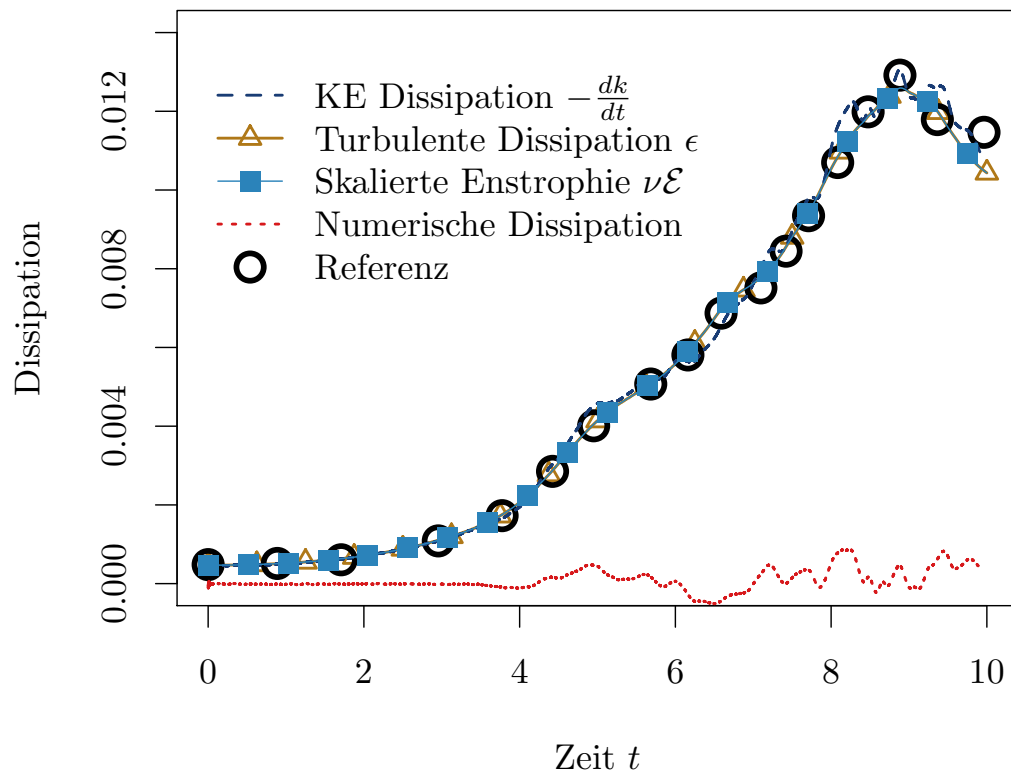


Abbildung 5.14: Simulation eines dreidimensionalen Taylor-Green-Wirbels bei $Re = 1600$ mit der SLLBM. Die Abbildung zeigt verschiedene Ausdrücke für die Dissipation ($-dk/dt$, ϵ und $\nu\mathcal{E}$) im Vergleich mit der Referenz [21] sowie die numerische Dissipation $-dk/dt - \epsilon$.

5.4.6. Vergleich mit einer Finite-Volumen-Lattice-Boltzmann-Methode

Im Folgenden wird der dreidimensionale Taylor-Green-Wirbel bei einer Reynoldszahl von $Re = 300$ simuliert, um die SLLBM mit DUGKS zu vergleichen. Die Finite-Volumen-LBM DUGKS hat in den letzten Jahren einige Aufmerksamkeit erregt [14, 79, 80, 223, 241]. In [14] wurde mit DUGKS ein Taylor-Green-Wirbel bei $Re = 300$ simuliert und mit einem Pseudo-Spektral-Code [58, 220] und der Standard-LBM verglichen.

Die Simulationen dort wurden auf einem Knoten des Yellowstone-Clusters am NCAR-Wyoming Supercomputing Center gerechnet. Jeder Knoten dort ist mit zwei Prozessoren des Typs Intel Xeon E5-2670 (Sandy Bridge) ausgestattet. Um eine Vergleichbarkeit der Laufzeiten zu gewährleisten, wurden die in diesem Unterabschnitt beschriebenen Simulationen auf dem WR11-Cluster der Hochschule Bonn-Rhein-Sieg gerechnet, der mit den gleichen Prozessoren ausgestattet ist.

Tabelle 5.2: Vergleich der Rechenzeiten der SLLBM mit [14]. Die Rechenzeiten beziehen sich auf die Simulation eines dreidimensionalen Taylor-Green-Wirbels über das Zeitintervall $t \in [0, 5]$ auf 16 Prozessorkernen [219]. Die Zeitschritte aus [14] wurden auf die hier verwendete charakteristische Geschwindigkeit von $u_0 = 1$ skaliert. Die CPU-Zeit bezeichnet die Rechenzeit multipliziert mit der Anzahl an MPI-Prozessen.

Methoden	Gitter	δ_t	# Zeitschritte	Rechenzeit pro Zeitschritt [s]	CPU-Zeit [s]
Pseudo-Spektral [14]	128^3	0.005	1 000	0.185	2 969
Standard-LBM [14]	128^3	0.005	1 000	0.782	12 519
DUGKS [14]	128^3	0.0025	2 000	0.939	30 052
SLLBM ($N = 2$)	128^3	0.0025	2 000	1.468	46 970
SLLBM ($N = 2$)	128^3	0.005	1 000	1.596	25 534
SLLBM ($N = 2$)	128^3	0.009	555	1.880	16 698
SLLBM ($N = 4$)	64^3	0.002	2 500	0.440	17 591
SLLBM ($N = 4$)	64^3	0.005	1 000	0.471	7 554
SLLBM ($N = 4$)	64^3	0.015	333	0.523	2 786
SLLBM ($N = 3$)	96^3	0.0013	3 750	0.836	50 180
SLLBM ($N = 3$)	96^3	0.01	500	0.996	7 968

Außerdem wurden Gauß-Lobatto-Tschebyscheff-Punkte bis zum Grad $N = 4$ als Stützstellen für die Interpolation verwendet. Wie in Unterabschnitt 5.3.3 erläutert, erlaubt diese Wahl stabile Simulationen auf unstrukturierten Gittern ($N = 4$ mit dem oben angesprochenen Vorbehalt). Auch in dieser Hinsicht sind DUGKS und die SLLBM somit vergleichbar, wengleich alle Simulationen für die folgende Studie auf regulären Gittern gerechnet wurden.

5.4.6.1. Simulationen zweiter Ordnung

Zunächst wurde die SLLBM mit Ansatzfunktionen zweiter Ordnung verwendet. Tabelle 5.2 zeigt die Laufzeiten aus [14] im Vergleich zu Laufzeiten der SLLBM in Natrium. Bei gleichem Zeitschritt und gleicher Auflösung war die SLLBM etwa 1.5 mal teurer als DUGKS, etwa doppelt so teuer wie die Standard-LBM und eine Größenordnung teurer als der Pseudo-Spektral-Code. Dabei ist zu beachten, dass letzterer hocheffizient implementiert ist, wobei die anderen Verfahren mit eher prototypischen Implementierungen gerechnet wurden.

Insgesamt stieg die Rechenzeit pro Zeitschritt in der SLLBM mit der Größe des Zeitschritts leicht an. Eine Vergrößerung des Zeitschritts bewirkt hier, dass die Ver-

teilungsfunktionen innerhalb eines Zeitschritts über eine größere Distanz advehiert werden. Dies erfordert zum einen mehr Kommunikation zwischen den MPI-Prozessen und zum anderen liegen die zur Interpolation benötigten Freiheitsgrade im Speicher weiter auseinander, sodass der Cache generell ‐kälter‐ ist.

Die SLLBM mit $N = 2$ und $\delta_t = 0.5$ ist äquivalent zur Standard-LBM. In den beiden anderen Simulationen wurde der Zeitschritt einmal halbiert, um konsistent mit DUGKS zu sein, und einmal fast verdoppelt. (Eine exakte Verdopplung wäre nicht empfehlenswert, da dann nur Verteilungsfunktionen auf jedem zweiten Gitterpunkt miteinander gekoppelt wären.)

Abbildung 5.15 (oben) zeigt die normierten Dissipationsraten, wobei alle Simulationen in der Abbildung – auch die der Referenz – auf 128^3 Gitterpunkten gerechnet wurden. Die Lösung mit dem Pseudo-Spektral-Code war bei einer Auflösung von 128^3 Punkten konvergiert und wird daher als die physikalisch korrekte Lösung angesehen.

Die Simulation mit der SLLBM und $\delta_t = 0.005$ stimmt wie erwartet mit der Standard-LBM überein. Sowohl die Vergrößerung als auch die Verkleinerung des Zeitschritts vergrößerten den numerischen Fehler. Interessanterweise war die Verkleinerung des Zeitschritts dabei kritischer. Dies hängt mit der in Unterabschnitt 5.4.1 besprochenen Fehlerordnung von $O(\delta_x^N + \delta_x^N/\delta_t)$ zusammen. Im Zusammenspiel mit der niedrigen räumlichen Ordnung führt der kleine Zeitschritt somit zu einer vergleichsweise großen numerischen Dissipation.

Die Simulation mit $\delta_t = 0.009$ lief stabil, obwohl der Zeitschritt bei gleicher Auflösung fast viermal so groß war wie bei DUGKS. Darüber hinaus waren die Fehler in der Dissipation nahezu identisch. Sogar mit $N = 2$ war es somit möglich, die Gesamtlaufzeit gegenüber DUGKS fast auf die Hälfte zu reduzieren, ohne das Ergebnis zu verschlechtern.

5.4.6.2. Simulationen höherer Ordnung

Als nächstes wurde der Taylor-Green-Wirbel mit $N = 4$ auf 64^3 Gitterpunkten simuliert. Auch in der Referenz waren Ergebnisse mit dieser Auflösung angegeben. Die Rechenzeiten in Tabelle 5.2 demonstrieren, dass der Zeitschritt durch die höhere Ordnung nicht eingeschränkt wurde. Durch die Verwendung der höheren Ordnung auf dem gröberen Gitter konnte die Gesamtlaufzeit insgesamt wesentlich reduziert werden. Dennoch wurde die Dissipationsrate ähnlich gut reproduziert wie auf dem feineren Gitter.

Abbildung 5.15 (mitte) zeigt, dass mit einem kleinen Zeitschritt $\delta_t = 0.002$ die Dissipationsrate genauer wiedergegeben wurde als mit der Standard-LBM (bei gleicher räumlicher Auflösung). Insbesondere war mit der höheren Ordnung keine Vergrößerung des Fehlers bei kleinen Zeitschritten zu erkennen. Gegenüber DUGKS konnte der Zeitschritt bei ungefähr gleicher Dissipation verdreifacht werden.

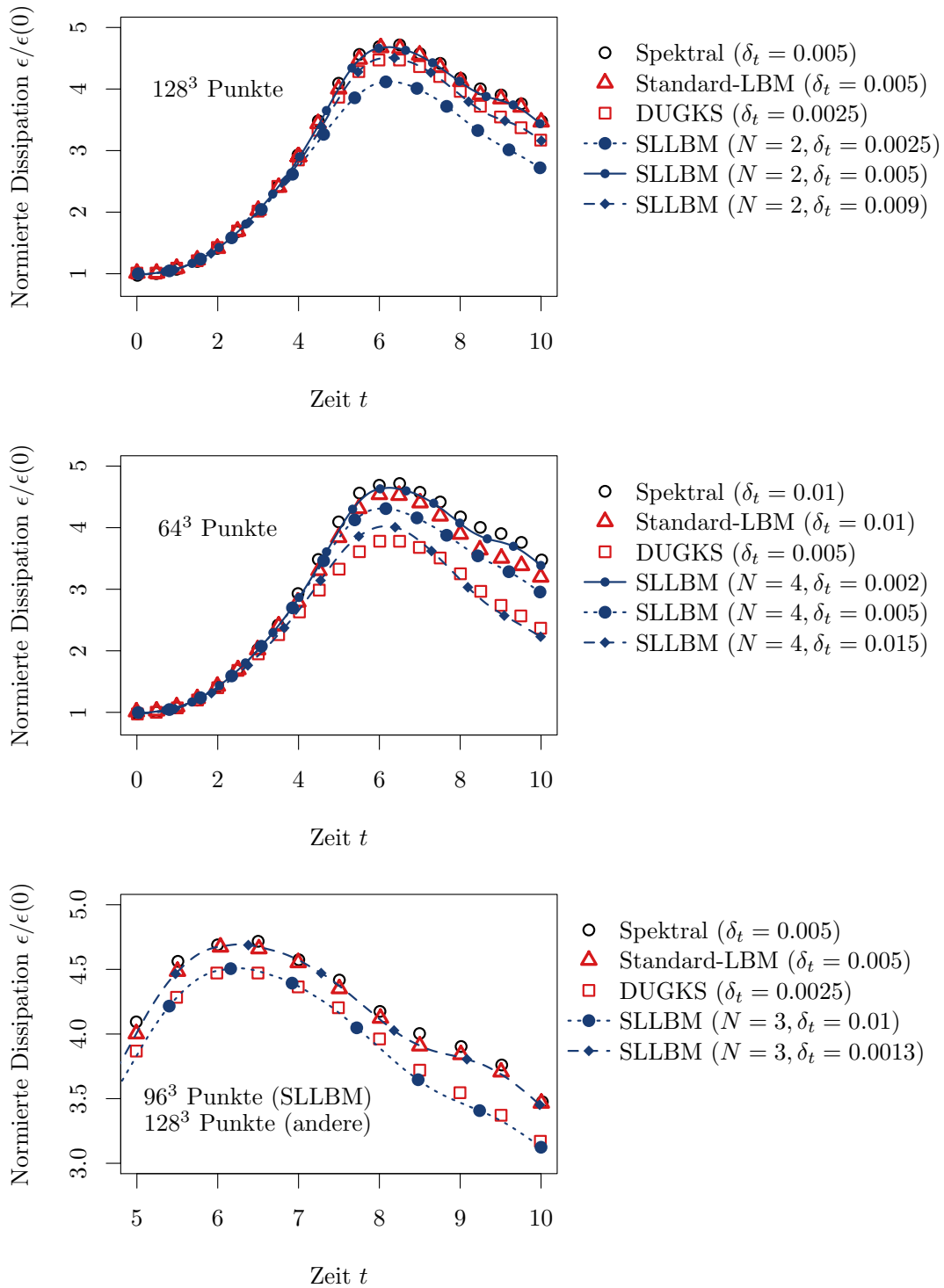


Abbildung 5.15: Normierte Dissipationsraten $\epsilon/\epsilon(0)$ in Simulationen eines dreidimensionalen Taylor-Green-Wirbels bei einer Reynoldszahl von $Re = 300$. Die Ergebnisse für den Pseudo-Spektral-Code, die Standard-LBM und DUGKS stammen aus [14]. Dort wurden die normierten Dissipationsraten für 64^3 und 128^3 angegeben. Die Auflösungen sind in den Abbildungen jeweils angegeben. Die Ordnung der SLLBM war $N = 2$ (oben), $N = 4$ (Mitte) und $N = 3$ (unten). Die initiale Dissipationsrate ist $\epsilon(0) = 0.0025$.

Für $N = 3$ wurde ein Gitter mit 96^3 Punkten verwendet, um zu untersuchen, ob die Rechenzeit durch die Verwendung einer niedrigeren Auflösung weiter gesenkt werden kann. Abbildung 5.15 (unten) zeigt, dass mit einem kleinen Zeitschritt von $\delta_t = 0.0025$ ein ähnlich gutes Ergebnis erzielt werden konnte wie mit der Standard-LBM und 128^3 Punkten. Gegenüber DUGKS waren die Simulationen mit einem viermal größeren Zeitschritt noch stabil und lieferten sehr ähnliche Dissipationsraten. So konnte die Simulationszeit nahezu geviertelt werden (siehe Tabelle 5.2).

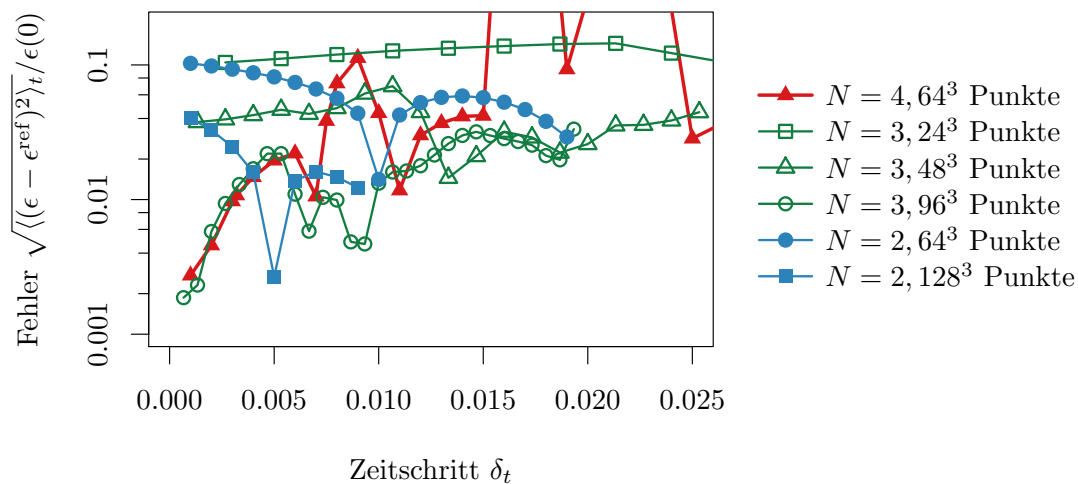


Abbildung 5.16: Zeitlich gemittelte Fehler in der Dissipationsrate für die SLLBM gegenüber der Standard-LBM mit 256^3 Gitterpunkten. Die Abbildung zeigt die Fehler in Abhängigkeit des Zeitschritts δ_t und für verschiedene Ordnungen N und Gitterauflösungen.

Abbildung 5.16 zeigt schließlich den Fehler in der Dissipationsrate für verschiedene Ordnungen, Gitterauflösungen und Zeitschritte. Bei Interpolationen zweiter Ordnung wird jeweils ein globales Minimum bei $\delta_t \xi_0 = \delta_x$ angenommen, wo die SLLBM äquivalent zur Standard-LBM ist. Für Interpolationen dritter und vierter Ordnung gibt es keine Zeitschritte, die auf eine äquivalente Standard-LBM führen. Wenn das Gitter hier fein genug aufgelöst war, konnten mit kleinen Zeitschritten jedoch genauere Simulationen durchgeführt werden als mit der Standard-LBM. Während die Genauigkeit für $N = 4$ stark vom Zeitschritt abhing, war die Variation der Fehler für $N = 3$ kleiner. Insgesamt bot $N = 3$ über einen vergleichsweise großen Bereich von Zeitschritten eine gute Genauigkeit.

5.5. Diskussion der Semi-Lagrangeschen Lattice-Boltzmann-Methode

Im Folgenden werden die wesentlichen Eigenschaften der SLLBM noch einmal zusammengefasst und diskutiert.

5.5.1. Konvergenz

Die theoretische globale Fehlerordnung der SLLBM im Hinblick auf die inkompressiblen Navier-Stokes-Gleichungen ist

$$E_{\text{SLLBM}} = O(\delta_x^N + \delta_x^N / \delta_t + \delta_t^2 + Ma^2).$$

Diese Konvergenzordnung kombiniert sich aus dem Kompressibilitätsfehler und zeitlichen Fehler der Standard-LBM und der globalen Ordnung des Semi-Lagrangeschen Strömungsschritts. Dieses Konvergenzverhalten wurde in allen Simulationen in diesem Abschnitt bestätigt. Die hohe räumliche Ordnung stimmt mit den Ergebnissen von [54, 95, 97, 161] überein, wo die Konvergenzordnung des Advektionsoperators auch die Konvergenzordnung der OLBM vorgibt. Somit ist die SLLBM die erste interpolationsbasierte OLBM, für die eine hohe räumliche Konvergenzordnung nachgewiesen werden konnte.

5.5.2. Stabilität

Gegenüber den meisten OLBMs ist der Strömungsschritt der SLLBM für die Ordnung $N = 2$ bedingungslos stabil. Insbesondere muss keine CFL-Bedingung eingehalten werden. Bei der Verwendung von Gauß-Lobatto-Tschebyscheff-Punkten wird sogar für $N = 3$ unbedingte Stabilität erreicht und auch für $N = 4$ ist der Spektralradius für fast alle Zeitschritte ≤ 1 . Die unbedingte Stabilität gilt dabei sowohl für reguläre als auch für irreguläre Gitter.

Für höhere Ordnungen lassen sich auf regulären Gittern hinreichende Bedingungen für die Stabilität herleiten (siehe Unterabschnitt 5.3.3). Hier hängt die Stabilität nichtlinear vom Zeitschritt ab.

5.5.3. Performanz

Die Performanz der SLLBM wurde mit zwei anderen OLBMs verglichen. Gegenüber der in NATrium implementierten SEDG-LBM (mit klassischem Runge-Kutta-Verfahren) [161] wurde die Laufzeit pro Zeitschritt um einen Faktor fünf verbessert. Gegenüber impliziten Integratoren ist dieser Faktor sogar größer als zehn.

Im Vergleich zu DUGKS war die SLLBM ungefähr um einen Faktor 1.5 langsamer. Dies konnte jedoch durch die Verwendung von Ansatzfunktionen höherer Ordnung und größeren Zeitschritten mehr als kompensiert werden. So wurde die CPU-Zeit bei der Simulation eines dreidimensionalen Taylor-Green-Wirbels gegenüber DUGKS um den Faktor vier verringert, obwohl die numerischen Dissipationsraten beider Verfahren ähnlich waren.

Verglichen mit dem Standard-LBM-Code Palabos (siehe Unterabschnitt 5.4.2) und einem Pseudo-Spektral-Code (siehe Unterabschnitt 5.4.6) war die SLLBM um eine Größenordnung langsamer. Dabei ist jedoch zu berücksichtigen, dass die NATrium-Implementierung der SLLBM nicht optimiert ist. So wurde im Kollisionsschritt keine Vektorisierung und insgesamt keine *Shared-Memory*-Parallelisierung verwendet. Aufwendig ist auch die matrixbasierte Implementierung des Advektionsschritts. Diese setzt nicht nur einen großen Arbeitsspeicher voraus, sondern benötigt auch sehr viele Speicherzugriffe. So muss jedes Nicht-Null-Element der dünnbesetzten Advektionsmatrizen einmal pro Zeitschritt ausgelesen werden. Hier ist durch eine matrixfreie Implementierung eine Steigerung der Performanz zu erwarten.

5.5.4. Numerische Dissipation

Durch Interpolationen höherer Ordnung ($N > 2$) kann die numerische Dissipation der SLLBM gering gehalten werden. Für kleine Zeitschritte war die numerische Dissipation sogar kleiner als mit der Standard-LBM. Dieses Ergebnis ist besonders interessant, da interpolationsbasierte LBMs in der Vergangenheit wegen ihrer dissipativen Eigenschaften kritisiert wurden [135]. Obwohl DUGKS gegenüber anderen Finite-Volumen-LBMs nur wenig numerische Dissipation erzeugt [241], war die SLLBM in dieser Hinsicht selbst mit wesentlich größeren Zeitschritten weniger dissipativ als DUGKS.

6. Zusammenfassung und Ausblick

Ziel dieser Arbeit war die effiziente methodische Weiterentwicklung der Lattice-Boltzmann-Methode (LBM) im Hinblick auf Simulationen inkompressibler Wirbelströmungen. Dabei wurden die zwei Hauptprobleme der Standard-LBM durch neuartige methodische Erweiterungen behoben: ihre Instabilität bei unteraufgelösten turbulenten Strömungen und ihre Einschränkung auf reguläre Gitter.

Dazu wurden im Wesentlichen zwei neue Methoden entwickelt. Während die *Pseudo-Entropische Stabilisierung* (PES) stabile Simulationen unteraufgelöster Turbulenz erlaubt, ermöglicht die *Semi-Lagrangesche LBM* (SLLBM) effiziente Simulationen auf irregulären Gittern. Die entwickelten Methoden und wesentlichen Erkenntnisse dieser Arbeit werden im Folgenden zusammengefasst.

Formulierung stabiler Kollisionsmodelle

Die Standard-LBM mit Bhatnagar-Gross-Krook (BGK)-Kollisionsmodell führt bei unteraufgelösten turbulenten Strömungen oft zu instabilen Simulationen, da bei dieser Konfiguration sowohl der Strömungsschritt als auch der Kollisionsschritt kaum numerische Dissipation erzeugt. Die meisten Modelle aus der Literatur wirken dieser Instabilität entgegen, indem sie entweder verschiedene Observablen (Momente) mit verschiedenen Raten zum Gleichgewicht relaxieren (*Multiple-Relaxation-Time* (MRT)-Modelle) [42, 47, 71, 74, 135] oder die Relaxation durch eine Entropiefunktion kontrollieren [118].

In der vorliegenden Arbeit wurden diese beiden Konzepte zu einer Pseudo-Entropischen Stabilisierung kombiniert. Die Herleitung dieser Methode knüpft an ein vor Kurzem veröffentlichtes entropisches MRT-Modell (KBC, [117]) an. Durch die Verwendung einer quadratischen Pseudo-Entropiefunktion konnte die PES als lokaler linearer Stabilisierungsoperator formuliert werden, der flexibel mit anderen Kollisionsmodellen kombinierbar ist.

Eine praktische Auswertung zeigte die exzellenten Stabilitätseigenschaften der PES.

Bei Simulationen einer zweidimensionalen Scherschicht mit einer Reynoldszahl von $Re = 30\,000$ waren die größten Strukturen selbst bei einer Auflösung von 16×16 Gitterpunkten noch klar zu erkennen. Im Gegensatz dazu waren die Simulationen mit dem BGK-Modell erst ab einer Auflösung von 256×256 Punkten stabil. Während in [18] mit einer Auflösung von 128×128 Punkten bei drei von vier KBC-Varianten und bei der Entropischen LBM unphysikalische Wirbel auftraten, waren die Simulationen mit der PES selbst bei einer Auflösung von 16×16 Punkten frei von sekundären Wirbeln.

In Simulationen eines dreidimensionalen Taylor-Green-Wirbels auf einem Gitter mit $80 \times 80 \times 80$ Punkten war das BGK-Modell ab einer Reynoldszahl von $Re \geq 1600$ instabil. Die PES dagegen war noch für $Re \geq 15\,000$ stabil und konnte auch in unteraufgelösten Simulationen den Verlauf der kinetischen Energie qualitativ vorhersagen. Eine Analyse der numerischen Viskosität zeigte die Funktionsweise der Stabilisierung: Durch die lokale Maximierung der Pseudo-Entropie wurden die höheren Moden und somit die Gradienten der Strömung geglättet. Die dadurch erzeugte numerische Dissipation war in guter Übereinstimmung mit der physikalischen Dissipation der nicht aufgelösten Skalen. Somit fungierte die PES in diesen Simulationen effektiv als lokales, adaptives *Subgrid*-Modell.

Im theoretischen Vergleich mit anderen Kollisionsmodellen wurden zunächst einige Vorteile gegenüber existierenden Ansätzen herausgestellt. Im Vergleich zur Entropischen LBM umgeht die PES die iterative Lösung einer nichtlinearen Gleichung durch die quadratische Approximation der Entropie. Im Gegensatz zu MRT-Modellen und ihren Derivaten besitzt das Modell keine frei wählbaren Parameter. Stattdessen werden die nicht-hydrodynamischen Momente zur Stabilisierung der Simulation an die hydrodynamischen Momente adaptiert. Anders als KBC entkoppelt die PES die höheren Momente voneinander, wodurch die Methode nicht mehr von der konkreten Wahl der Momentenbasis abhängt.

Besonders interessant ist aber der Vergleich mit der Regularisierten LBM [139]. Es konnte gezeigt werden, dass die beiden Methoden trotz unterschiedlicher Herleitungen und Formulierungen äquivalent sind. Diese Äquivalenz liefert eine entropiegestützte Interpretation für die guten Stabilitätseigenschaften der Regularisierten LBM, da die Maximierung der Pseudo-Entropie konstant zur Randomisierung der Verteilungsfunktionen und somit zur Stabilisierung beiträgt. Weil die Regularisierte LBM sich als MRT-Modell in einer Hermite-induzierten Momentenbasis formulieren lässt, überträgt sich diese Interpretation direkt auf MRT-Modelle und unterstützt eine mögliche Überlegenheit von Hermite-induzierten MRT-Basen [41–43]: Während eine volle Relaxation der höheren Momente in Hermite-Basen somit eine Randomisierung auf Grundlage einer quadratischen Pseudo-Entropie bewirkt, existiert für die standardmäßig verwendeten kartesischen Basen [47, 135] keine solche Interpretation.

Schlussendlich verifiziert die Äquivalenz mit der Regularisierten LBM auch den pseudo-entropischen Ansatz aufgrund der vielen Simulationen, die bereits mit ersterer

durchgeführt wurden, siehe zum Beispiel [11, 139, 166, 167]. Ein praktischer Grund für die Verwendung der hier hergeleiteten Operatoren ist ihre Unabhängigkeit von der Wahl der Gleichgewichtsverteilung und von der Relaxation der Schermomente. Dadurch kann die PES als Matrix-Vektor-Multiplikation nach dem Kollisionsschritt ohne eine Neuformulierung mit anderen Kollisionsmodellen kombiniert werden. Mögliche Anwendungsszenarien sind beispielsweise die Stabilisierung von Kollisionsmodellen für stationäre Probleme [82] oder kraftgetriebene und Mehrphasen-Strömungen [194]. Darüber hinaus kann der pseudo-entropische Ansatz weiter verfeinert werden, zum Beispiel durch die Verwendung besserer Entropiefunktionen oder durch eine adaptive Stabilisierung.

Untersuchung der Spektralelement-Discontinuous-Galerkin-Lattice-Boltzmann-Methode

Das zweite betrachtete Problem der Standard-LBM war die Einschränkung der Methode auf kartesische Rechengitter, die insbesondere für wandgebundene Strömungen nachteilig ist. Aufbauend auf einer existierenden Spektralelement-Discontinuous-Galerkin (SEDG)-LBM wurde gezeigt, wie diese Restriktion durch *Off-Lattice-Boltzmann-Methoden* (OLBMs) behoben werden kann. Dies führte zu einer deutlichen Reduktion der Rechenzeit bei der Simulation von Scherströmungen in sinusförmigen Geometrien. Anhand der Simulation einer turbulenten Kanalströmung wurden jedoch drastische Schwächen der existierenden OLBMs-Ansätze deutlich. Selbst nach einer Erweiterung des Ansatzes um stabilere Zeitintegratoren verbot der hohe numerische Aufwand des SEDG-Strömungsschritts die Simulation fein aufgelöster Strömungen. Dieser hohe Aufwand ergab sich aus drei Faktoren:

- 1) **Große Advektionsoperatoren:** Eulersche OLBMs lösen in jedem Zeitschritt $Q - 1$ Advektionsgleichungen. Dazu müssen in jedem Strömungsschritt die Gradienten aller $Q - 1$ Verteilungen numerisch berechnet werden. Diese Berechnung erfordert den mehrmaligen Zugriff auf alle Verteilungsfunktionen und – falls der Operator explizit abgespeichert wird – den Zugriff auf alle Matrixelemente in jedem Iterationsschritt. Aufgrund der limitierten Speicherbandbreite ist dies mit einem hohen Aufwand verbunden.
- 2) **Mehrmalige Anwendung der Advektionsoperatoren:** Die meisten Eulerschen OLBMs nutzen zur Zeitintegration explizite Runge-Kutta-Verfahren mindestens zweiter Ordnung. In diesem Zusammenhang müssen die Ableitungsoperatoren pro Iteration mehrfach ausgewertet werden. Die Verwendung impliziter Zeitintegratoren erfordert sogar eine unbestimmte Anzahl von Auswertungen. Der damit verbundene Aufwand ist für transiente Probleme im Allgemeinen nicht tragbar.

- 3) **Kleine Zeitschritte:** Explizite Zeitintegratoren sind nur für kleine Zeitschritte stabil. Die kritische CFL-Zahl der Advektionsgleichung hängt jedoch nicht von der Fluidgeschwindigkeit, sondern von der maximalen Teilchengeschwindigkeit ab, die die Fluidgeschwindigkeit um eine Größenordnung übersteigt. Somit ist die CFL-Bedingung wesentlich kritischer als in klassischen Navier-Stokes-Lösern. Die Verwendung von feinen Diskretisierungen oder Verfahren höherer Ordnung schränkt den Zeitschritt weiter ein, sodass die Zeitschrittweite in den Simulationen mit der SEDG-LBM nicht mehr durch die physikalisch relevanten Zeitskalen der Strömung, sondern nur noch durch numerische Aspekte des Advektionslösers motiviert war. Die CFL-Bedingung ist somit der kritischste Aspekt der Eulerschen OLBMs.

Diese Faktoren sind universelle Probleme der Eulerschen OLBMs [14, 142]. Dennoch wurde der Eulersche Ansatz in vielen Veröffentlichungen verfolgt. Aufgrund der genannten Schwierigkeiten ist jedoch kaum zu erwarten, dass sich diese als echte Alternative zur Standard-LBM etablieren können.

Entwicklung einer Semi-Lagrangeschen Lattice-Boltzmann-Methode

Basierend auf diesen Erkenntnissen wurde eine neuartige Methode entwickelt, die auf einer Semi-Lagrangeschen Modellierung des Strömungsschritts basiert. Diese SLLBM erweitert die LBM in einzigartiger Weise um eine hohe Konvergenzordnung und um die Verwendung großer Zeitschritte. Der Strömungsschritt wurde als eine einzige dünnbesetzte Matrix-Vektor-Multiplikation formuliert. Somit wurden die Probleme 2) und 3) der Eulerschen OLBMs gelöst.

Anhand von Beispielsimulationen wurde demonstriert, wieso dieser Ansatz anderen aktuellen Off-Lattice-Boltzmann-Methoden in Effizienz und Genauigkeit überlegen ist. Im Vergleich zur hier implementierten SEDG-LBM konnte die Laufzeit in der SLLBM pro Iterationsschritt um einen Faktor vier bis fünf verringert werden. Des Weiteren konnten Zeitschritte verwendet werden, die weit jenseits der Stabilitätsgrenze der SEDG-LBM lagen. Gleichzeitig blieben aber andere Vorteile der SEDG-LBM erhalten. So konnte bei einem Konvergenztest die hohe räumliche Konvergenzordnung bestätigt werden, was im Kontext interpolationsbasierter LBM eine Neuheit darstellt.

Zu Demonstrationszwecken wurde außerdem eine erfolgreiche Simulation auf einem irregulären Gitter durchgeführt. Es konnte argumentiert werden, dass die SLLBM auch bei einer Genauigkeit dritter Ordnung auf irregulären Gittern unbeding stabil bleibt. Die SLLBM ist so formuliert, dass sie sich direkt auf unstrukturierte quadrilaterale (2D) und hexagonale Gitter (3D) anwenden lässt. Dabei können auch hierarchische

Gitter mit hängenden Knoten verwendet werden, was die Vernetzung der Gitter vereinfacht. Erste vorläufige Tests zeigen dabei auch positive Ergebnisse in Simulationen wandgebundener Strömungen, die über diese Arbeit hinausgehen, siehe Anhang D.

Beim Vergleich mit einer modernen Finite-Volumen-LBM, dem *Discrete Unified Gas-Kinetic Scheme* (DUGKS), erzeugte die SLLBM wesentlich weniger numerische Dissipation, auch wenn größere Zeitschritte verwendet wurden. Dadurch konnte die Laufzeit gegenüber DUGKS bei gleichwertigen Ergebnissen um den Faktor vier reduziert werden. Bei Verwendung kleiner Zeitschritte und Ansatzfunktionen höherer Ordnung $N > 2$ war die numerische Dissipation sogar geringer als mit der Standard-LBM. Dieses Ergebnis ist überraschend, da die dissipativen Eigenschaften interpolationsbasierter LBMs in der Vergangenheit kritisiert wurden [135]. Die vorteilhaften Ergebnisse der SLLBM unterstützen die SLLBM als vielversprechende Methode für eine weitreichendere Anwendung.

Der nächste Schritt in der Weiterentwicklung der SLLBM ist die Anwendung auf praktische Problemstellungen. In dieser Arbeit wurden dafür alle theoretischen und technischen Voraussetzungen gelegt. Trotz der vielversprechenden Eigenschaften der SLLBM bleibt aber offen, ob sich der Semi-Lagrangesche Strömungsschritt neben dem exakten Strömungsschritt der Standard-LBM etablieren kann. Letzterer bleibt weiterhin ein wichtiger Trumpf der LBM in Bezug auf Effizienz und Genauigkeit. In diesem Zusammenhang ist jedoch besonders interessant, dass sich die Standard-LBM als Spezialfall der SLLBM für $\delta_x = \xi_0 \delta_t$ ergibt. Somit lassen sich Geometrien hybrid vernetzen, sodass das Verfahren nur entlang ausgewählter Koordinatenachsen oder in ausgewählten Bereichen mit der Standard-LBM übereinstimmt. Ein ähnlicher Ansatz zeigte für eine Finite-Differenzen-LBM gute Ergebnisse [59]. Ein wichtiger Faktor ist dabei, dass die meisten praktischen Anwendungen der LBM ohnehin hierarchische Verfeinerungen und somit Interpolationen verwenden.

Eine möglichst effiziente Kombination der SLLBM mit der Standard-LBM müsste dabei die für die Standard-LBM optimierten Datenstrukturen mit den hier verwendeten Strukturen der SLLBM koppeln. Die durch die SLLBM gewonnene Flexibilität könnte in vielen Anwendungen Gitterpunkte und Rechenzeit einsparen, insbesondere in wandgebundenen Strömungen und komplexen Geometrien.

Die Flexibilität der SLLBM bezieht sich auch auf die Wahl der Geschwindigkeitsdiskretisierung, ein Aspekt, der in dieser Arbeit kaum betrachtet wurde. Die hier verwendeten D2Q9- und D3Q19-Sterne sind vor allem motiviert durch die regulären Gitter der Standard-LBM. Da Gitter und Geschwindigkeitsstern in der SLLBM aber entkoppelt sind, motiviert dies die Suche nach besseren Diskretisierungen des Geschwindigkeitsraums. So kann zum Beispiel mit dem Stern der *Crystallographic LBM* eine Entkopplung der dritten Momente vom Impuls erreicht werden [170]. Eine weitere Optimierung der Geschwindigkeitsdiskretisierung für inkompressible OLBMs wurde jedoch in der Literatur bisher kaum betrachtet.

Man kann aber durch die Variation des Geschwindigkeitssterns mehr unabhängige

Freiheitsgrade und daher mehr physikalische Momente abbilden. Dies ist auch ein Grund für die bevorzugte Verwendung von OLBMs zur Simulation kompressibler Strömungen. Dort werden oft alternative Sterne [122] verwendet, um den Isentropenexponenten oder die Prandtl-Zahl unabhängig einstellen zu können, was die Anwendung von OLBMs erfordert [78]. Kompressible OLBMs erzeugen aber oft viel Dissipation und konnten daher in praktischen Anwendungen bisher nicht mit klassischen CFD-Lösern konkurrieren [68]. Es ist anzunehmen, dass die SLLBM auch in kompressiblen Strömungen weniger Dissipation erzeugt als viele andere OLBMs. Zur besseren Darstellung von Verdichtungsstößen wäre in diesem Fall die Verwendung einer Discontinuous-Galerkin-SLLBM [16] denkbar. Zum Beispiel könnten alle Stützstellen für die Interpolation ins Innere der Gitterzellen gesetzt werden. Eine solche Platzierung erlaubt unstetige Lösungen an den Zellgrenzen, wodurch dispersive und dissipative Effekte im Verdichtungsstoß abgeschwächt werden könnten [63].

Weitere Aspekte

Weitere neuartige Aspekte dieser Arbeit sind die Implementierung eines modularen OLBM-Codes und die Erweiterung der LBM um implizite Mehrschrittverfahren.

Der Code NATrium ist zur Publikation eingereicht [127] und soll in diesem Rahmen online verfügbar gemacht werden. In einer Anschlusspromotion sollen weitere Randbedingungen für die SLLBM entwickelt und implementiert werden, um das Verfahren auf praktische Problemstellungen anzuwenden.

Durch die Kombination der LBM mit impliziten Mehrschrittverfahren (Anhang A) konnte die zeitliche Ordnung der Methode erhöht werden, allerdings nur auf Kosten der bedingungslosen Stabilität. Daraus ergab sich in Übereinstimmung mit der zweiten Dahlquist-Barriere [136], dass die zeitliche Diskretisierung der Standard-LBM optimal ist. Aus diesem Grund wurde von einer Weiterverfolgung dieses Ansatzes abgesehen.

Fazit

Insgesamt wurde das Ziel dieser Arbeit erreicht, die beiden Hauptprobleme der Standard-LBM bei der Simulation inkompressibler Wirbelströmungen zu lösen. Mit der hier entwickelten Pseudo-Entropischen Stabilisierung wurde der Kollisionsschritt der LBM so verbessert, dass stabile Simulationen unteraufgelöster turbulenter Strömungen möglich sind. Durch die Entwicklung der Semi-Lagrangeschen LBM ist es gelungen, kritische Schwachstellen von Off-Lattice-Boltzmann-Verfahren zu beheben und die LBM in natürlicher und effizienter Weise auf irreguläre Gitter zu erweitern.

A. Mehrschritt-Lattice-Boltzmann-Methoden

In Abschnitt 2.4.4 wurde mithilfe einer Transformation der Verteilungsfunktionen (Gleichung (2.7)) die implizite Zeitdiskretisierung in eine explizite Form gebracht. Diese Transformation ist ein entscheidender Faktor für die Stabilität und Genauigkeit von Lattice-Boltzmann-Verfahren. In [200] wurde angedeutet, dass sich eine solche explizite Form auch verwenden lässt, wenn der Kollisionsterm mit einem (impliziten) Mehrschritt-Verfahren diskretisiert wird. Dort wurde allerdings nicht gezeigt, dass die Verwendung des Mehrschrittverfahrens die Konvergenzordnung tatsächlich erhöht.

Im Folgenden wird daher die Möglichkeit untersucht, beliebige implizite Mehrschrittverfahren in der LBM zu verwenden. Zunächst wird eine generelle Form der Transformation (2.7) eingeführt. Diese Transformation wird dann auf zwei verschiedene Mehrschrittverfahren angewandt und in Simulationen eines zweidimensionalen Taylor-Green-Wirbels getestet.

Um die Herleitung übersichtlicher zu gestalten, wird die folgende Notation eingeführt

$$f_i^{[\kappa]} := f_{(i)}(t + \kappa\delta_t, \mathbf{x} + \kappa\delta_t\mathbf{e}_{(i)})$$

sodass sich die diskrete BGK-Gleichung mit einem Kollisionsoperator Ω schreiben lässt als

$$\frac{df^{[\kappa]}}{d\kappa} = \Omega(f).$$

Ein beliebiges implizites Mehrschrittverfahren für die diskrete BGK-Gleichung ergibt sich dann als

$$f^{[+1]} - f^{[0]} \approx \sum_{\kappa=-1}^{s-1} a_{-\kappa} \Omega(f^{[-\kappa]}) + \sum_{\kappa=0}^{s-1} b_{-\kappa} f^{[-\kappa]},$$

wobei s die Anzahl der Schritte ist und a_i und b_i die Koeffizienten des Verfahrens bezeichnen. Beliebte implizite Zweischrittverfahren sind das Adams-Moulton-Verfahren

vierter Ordnung (AM4) und das BDF-Verfahren zweiter Ordnung (BDF2). AM4 erreicht die höhere Ordnung durch den Koeffizientensatz $a_1 = 5\delta_t/12$, $a_0 = 2\delta_t/3$, $a_{-1} = -1\delta_t/12$ und $b_0 = b_{-1} = 0$. BDF2 wird wegen seiner guten Stabilitätseigenschaften gerne zur Lösung steifer Differentialgleichungen genutzt: $a_1 = 2\delta_t/3$, $a_0 = a_{-1} = 0$, $b_0 = 1/3$ und $b_{-1} = -1/3$.

Die impliziten Terme werden zu einer transformierten Verteilungsfunktion kombiniert:

$$\bar{f}^{[+1]} := f^{[+1]} - a_1 \Omega(f^{[+1]}) = f^{[0]} + \sum_{\kappa=0}^{s-1} a_{-\kappa} \Omega(f^{[-\kappa]}) + \sum_{\kappa=0}^{s-1} b_{-\kappa} f^{[-\kappa]}.$$

Die Transformation $\bar{f} := f - a_1 \Omega(f)$ wird dann auch auf die rechte Seite angewandt:

$$\bar{f}^{[+1]} = \bar{f}^{[0]} - a_1 \Omega(f^{[0]}) + \sum_{\kappa=0}^{s-1} a_{-\kappa} \Omega(\bar{f}^{[-\kappa]} - a_1 \Omega(f^{[-\kappa]})) \quad (\text{A.1})$$

$$+ \sum_{\kappa=0}^{s-1} b_{-\kappa} (\bar{f}^{[-\kappa]} - a_1 \Omega(f^{[-\kappa]})). \quad (\text{A.2})$$

Setzt man wie in Abschnitt 2.4.4 einen Kollisionsoperator der Form $\Omega(f) = Rf + R^{\text{eq}}(f)$ mit $R^{\text{eq}}(\bar{f}) = R^{\text{eq}}(f)$ voraus, so ergibt sich

$$\Omega(f) = \Omega(\bar{f}) + a_1 R \Omega(f) \Rightarrow \Omega(f) = (I - a_1 R)^{-1} \Omega(\bar{f}).$$

Einsetzen in Gleichung (A.1) liefert einen expliziten Ausdruck

$$\bar{f}^{[+1]} = \bar{f} + a_1 (I - a_1 R)^{-1} \Omega(\bar{f}^{[0]}) + \sum_{\kappa=0}^{s-1} [b_{-\kappa} \bar{f}^{[-\kappa]} + (a_{-\kappa} + a_1 b_{-\kappa}) (I - a_1 R)^{-1} \Omega(\bar{f}^{[-\kappa]})].$$

Für AM4 und BDF2 ergeben sich damit die folgenden Verfahren.

A.1. Adams-Moulton-Verfahren vierter Ordnung

$$\bar{f}^{[+1]} = \bar{f}^{[0]} + \frac{13\delta_t}{12} \left(I - \frac{5\delta_t}{12} R \right)^{-1} \Omega(\bar{f}^{[-1]}) - \frac{\delta_t}{12} \left(I - \frac{5\delta_t}{12} R \right)^{-1} \Omega(\bar{f}^{[-1]}).$$

Für das BGK-Kollisionsmodell vereinfacht sich dies zu

$$\bar{f}^{[+1]} = \bar{f}^{[0]} - \frac{1}{\tau_0} (\bar{f}^{[0]} - f^{\text{eq},[0]}) + \frac{1}{\tau_{-1}} (\bar{f}^{[-1]} - f^{\text{eq},[-1]})$$

mit $\tau_0 = \frac{12}{13} \frac{\lambda}{\delta_t} + \frac{5}{13}$ und $\tau_{-1} = 12 \frac{\lambda}{\delta_t} + 5$.

A.2. BDF-Verfahren zweiter Ordnung

$$\bar{f}^{[+1]} = \frac{4}{3}\bar{f}^{[0]} - \frac{1}{3}\bar{f}^{[-1]} + \frac{8\delta_t}{9} \left(I - \frac{2\delta_t}{3}R \right)^{-1} \Omega(\bar{f}^{[0]}) - \frac{2\delta_t}{9} \left(I - \frac{2\delta_t}{3}R \right)^{-1} \Omega(\bar{f}^{[-1]}).$$

Für das BGK-Kollisionsmodell vereinfacht sich dies zu

$$\bar{f}^{[+1]} = \frac{4}{3}\bar{f}^{[0]} - \frac{1}{3}\bar{f}^{[-1]} - \frac{1}{\tau_0} (\bar{f}^{[0]} - f^{\text{eq},[0]}) + \frac{1}{\tau_{-1}} (\bar{f}^{[-1]} - f^{\text{eq},[-1]})$$

mit $\tau_0 = \frac{9}{8} \frac{\lambda}{\delta_t} + \frac{3}{4}$ und $\tau_{-1} = \frac{9}{2} \frac{\lambda}{\delta_t} + 3$.

A.3. Numerischer Test

Um zu zeigen, dass diese Mehrschritt-LBMs das Potential haben, die Ordnung und Stabilität der LBM zu verbessern, wurden der Taylor-Green-Wirbel aus Abschnitt 2.6.1 mit AM4 und BDF2 gerechnet.

Abbildung A.1 zeigt die Fehler bei der Simulation des zweidimensionalen Taylor-Green-Wirbels. In allen Verfahren konvergierten die Fehler mit der Gitterauflösung bis der Kompressibilitätsfehler dominant wurde. BDF2 hatte hier die gleiche Fehlerordnung wie das BGK-Verfahren, was mit der theoretischen Konvergenzrate übereinstimmt. Allerdings waren die Fehler um eine Größenordnung größer. Bei AM4 zeigte sich eine verbesserte Konvergenzordnung von 3.5. Allerdings waren die Simulationen bei einer Machzahl von 0.05 und einer Auflösung von 16×16 instabil. Die schlechtere Stabilität der AM4-LBM hat sich auch in weitergehenden Untersuchungen bestätigt.

So traten bei einer Simulation der Scherschicht aus Unterabschnitt 2.6.2 mit AM4 schon innerhalb der ersten zehn Zeitschritte Geschwindigkeiten auf, die eine Größenordnung über u_0 lagen. Bei der Simulation des gleichen Problems mit BDF2 war die Simulation zwar stabil, allerdings wurde die kinetische Energie viel zu schnell dissipiert.

A.4. Diskussion

Der numerische Test der Mehrschritt-LBMs zeigt auf der einen Seite, dass sowohl die zeitliche Konvergenzordnung der LBM als auch die Stabilität durch die Verwendung von Mehrschrittverfahren verbessert werden kann. Allerdings wurde auf der anderen Seite mit AM4 die Stabilität und mit BDF2 die Größenordnung der Fehler verschlechtert.

Diese Ergebnisse lassen sich mit der zweiten Dahlquist-Barriere erklären [136]. Diese besagt, dass ein A-stabiles implizites Mehrschritt-Verfahren höchstens quadratische

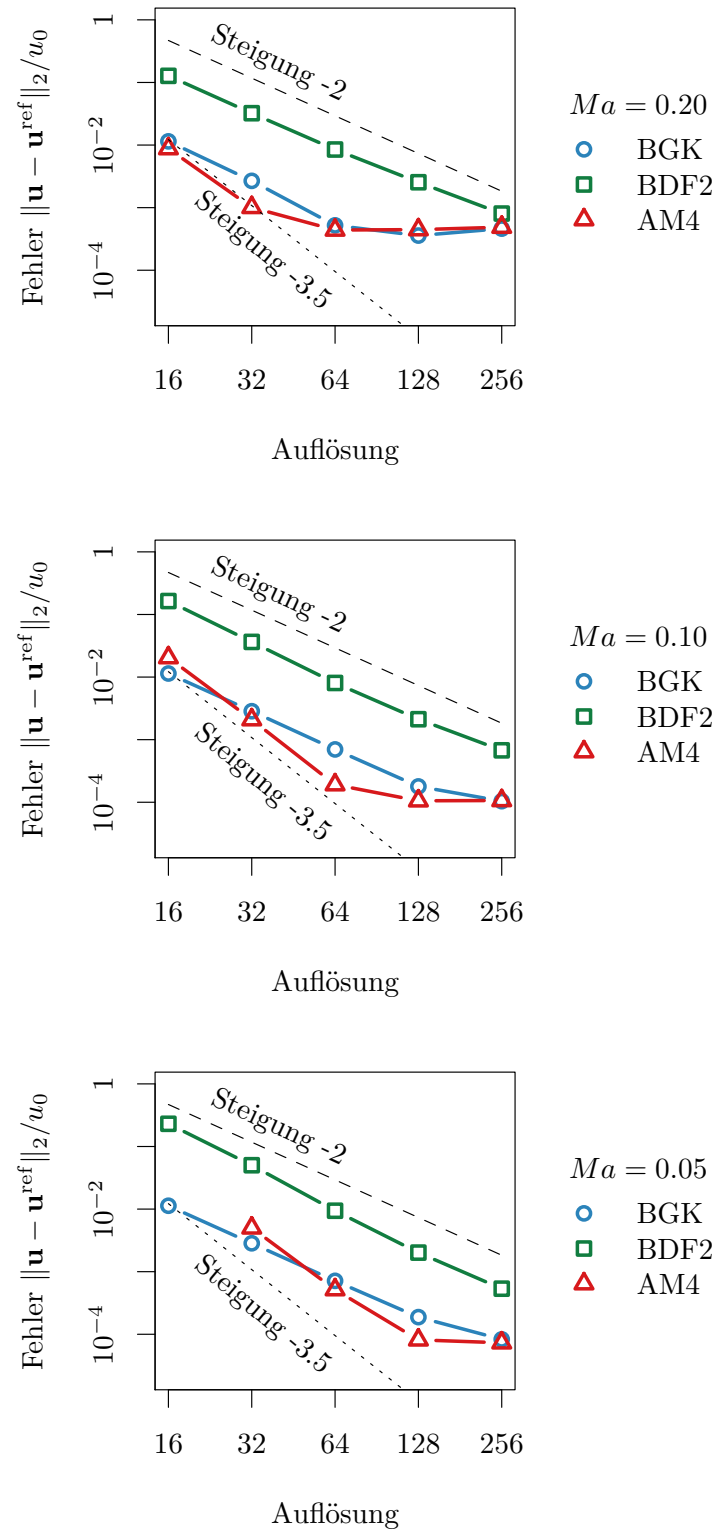


Abbildung A.1: Numerischer Fehler bei der Simulation des zweidimensionalen Taylor-Green-Wirbels mit Mehrschritt-LBM (BDF-Verfahren zweiter Ordnung und Adams-Moulton-Verfahren vierter Ordnung). Die Fehler für das Standard-BGK-Verfahren sind zum Vergleich eingetragen.

Konvergenzordnung haben kann und dass unter diesen Verfahren die Trapezregel die kleinste Fehlerkonstante besitzt. Insbesondere sind die Adams-Moulton-Verfahren höherer Ordnung nicht A-stabil. Die Stabilität solcher Verfahren hängt also kritisch von der Relaxationszeit und der Gitterauflösung ab.

Diese Überlegung zeigt, dass Adams-Moulton-Verfahren höherer Ordnung höchstens in speziellen Situationen bessere Eigenschaften besitzen als die Standard-LBM. Die dort verwendete Trapezregel ist jedoch das optimale A-stabile Verfahren und wird deshalb die Methode der Wahl für die zeitliche Diskretisierung der BGK-Gleichung bleiben. Aus diesen Gründen wurde die Entwicklung der Mehrschritt-LBM in dieser Arbeit nicht weiter verfolgt. Die hier hergeleitete allgemeine Form kann jedoch als Grundlage für weitere Überlegungen dienen.

B. Chapman-Enskog-Analyse der BGK-Gleichung

Die Chapman-Enskog-Analyse verbindet die mesoskopischen Gleichungen der Gaskinetik mit den makroskopischen Erhaltungsgleichungen, siehe dazu auch Abschnitt 2.4.3. Sie wird in der Folge für die diskrete BGK-Gleichung exemplarisch durchgeführt, kann aber in ähnlicher Weise auch direkt auf die Lattice-BGK-Gleichung oder auch auf andere gaskinetische Gleichungen angewandt werden [30, 86].

Dazu wird ein Parameter $\varepsilon > 0$ definiert, der mit der Knudsenzahl identifiziert werden kann. Beim Übergang zu einem Kontinuum, dem Geltungsbereich der hydrodynamischen Gleichungen, werden die Knudsenzahl und damit die Nichtgleichgewichtsanteile der Geschwindigkeitsverteilungsfunktion f klein. Da gerade die viskosen Spannungen sich aber in den Nichtgleichgewichtsanteilen manifestieren, müssen diese in der Analyse besonders berücksichtigt werden. Dazu werden die Verteilungsfunktionen in eine asymptotische Reihe entwickelt

$$f = f^{(0)} + \varepsilon f^{(1)} + \varepsilon^2 f^{(2)} + \dots$$

Die Anteile $f^{(k)}$ modellieren dabei verschiedene Ordnungen der Abweichung vom Gleichgewicht.

Um die unterschiedlichen Anteile der Gleichgewichtsverteilung miteinander zu koppeln, werden die Ableitungsoperatoren umdefiniert zu

$$\frac{\partial}{\partial x_i} = \varepsilon \partial_{x1i} \quad \text{und} \quad \frac{\partial}{\partial t} = \varepsilon \partial_{t1} + \varepsilon^2 \partial_{t2}.$$

Durch die Aufspaltung der zeitlichen Ableitung in eine schnelle und eine langsamere Zeitskala können die Effekte der Teilchenkollisionen auf Diffusionsprozesse analysiert werden.

Die Chapman-Enskog-Analyse der diskreten BGK-Gleichung erfordert zwei Vorüberlegungen zur Diskretisierung des Geschwindigkeitsraums, die zum Beispiel in [86] bewiesen werden.

Lemma B.1 Die Gewichte ω_i und diskreten Geschwindigkeiten \mathbf{e}_i erfüllen die folgenden Gitterinvarianten.

$$\sum_i \omega_i = 1, \quad (\text{B.1})$$

$$\sum_i \omega_{(i)} e_{(i)j} = 0, \quad (\text{B.2})$$

$$\sum_i \omega_{(i)} e_{(i)j} e_{(i)k} = c_s^2 \delta_{jk}, \quad (\text{B.3})$$

$$\sum_i \omega_{(i)} e_{(i)j} e_{(i)k} e_{(i)l} = 0, \quad (\text{B.4})$$

$$\sum_i \omega_{(i)} e_{(i)j} e_{(i)k} e_{(i)l} e_{(i)r} = c_s^4 (\delta_{jk} \delta_{lr} + \delta_{jl} \delta_{kr} + \delta_{jr} \delta_{kl}). \quad (\text{B.5})$$

□

Mit diesen Gitterinvarianten lassen sich die Momente der quadratischen Gleichgewichtsverteilung bestimmen.

Lemma B.2 Die Momente der diskreten Gleichgewichtsverteilung

$$f_i^{\text{eq}}(x, t) = w_{(i)} \rho \left(1 + \frac{u_{(i)j} e_{(i)j}}{c_s^2} - \frac{u_{(i)j} u_{(i)j}}{2c_s^2} + \frac{u_{(i)j} e_{(i)j} u_{(i)k} e_{(i)k}}{2c_s^4} \right)$$

ergeben sich wie folgt:

$$\sum_i f_i^{\text{eq}} = \rho, \quad (\text{B.6})$$

$$\sum_i f_{(i)}^{\text{eq}} e_{(i)j} = \rho u_j, \quad (\text{B.7})$$

$$\Pi_{jk}^{(0)} := \sum_i f_{(i)}^{\text{eq}} e_{(i)j} e_{(i)k} = c_s^2 \rho \delta_{jk} + \rho u_j u_k, \quad (\text{B.8})$$

$$Q_{jkl}^{(0)} := \sum_i f_{(i)}^{\text{eq}} e_{(i)j} e_{(i)k} e_{(i)l} = c_s^2 \rho (\delta_{jk} u_l + \delta_{jl} u_k + \delta_{kl} u_j). \quad (\text{B.9})$$

□

Die folgende Analyse teilt sich in drei Schritte. Zuerst wird die diskrete BGK-Gleichung in die verschiedenen Skalen aufgespalten. Dann werden auf jeder Skala die Momentengleichungen aufgestellt und analysiert. Schließlich werden die Gleichungen wieder zusammengesetzt und mit den Navier-Stokes-Gleichungen verglichen.

B.1. Mehrskalen-Entwicklung

In die diskrete BGK-Gleichung

$$\frac{\partial f_i}{\partial t} + e_{(i)j} \frac{\partial f_{(i)}}{\partial x_j} = -\frac{1}{\lambda} (f_i(x, t) - f_i^{\text{eq}}(x, t))$$

werden zunächst die asymptotische Entwicklung der Verteilungsfunktionen und die undefinierten Ableitungsoperatoren eingesetzt. Dies ergibt eine lange Gleichung, die in verschiedene Ordnungen von ε aufgespalten wird.

$$\text{(Für } \varepsilon^0 \text{ :)} \quad 0 = -\frac{1}{\lambda} (f_i^{(0)} - f_i^{\text{eq}}), \quad (\text{B.10})$$

$$\text{(für } \varepsilon^1 \text{ :)} \quad \partial_{t1} f_i^{(0)} + e_{(i)j} \partial_{x1j} f_{(i)}^{(0)} = -\frac{1}{\lambda} f_i^{(1)}, \quad (\text{B.11})$$

$$\text{(für } \varepsilon^2 \text{ :)} \quad \partial_{t2} f_i^{(0)} + \partial_{t1} f_i^{(1)} + e_{(i)j} \partial_{x1j} f_{(i)}^{(1)} = -\frac{1}{\lambda} f_i^{(2)}. \quad (\text{B.12})$$

B.2. Analyse der Momentengleichungen

B.2.1. Analyse der Gleichung nullter Ordnung

Aus der Gleichung nullter Ordnung (B.10) ergibt sich sofort $f_i^{(0)} = f_i^{\text{eq}}$. Durch Aufsummieren und Einsetzen der Gleichung (B.6) sieht man, dass nur die Gleichgewichtsanteile einen Beitrag zur Dichte liefern:

$$\sum_i f_i^{(0)} = \rho,$$

$$\sum_i f_i^{(1)} = \sum_i f_i^{(2)} = 0.$$

Die gleiche Aussage erhält man mit (B.7) für den Impuls durch Multiplikation mit e_{ij} und Aufsummieren:

$$f_i^{(0)} e_{ij} = \rho u_j,$$

$$f_i^{(1)} e_{ij} = f_i^{(2)} e_{ij} = 0.$$

B.2.2. Analyse der Gleichung erster Ordnung

Aufsummieren von (B.11) ergibt die Kontinuitätsgleichung auf der langsamen Zeitskala

$$\boxed{\partial_{t1} \rho + \partial_{x1i} (\rho u_i) = 0.} \quad (\text{B.13})$$

Für den Impuls erhält man

$$\partial_{t1}(\rho u_i) + \partial_{x1j} \Pi_{ij}^{(0)} = 0.$$

Einsetzen des Impulsflusstensors (B.8) liefert die Euler-Gleichung für reibungsfreie Strömungen

$$\boxed{\partial_{t1}(\rho u_i) + \partial_{x1j}(\rho u_i u_j) = -\partial_{x1i}(c_s^2 \rho)}, \quad (\text{B.14})$$

wobei sich der Druck durch die ideale Gasgleichung als $p = c_s^2 \rho + \text{const}$ ergibt.

Als Letztes kann aus der Gleichung erster Ordnung (durch Multiplikation mit $e_{(i)j} e_{(i)k}$ und Aufsummieren) der Nichtgleichgewichtsanteil des Impulsflusstensors hergeleitet werden

$$\begin{aligned} \Pi_{jk}^{(1)} &:= \sum_i f_{(i)}^{(1)} e_{(i)j} e_{(i)k} = -\lambda \left(\partial_{t1} \Pi_{jk}^{(0)} + \partial_{x1l} Q_{jkl}^{(0)} \right) \\ &= -\lambda \left(\partial_{t1} (c_s^2 \rho \delta_{jk} + \rho u_j u_k) + \partial_{x1l} (c_s^2 \rho (\delta_{jk} u_l + \delta_{jl} u_k + \delta_{kl} u_j)) \right), \end{aligned}$$

mit Gleichungen (B.8) und (B.9).

Durch mehrfaches Anwenden der Produktregel und Einsetzen der Kontinuitätsgleichung (B.13) wird dies umgeformt zu

$$\Pi_{ij}^{(1)} = -\lambda c_s^2 \rho (\partial_{x1j} u_i + \partial_{x1i} u_j) + \lambda \partial_{x1k} (\rho u_i u_j u_k). \quad (\text{B.15})$$

B.2.3. Analyse der Gleichung zweiter Ordnung

Aufsummieren der Gleichung zweiter Ordnung (B.12) liefert

$$\boxed{\partial_{t2} \rho = 0}. \quad (\text{B.16})$$

Für den Impuls erhält man

$$\partial_{t2}(\rho u_i) + \partial_{x1j} \Pi_{ij}^{(1)} = 0.$$

Einsetzen des oben berechneten Tensors $\Pi_{ij}^{(1)}$ aus Gleichung (B.15) liefert

$$\boxed{\partial_{t2}(\rho u_i) - \lambda c_s^2 \partial_{x1j} [\rho (\partial_{x1j} u_i + \partial_{x1i} u_j)] + \lambda \partial_{x1jk}^2 (\rho u_i u_j u_k) = 0}. \quad (\text{B.17})$$

B.3. Zusammensetzen der Skalen

Die umrahmten Gleichungen beschreiben jeweils die Dichte- und Impulserhaltung auf der schnellen und langsamen Zeitskala. Diese Gleichungen werden jetzt mit ε

beziehungsweise ε^2 multipliziert und wieder zusammengesetzt. Dadurch gewinnt man die ursprünglichen Ableitungsoperatoren zurück.

Für die Dichte erhält man exakt die kompressible Kontinuitätsgleichung (2.1)

$$\frac{\partial \rho}{\partial t} + \frac{\partial \rho u_i}{\partial x_i} = 0.$$

Für den Impuls erhält man die Gleichung

$$\boxed{\frac{\partial \rho u_i}{\partial t} + \frac{\partial \rho u_i u_j}{\partial x_j} = -\frac{\partial c_s^2 \rho}{\partial x_i} + \lambda c_s^2 \frac{\partial}{\partial x_j} \left[\rho \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \right] - \lambda \frac{\partial^2 \rho u_i u_j u_k}{\partial x_j \partial x_k}}.$$

Bis auf den letzten Term entspricht diese der kompressiblen Impulsgleichung (2.2) mit Stokesschem Spannungstensor, wobei der Druck durch $p = c_s^2 \rho + \text{const}$ und die Viskosität durch $\nu = \lambda c_s^2$ definiert sind.

Der Fehlerterm

$$E_{\rho u_i} := -\lambda \frac{\partial^2 \rho u_i u_j u_k}{\partial x_j \partial x_k}. \quad (\text{B.18})$$

ist ein Artefakt der Diskretisierung des Geschwindigkeitsraums. Da die Dichtefluktuationen mit $O(Ma^2)$ und die Geschwindigkeiten mit $O(Ma)$ skalieren, ist dieser Term von der Ordnung $O(Ma^5)$, während der viskose Term von der Ordnung $O(Ma^3)$ ist. Die Abweichung vom Spannungstensor der kompressiblen Navier-Stokes-Gleichungen ist somit von der Ordnung $O(Ma^2)$.

Im Bereich kleiner Machzahlen ist dieser Fehler vernachlässigbar, weshalb die Standard-LBM für schwach kompressible Strömungen physikalische Ergebnisse liefert. Durch die Verwendung größerer Geschwindigkeitsdiskretisierungen kann der Fehler reduziert werden, um die LBM auch für Strömungen mit größeren Machzahlen zu verwenden. Im inkompressiblen Grenzbereich erzeugt auch die Kopplung von Druck und Dichte einen Fehler zweiter Ordnung, da der Druck sich in der LBM in Dichtefluktuationen der Ordnung $O(Ma^2)$ ausdrückt.

Obwohl die Chapman-Enskog-Analyse auf mathematisch relativ losen Annahmen beruht, ist sie in der LBM die Standardmethode zur Herleitung makroskopischer Erhaltungsgleichungen geworden. Sie bildet somit unter anderem das wichtigste Werkzeug bei der Entwicklung von Kollisionsmodellen.

C. Momentenmatrizen und Stabilisierungsoperatoren

C.1. Matrizen für den D2Q9-Stern

Die Momente für den D2Q9-Stern von Lallemand und Luo [135] sind in Tabelle C.1 definiert.

Die Matrix A aus Gleichung (3.7) ergibt sich daraus als

$$\begin{pmatrix} \hat{\varepsilon}^{\text{pc}} \\ q_x^{\text{pc}} \\ q_y^{\text{pc}} \end{pmatrix} = m_Q = A \cdot m_P = \begin{pmatrix} -1 & 0 & 0 & -1 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} \rho \\ j_x \\ j_y \\ e^{\text{pc}} \\ p_{(x)(x)}^{\text{pc}} \\ p_{xy}^{\text{pc}} \end{pmatrix}.$$

Ausgeschrieben sind die stabilisierten Momente also wie folgt gekoppelt:

$$\begin{aligned} \hat{\varepsilon}^{\text{pc}} &= -\rho - e^{\text{pc}}, \\ q_x^{\text{pc}} &= -j_x, \\ q_y^{\text{pc}} &= -j_y. \end{aligned}$$

Tabelle C.1: Momentenbasis von Lallemand und Luo [135].

f_i	f_0	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8
\mathbf{e}_i/ξ_0	0	1	0	-1	0	1	-1	-1	1
	0	0	1	0	-1	1	1	-1	-1
w_i	$\frac{4}{9}$	$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{36}$	$\frac{1}{36}$	$\frac{1}{36}$	$\frac{1}{36}$
ρ	1	1	1	1	1	1	1	1	1
e	-4	-1	-1	-1	-1	2	2	2	2
$\hat{\varepsilon}$	4	-2	-2	-2	-2	1	1	1	1
j_x	0	1	0	-1	0	1	-1	-1	1
q_x	0	-2	0	2	0	1	-1	-1	1
j_y	0	0	1	0	-1	1	1	-1	-1
q_y	0	0	-2	0	2	1	1	-1	-1
$p_{(x)(x)}$	0	1	-1	1	-1	0	0	0	0
p_{xy}	0	0	0	0	0	1	-1	1	-1

Die Matrix B aus Gleichung (3.8) ergibt sich als

$$B = \begin{pmatrix} \frac{8}{9} & \frac{2}{9} & \frac{2}{9} & \frac{2}{9} & \frac{2}{9} & -\frac{4}{9} & -\frac{4}{9} & -\frac{4}{9} & -\frac{4}{9} \\ \frac{1}{18} & \frac{13}{18} & -\frac{1}{9} & \frac{1}{18} & -\frac{1}{9} & \frac{5}{9} & -\frac{1}{9} & -\frac{1}{9} & \frac{5}{9} \\ \frac{1}{18} & -\frac{1}{9} & \frac{13}{18} & -\frac{1}{9} & \frac{1}{18} & \frac{5}{9} & \frac{5}{9} & -\frac{1}{9} & -\frac{1}{9} \\ \frac{1}{18} & \frac{1}{18} & -\frac{1}{9} & \frac{13}{18} & -\frac{1}{9} & -\frac{1}{9} & \frac{5}{9} & \frac{5}{9} & -\frac{1}{9} \\ \frac{1}{18} & -\frac{1}{9} & \frac{1}{18} & -\frac{1}{9} & \frac{13}{18} & -\frac{1}{9} & -\frac{1}{9} & \frac{5}{9} & \frac{5}{9} \\ -\frac{1}{36} & \frac{5}{36} & \frac{5}{36} & -\frac{1}{36} & -\frac{1}{36} & \frac{5}{9} & -\frac{1}{9} & \frac{2}{9} & -\frac{1}{9} \\ -\frac{1}{36} & -\frac{1}{36} & \frac{5}{36} & \frac{5}{36} & -\frac{1}{36} & -\frac{1}{9} & \frac{5}{9} & -\frac{1}{9} & \frac{2}{9} \\ -\frac{1}{36} & -\frac{1}{36} & -\frac{1}{36} & \frac{5}{36} & \frac{5}{36} & \frac{2}{9} & -\frac{1}{9} & \frac{5}{9} & -\frac{1}{9} \\ -\frac{1}{36} & \frac{5}{36} & -\frac{1}{36} & -\frac{1}{36} & \frac{5}{36} & -\frac{1}{9} & \frac{2}{9} & -\frac{1}{9} & \frac{5}{9} \end{pmatrix}. \quad (\text{C.1})$$

Die Basiswechselmatrix aus der Momentenbasis von Lallemand und Luo [135] in die Basis von Dellar [42] findet sich in Tabelle C.2. Diese zeigt, dass eine Optimierung von $\hat{\varepsilon}$, q_x und q_y keinen Einfluss auf die hydrodynamischen Momente der Basis von

Tabelle C.2: Basiswechselmatrix aus der Momentenbasis [135] in die Momentenbasis [42]. Leere Tabelleneinträge entsprechen Nullen in der Matrix.

	ρ	\dot{j}_x	\dot{j}_y	e	$p_{(x)(x)}$	p_{xy}	$\hat{\varepsilon}$	q_x	q_y
ρ	1.00								
\dot{j}_x		1.00							
\dot{j}_y			1.00						
$\Pi_{(x)(x)}$	1.50			0.75	2.25				
Π_{xy}						9.00			
$\Pi_{(y)(y)}$	1.50			0.75	-2.25				
\hat{N}	1.00			1.00			1.00		
J_x		2.00						2.00	
J_y			2.00						2.00

Dellar hat.

C.2. Matrizen für den D3Q19-Stern

Die Momente für den D3Q19-Stern von D’Humières et al. [47] sind in Tabelle C.3 definiert.

Die Matrix A aus Gleichung (3.7) ergibt sich daraus als

$$\begin{pmatrix} \hat{\varepsilon}^{\text{pc}} \\ q_x^{\text{pc}} \\ q_y^{\text{pc}} \\ q_z^{\text{pc}} \\ 3\pi_{(x)(x)}^{\text{pc}} \\ \pi_{(w)(w)}^{\text{pc}} \\ m_x^{\text{pc}} \\ m_y^{\text{pc}} \\ m_z^{\text{pc}} \end{pmatrix} = m_Q = A \cdot m_P = \begin{pmatrix} -\frac{7}{38} & -\frac{11}{38} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{2}{3} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -\frac{2}{3} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -\frac{2}{3} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -\frac{1}{2} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -\frac{1}{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} \rho \\ e^{\text{pc}} \\ \dot{j}_x \\ \dot{j}_y \\ \dot{j}_z \\ 3p_{(x)(x)}^{\text{pc}} \\ p_{(w)(w)}^{\text{pc}} \\ p_{xy}^{\text{pc}} \\ p_{yz}^{\text{pc}} \\ p_{xz}^{\text{pc}} \end{pmatrix}.$$

Tabelle C.3: Momentenbasis von D'Humières et al. [47].

f_i	f_0	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}	f_{13}	f_{14}	f_{15}	f_{16}	f_{17}	f_{18}
e_i/ξ_0	0	1	0	-1	0	0	0	1	-1	-1	1	1	1	-1	-1	0	0	0	0
	0	0	0	0	0	-1	1	0	0	0	0	-1	1	1	-1	-1	1	1	-1
	0	0	1	0	-1	0	0	1	1	-1	-1	0	0	0	0	1	1	-1	-1
w_i	$\frac{1}{3}$	$\frac{1}{18}$	$\frac{1}{18}$	$\frac{1}{18}$	$\frac{1}{18}$	$\frac{1}{18}$	$\frac{1}{18}$	$\frac{1}{36}$	$\frac{1}{36}$	$\frac{1}{36}$	$\frac{1}{36}$	$\frac{1}{36}$	$\frac{1}{36}$	$\frac{1}{36}$	$\frac{1}{36}$	$\frac{1}{36}$	$\frac{1}{36}$	$\frac{1}{36}$	$\frac{1}{36}$
ρ	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
e	-30	-11	-11	-11	-11	-11	-11	8	8	8	8	8	8	8	8	8	8	8	8
$\hat{\varepsilon}$	12	-4	-4	-4	-4	-4	-4	1	1	1	1	1	1	1	1	1	1	1	1
j_x	0	1	0	-1	0	0	0	1	-1	-1	1	1	1	-1	-1	0	0	0	0
q_x	0	-4	0	4	0	0	0	1	-1	-1	1	1	1	-1	-1	0	0	0	0
j_y	0	0	0	0	0	-1	1	0	0	0	0	-1	1	1	-1	-1	1	1	-1
q_y	0	0	0	0	0	4	-4	0	0	0	0	-1	1	1	-1	-1	1	1	-1
j_z	0	0	1	0	-1	0	0	1	1	-1	-1	0	0	0	0	1	1	-1	-1
q_z	0	0	-4	0	4	0	0	1	1	-1	-1	0	0	0	0	1	1	-1	-1
$3p_{(x)(x)}$	0	2	-1	2	-1	-1	-1	1	1	1	1	1	1	1	1	-2	-2	-2	-2
$3\pi_{(x)(x)}$	0	-4	2	-4	2	2	2	1	1	1	1	1	1	1	1	-2	-2	-2	-2
$p_{(w)(w)}$	0	0	-1	0	-1	1	1	-1	-1	-1	-1	1	1	1	1	0	0	0	0
$\pi_{(w)(w)}$	0	0	2	0	2	-2	-2	-1	-1	-1	-1	1	1	1	1	0	0	0	0
p_{xy}	0	0	0	0	0	0	0	0	0	0	0	-1	1	-1	1	0	0	0	0
p_{yz}	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-1	1	-1	1
p_{xz}	0	0	0	0	0	0	0	1	-1	1	-1	0	0	0	0	0	0	0	0
m_x	0	0	0	0	0	0	0	-1	1	1	-1	1	1	-1	-1	0	0	0	0
m_y	0	0	0	0	0	0	0	0	0	0	0	1	-1	-1	1	-1	1	1	-1
m_z	0	0	0	0	0	0	0	1	1	-1	-1	0	0	0	0	-1	-1	1	1

Ausgeschrieben sind die stabilisierten Momente also wie folgt gekoppelt:

$$\begin{aligned}\hat{\varepsilon} &= -\frac{7}{38}\rho - \frac{11}{38}e^{\text{pc}}, \\ q_{x,y,z}^{\text{pc}} &= -\frac{2}{3}j_{x,y,z}, \\ 3\pi_{(x)(x)}^{\text{pc}} &= -\frac{1}{2}(3p_{(x)(x)}^{\text{pc}}), \\ \pi_{(w)(w)}^{\text{pc}} &= -\frac{1}{2}p_{(w)(w)}^{\text{pc}}, \\ m_{x,y,z}^{\text{pc}} &= 0.\end{aligned}$$

D. Randbedingungen für die Semi-Lagrangesche Methode

In dieser Arbeit wurden einige periodische Testfälle mit der SLLBM gerechnet, um ein sauberes Fundament für die Methode zu schaffen. Erste Tests, die über diese Arbeit hinausgehen, haben jedoch gezeigt, dass Randbedingungen definiert werden können, die die Vorteile der SLLBM nicht beeinträchtigen. In diesem Anhang werden diese Ansätze kurz beschrieben.

Im Semi-Lagrangeschen Strömungsschritt werden die Verteilungsfunktionen $f_i(\mathbf{x}_{cj})$ an den Gitterpunkten \mathbf{x}_{cj} der Zelle D_c durch Interpolationen konstruiert. Dazu wird die interpolierte Verteilungsfunktion $f_i(t - \delta_t)$ an den Ausgangspunkten $\mathbf{x}_{cj} - \delta_t \mathbf{e}_i$ ausgewertet, siehe Gleichung (5.3). Diese Interpolationen sind nicht definiert, wenn der Ausgangspunkt außerhalb des Strömungsgebiets liegt. Hier lassen sich die folgenden Fälle unterscheiden.

D.1. Konstruktion der Verteilungsfunktionen an Randpunkten

Im ersten, einfacheren Fall ist \mathbf{x}_{cj} ein Randpunkt des Gebiets, womit direkt Randbedingungen der Standard-LBM verwendet werden können. Dort werden die bekannten (ausströmenden) Geschwindigkeitsverteilungen und die am Rand definierten Drücke/Geschwindigkeiten kombiniert, um $\rho(t, \mathbf{x}_{cj})$, $\mathbf{u}(t, \mathbf{x}_{cj})$ und den Spannungstensor $\mathbf{\Pi}^{(1)}(t, \mathbf{x}_{cj})$ zu berechnen [140, 201, 244]. Diese Größen können dann verwendet werden, um die lokalen Verteilungsfunktionen zu berechnen und das Problem damit zu schließen, zum Beispiel durch [140]

$$f_i := f_i^{\text{eq}}(\rho, \mathbf{u}) + \frac{\omega^{(i)}}{2c_s^4} (e_{(i)j}e_{(i)l} - c_s^2\delta_{jl}) \Pi_{jl}^{(1)}. \quad (\text{D.1})$$

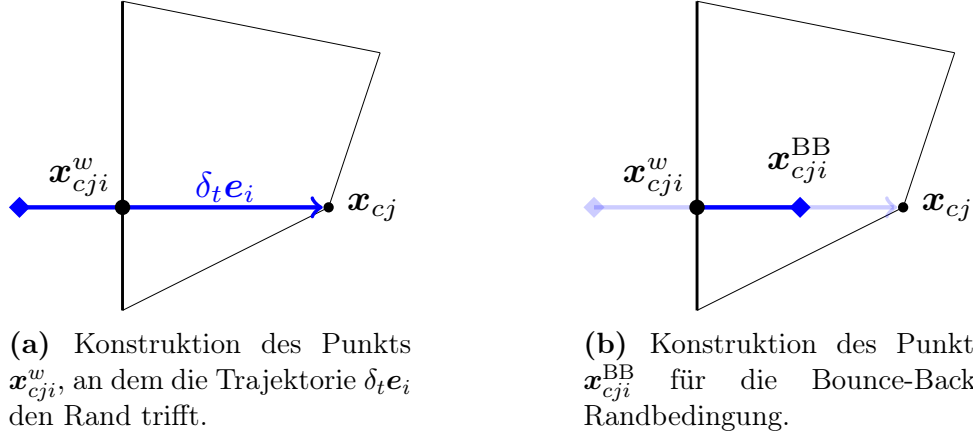


Abbildung D.1: Konstruktion der Punkte \mathbf{x}_{cji}^w und \mathbf{x}_{cji}^{BB} in den Randbedingungen der SLLBM ausgehend von der Stützstelle \mathbf{x}_{cj} .

D.2. Konstruktion der Verteilungsfunktionen an inneren Punkten

Im zweiten Fall ist \mathbf{x}_{cj} kein Randpunkt des Gebiets. Diese Situation tritt nur bei großen CFL-Zahlen auf. Hier kann zuerst der Randpunkt \mathbf{x}_{cji}^w konstruiert werden, an dem die Trajektorie der Verteilungsfunktion den Rand trifft, siehe Abbildung D.1a. Der Zeitpunkt θ_{cji}^w des Auftreffens auf dem Rand lässt sich bestimmen als

$$\theta_{cji}^w := t - \frac{|\mathbf{x}_{cj} - \mathbf{x}_{cji}^w|}{|\mathbf{e}_i|} \in [t - \delta_t, t].$$

Wegen der Linearität des Strömungsschritts (siehe Gleichung (5.1)) erhält man

$$f_i(t, \mathbf{x}_{cj}) = f_i(\theta_{cji}^w, \mathbf{x}_{cji}^w).$$

Wie in den Randbedingungen der Standard-LBM (siehe z.B. [140]) können aus den ausströmenden Verteilungsfunktionen und den vorgeschriebenen Randbedingungen die folgenden Größen berechnet werden:

$$\rho(t, \mathbf{x}_{cji}^w), \mathbf{u}(t, \mathbf{x}_{cji}^w), \mathbf{\Pi}^{(1)}(t, \mathbf{x}_{cji}^w).$$

Aus dem vorigen Zeitschritt sind alle Verteilungsfunktionen und damit die Größen

$$\rho(t - \delta_t, \mathbf{x}_{cji}^w), \mathbf{u}(t - \delta_t, \mathbf{x}_{cji}^w), \mathbf{\Pi}^{(1)}(t - \delta_t, \mathbf{x}_{cji}^w).$$

bekannt. Durch eine lineare Interpolation in der Zeit erhält man zum Zeitpunkt θ_{cji}^w :

$$\Gamma(\theta_{cji}^w, \mathbf{x}_{cji}^w) := \frac{t - \theta_{cji}^w}{\delta_t} \cdot \Gamma(t - \delta_t, \mathbf{x}_{cji}^w) + \frac{\theta_{cji}^w - (t - \delta_t)}{\delta_t} \cdot \Gamma(t, \mathbf{x}_{cji}^w),$$

wobei Γ als Platzhalter für ρ , \mathbf{u} und $\mathbf{\Pi}^{(1)}$ steht. Diese interpolierten Werte können dann in Gleichung (D.1) eingesetzt werden, um die unbekannt einströmenden Verteilungsfunktionen zu berechnen.

D.3. Bounce-Back-Randbedingung

Die *Bounce-Back*-Randbedingung der Nichtgleichgewichtsverteilung aus Abschnitt 4.2.3 kann konsistent mit dem Semi-Lagrangeschen Strömungsschritt formuliert werden. Dazu sei l der Index mit $\mathbf{e}_i = -\mathbf{e}_l$.

Damit können die einfließenden Verteilungsfunktionen f_i wie folgt gesetzt werden:

$$f^{(i)}(\theta_{(cji)}^w, \mathbf{x}_{(cji)}^w) = f_l(\theta_{(cji)}^w, \mathbf{x}_{(cji)}^w) + 2\omega^{(i)}\rho_0(\mathbf{e}_{(i)} \cdot \mathbf{u}_w)/c_s^2, \quad (\text{D.2})$$

wobei \mathbf{u}_w die vorgeschriebene Wandgeschwindigkeit bezeichnet. Dies entspricht Gleichung (4.4). Die Verteilungsfunktionen f_l erhält man, indem man der Trajektorie \mathbf{e}_l weiter zum Ausgangspunkt $\mathbf{x}_{cji}^{\text{BB}}$ folgt, siehe Abbildung D.1b. Damit ergibt sich

$$\begin{aligned} \mathbf{x}_{cji}^{\text{BB}} &= \mathbf{x}_{cji}^w - (\theta_{cji}^w - (t - \delta_t)) \mathbf{e}_l, \\ f_l(\theta_{(cji)}^w, \mathbf{x}_{(cji)}^w) &= f_l(t - \delta_t, \mathbf{x}_{cji}^{\text{BB}}). \end{aligned}$$

Da die Verteilungen zum Zeitpunkt $t - \delta_t$ bekannt sind, kann $f_l(t - \delta_t, \mathbf{x}_{cji}^{\text{BB}})$ durch Interpolation rekonstruiert werden, siehe Gleichung (5.3). Dieser Ausdruck wird in Gleichung (D.2) eingesetzt, um die unbekannt Verteilungsfunktionen zu erhalten.

Somit sind die grundsätzlichen Konzepte entwickelt, die notwendig sind, um die SLLBM für wandgebundene Strömungen zu verwenden. Die hier definierten Randbedingungen sind teilweise in NATrium implementiert und zeigen in ersten Tests vielversprechende Ergebnisse. Die detaillierte Diskussion dieser Randbedingungen geht jedoch über diese Arbeit hinaus.

Literaturverzeichnis

- [1] Exa PowerFLOW. <http://exa.com>, 2017.
- [2] Palabos - CFD complex physics. <http://www.palabos.org>, 2017.
- [3] AIDUN, C. K., AND CLAUSEN, J. R. Lattice-boltzmann method for complex flows. *Annual Review of Fluid Mechanics* 42, 1 (2010), 439–472.
- [4] AL-ZOUBI, A., AND BRENNER, G. Simulating fluid flow over sinusoidal surfaces using the lattice boltzmann method. *Computers & Mathematics with Applications* 55, 7 (2008), 1365–1376.
- [5] ALEXANDER, R. Diagonally implicit runge–kutta methods for stiff O.D.E.’s. *SIAM Journal on Numerical Analysis* 14, 6 (1977), 1006–1021.
- [6] ANDERSON, J. D. *Computational fluid dynamics: The basics with applications*, 8. [impr.] ed. McGraw-Hill series in aeronautical and aerospace engineering. McGraw-Hill, New York, 2001.
- [7] BANGERTH, W., BURSTEDDE, C., HEISTER, T., AND KRONBICHLER, M. Algorithms and data structures for massively parallel generic adaptive finite element codes. *ACM Transactions on Mathematical Software* 38, 2 (2011), 1–28.
- [8] BANGERTH, W., DAVYDOV, D., HEISTER, T., HELTAI, L., KANSCHAT, G., KRONBICHLER, M., MAIER, M., TURCK SIN, B., AND WELLS, D. The deal.ii library, version 8.4. *Journal of Numerical Mathematics* 24, 3 (2016).
- [9] BANGERTH, W., HARTMANN, R., AND KANSCHAT, G. deal.ii—a general-purpose object-oriented finite element library. *ACM Transactions on Mathematical Software* 33, 4 (2007), 24–es.
- [10] BARDOW, A., KARLIN, I. V., AND GUSEV, A. A. General characteristic-based algorithm for off-lattice boltzmann simulations. *Europhysics Letters (EPL)* 75, 3 (2006), 434–440.

- [11] BASHA, M., AND NOR AZWADI, C. S. Regularized lattice boltzmann simulation of laminar mixed convection in the entrance region of 2-d channels. *Numerical Heat Transfer, Part A: Applications* 63, 11 (2013), 867–878.
- [12] BHATNAGAR, P. L., GROSS, E. P., AND KROOK, M. A model for collision processes in gases. i. small amplitude processes in charged and neutral one-component systems. *Physical Review* 94, 3 (1954), 511–525.
- [13] BINNEY, J., AND TREMAINE, S. *Galactic Dynamics: (Second Edition)*. Princeton University Press, Princeton, 2011.
- [14] BO, Y., WANG, P., GUO, Z., AND WANG, L.-P. Dugks simulations of three-dimensional taylor–green vortex flow and turbulent channel flow. *Computers & Fluids* (2017).
- [15] BOGHOSIAN, B. M., YEPEZ, J., COVENEY, P. V., AND WAGER, A. Entropic lattice boltzmann methods. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 457, 2007 (2001), 717–766.
- [16] BOKANOWSKI, O., AND SIMARMATA, G. Semi-lagrangian discontinuous galerkin schemes for some first- and second-order partial differential equations. *ESAIM: Mathematical Modelling and Numerical Analysis* 50, 6 (2016), 1699–1730.
- [17] BORN, M., AND GREEN, H. S. A general kinetic theory of liquids. i. the molecular distribution functions. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 188, 1012 (1946), 10–18.
- [18] BÖSCH, F., CHIKATAMARLA, S. S., AND KARLIN, I. Entropic multi-relaxation models for simulation of fluid turbulence. *ESAIM: Proceedings and Surveys* 52 (2015), 1–24.
- [19] BOUZIDI, M., D’HUMIÈRES, D., LALLEMAND, P., AND LUO, L.-S. Lattice boltzmann equation on a two-dimensional rectangular grid. *Journal of Computational Physics* 172, 2 (2001), 704–717.
- [20] BRACHET, M. E. Direct simulation of three-dimensional turbulence in the taylor–green vortex. *Fluid Dynamics Research* 8, 1-4 (1991), 1–8.
- [21] BRACHET, M. E., MEIRON, D. I., ORSZAG, S. A., NICKEL, B. G., MORF, R. H., AND FRISCH, U. Small-scale structure of the taylor–green vortex. *Journal of Fluid Mechanics* 130, -1 (1983), 411.
- [22] BRENNER, G., AL-ZOUBI, A., MUKINOVIC, M., SCHWARZE, H., AND SWOBODA, S. Numerical simulation of surface roughness effects in laminar lubrication using the lattice-boltzmann method. *Journal of Tribology* 129, 3 (2007), 603.

- [23] BROWN, D. L. Performance of under-resolved two-dimensional incompressible flow simulations. *Journal of Computational Physics* 122, 1 (1995), 165–183.
- [24] BROWNLEE, R. A., GORBAN, A. N., AND LEVESLEY, J. Stability and stabilization of the lattice boltzmann method. *Physical review. E, Statistical, nonlinear, and soft matter physics* 75, 3 Pt 2 (2007), 036711.
- [25] BURSTEDDE, C., WILCOX, L. C., AND GHATTAS, O. p4est: Scalable algorithms for parallel adaptive mesh refinement on forests of octrees. *SIAM Journal on Scientific Computing* 33, 3 (2011), 1103–1133.
- [26] CAO, N., CHEN, S., JIN, S., AND MARTÍNEZ, D. Physical symmetry and lattice symmetry in the lattice boltzmann method. *Physical Review E* 55, 1 (1997), R21–R24.
- [27] CASH, J. R., AND KARP, A. H. A variable order runge-kutta method for initial value problems with rapidly varying right-hand sides. *ACM Transactions on Mathematical Software* 16, 3 (1990), 201–222.
- [28] CERCIGNANI, C. *Mathematical Methods in Kinetic Theory*. Springer US, Boston, MA and s.l., 1969.
- [29] CHAPMAN, S., COWLING, T. G., AND PARK, D. The mathematical theory of non-uniform gases. *American Journal of Physics* 30, 5 (1962), 389.
- [30] CHEN, S., AND DOOLEN, G. D. Lattice boltzmann method for fluid flows. *Annual Review of Fluid Mechanics* 30, 1 (1998), 329–364.
- [31] CHENG, M., AND HUNG, K. C. Lattice boltzmann method on nonuniform mesh. *International Journal of Computational Engineering Science* 05, 02 (2004), 291–302.
- [32] CHEW, Y. T., SHU, C., AND NIU, X. D. A new differential lattice boltzmann equation and its application to simulate incompressible flows on non-uniform grids. *Journal of Statistical Physics* 107, 1/2 (2002), 329–342.
- [33] CHIKATAMARLA, S., AND KARLIN, I. Comment on "rectangular lattice boltzmann method". *Physical review. E, Statistical, nonlinear, and soft matter physics* 83, 4 Pt 2 (2011), 048701.
- [34] CHIKATAMARLA, S. S., AND KARLIN, I. V. Entropic lattice boltzmann method for turbulent flow simulations: Boundary conditions. *Physica A: Statistical Mechanics and its Applications* 392, 9 (2013), 1925–1930.
- [35] CHOI, H., AND MOIN, P. Effects of the computational time step on numerical solutions of turbulent flow. *Journal of Computational Physics* 113, 1 (1994), 1–4.

- [36] CHORIN, A. J. A numerical method for solving incompressible viscous flow problems. *Journal of Computational Physics* 2, 1 (1967), 12–26.
- [37] CHORIN, A. J. On the convergence of discrete approximations to the navier-stokes equations. *Mathematics of Computation* 23, 106 (1969), 341.
- [38] CHORIN, A. J. A numerical method for solving incompressible viscous flow problems. *Journal of Computational Physics* 135, 2 (1997), 118–125.
- [39] COX, S. M., AND MATTHEWS, P. C. Exponential time differencing for stiff systems. *Journal of Computational Physics* 176, 2 (2002), 430–455.
- [40] DEBONIS, J. Solutions of the taylor-green vortex problem using high-resolution explicit finite difference methods. In *51st AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition* (Reston, Virginia, 2013), American Institute of Aeronautics and Astronautics.
- [41] DELLAR, P. J. Bulk and shear viscosities in lattice boltzmann equations. *Physical review. E, Statistical, nonlinear, and soft matter physics* 64, 3 Pt 1 (2001), 031203.
- [42] DELLAR, P. J. Incompressible limits of lattice boltzmann equations using multiple relaxation times. *Journal of Computational Physics* 190, 2 (2003), 351–370.
- [43] DELLAR, P. J. Lattice boltzmann algorithms without cubic defects in galilean invariance on standard lattices. *Journal of Computational Physics* 259 (2014), 270–283.
- [44] DENG, X., MAO, M., TU, G., ZHANG, H., AND ZHANG, Y. High-order and high accurate CFD methods and their applications for complex grid problems. *Communications in Computational Physics* 11, 04 (2012), 1081–1102.
- [45] DHAUBHADEL, M. N. Review: CFD applications in the automotive industry. *Journal of Fluids Engineering* 118, 4 (1996), 647.
- [46] D’HUMIÈRES, D. Generalized lattice boltzmann equations. *Progress in Aeronautics and Astronautics* 159 (1992), 450.
- [47] D’HUMIÈRES, D., GINZBURG, I., KRAFCZYK, M., LALLEMAND, P., AND LUO, L.-S. Multiple-relaxation-time lattice boltzmann models in three dimensions. *Philosophical transactions. Series A, Mathematical, physical, and engineering sciences* 360, 1792 (2002), 437–451.
- [48] DI ILIO, G., CHIAPPINI, D., UBERTINI, S., BELLA, G., AND SUCCI, S. Hybrid lattice boltzmann method on overlapping grids. *Physical review. E* 95, 1-1 (2017), 013309.

- [49] DONG, Y.-H., SAGAUT, P., AND MARIE, S. Inertial consistent subgrid model for large-eddy simulation based on the lattice boltzmann method. *Physics of Fluids* 20, 3 (2008), 035104.
- [50] DORSCHNER, B., CHIKATAMARLA, S. S., BÖSCH, F., AND KARLIN, I. V. Grad's approximation for moving and stationary walls in entropic lattice boltzmann simulations. *Journal of Computational Physics* 295 (2015), 340–354.
- [51] DORSCHNER, B., FRAPOLLI, N., CHIKATAMARLA, S. S., AND KARLIN, I. V. Grid refinement for entropic lattice boltzmann models. *Physical review. E* 94, 5-1 (2016), 053311.
- [52] DÜNWEG, B., AND LADD, ANTHONY J. C. Lattice boltzmann simulations of soft matter systems. In *Advanced computer simulation approaches for soft matter sciences*, P. G. Bolhuis, Ed., vol. 221 of *Advances in polymer science*. Springer, Berlin and Heidelberg, 2009, pp. 89–166.
- [53] DUPUIS, A., AND CHOPARD, B. Theory and applications of an alternative lattice boltzmann grid refinement algorithm. *Physical Review E* 67, 6 (2003).
- [54] DÜSTER, A., DEMKOWICZ, L., AND RANK, E. High-order finite elements applied to the discrete boltzmann equation. *International Journal for Numerical Methods in Engineering* 67, 8 (2006), 1094–1121.
- [55] EGGELS, J. G. M., UNGER, F., WEISS, M. H., WESTERWEEL, J., ADRIAN, R. J., FRIEDRICH, R., AND NIEUWSTADT, F. T. M. Fully developed turbulent pipe flow: A comparison between direct numerical simulation and experiment. *Journal of Fluid Mechanics* 268, -1 (1994), 175.
- [56] EHRHARDT, M. *Progress in Computational Physics Volume 3: Novel Trends in Lattice-Boltzmann Methods*, vol. v.3 of *Progress in Computational Physics*. Bentham Science Publishers, Sharjah, 2013.
- [57] EINKEMMER, L., AND OSTERMANN, A. On the error propagation of semi-lagrange and fourier methods for advection problems. *Computers & mathematics with applications (Oxford, England : 1987)* 69, 3 (2015), 170–179.
- [58] ESWARAN, V., AND POPE, S. B. An examination of forcing in direct numerical simulations of turbulence. *Computers & Fluids* 16, 3 (1988), 257–278.
- [59] FAKHARI, A., AND LEE, T. Numerics of the lattice boltzmann method on nonuniform grids: Standard lbm and finite-difference lbm. *Computers & Fluids* 107 (2015), 205–213.
- [60] FALCONE, M., AND FERRETTI, R. Convergence analysis for a class of high-order semi-lagrangian advection schemes. *SIAM Journal on Numerical Analysis* 35, 3 (1998), 909–940.

- [61] FALCONE, M., AND FERRETTI, R. *Semi-Lagrangian Approximation Schemes for Linear and Hamilton—Jacobi Equations*. Society for Industrial and Applied Mathematics, Philadelphia, PA, 2013.
- [62] FEICHTINGER, C., DONATH, S., KÖSTLER, H., GÖTZ, J., AND RÜDE, U. WaLBerla: HPC software design for computational engineering simulations. *Journal of Computational Science* 2, 2 (2011), 105–112.
- [63] FEISTAUER, M., KUCERA, V., AND PROKOPOVA, J. Discontinuous galerkin solution of compressible flow in time-dependent domains. *Mathematics and Computers in Simulation* 80, 8 (2010), 1612–1623.
- [64] FENG, Z.-G., AND MICHAELIDES, E. E. The immersed boundary-lattice boltzmann method for solving fluid–particles interaction problems. *Journal of Computational Physics* 195, 2 (2004), 602–628.
- [65] FEUCHTER, C., AND SCHLEIFENBAUM, W. High-order lattice boltzmann models for wall-bounded flows at finite knudsen numbers. *Physical Review E* 94, 1 (2016).
- [66] FILIPPOVA, O., AND HÄNEL, D. Grid refinement for lattice-BGK models. *Journal of Computational Physics* 147, 1 (1998), 219–228.
- [67] FOYSI, H., AND SARKAR, S. The compressible mixing layer: An les study. *Theoretical and Computational Fluid Dynamics* 24, 6 (2010), 565–588.
- [68] FRAPOLLI, N., CHIKATAMARLA, S. S., AND KARLIN, I. V. Entropic lattice boltzmann model for compressible flows. *Physical review. E, Statistical, nonlinear, and soft matter physics* 92, 6 (2015), 061301.
- [69] GASSNER, G. J., AND BECK, A. D. On the accuracy of high-order discretizations for underresolved turbulence simulations. *Theoretical and Computational Fluid Dynamics* 27, 3-4 (2013), 221–237.
- [70] GEIER, M., GREINER, A., AND KORVINK, J. G. Cascaded digital lattice boltzmann automata for high reynolds number flow. *Physical review. E, Statistical, nonlinear, and soft matter physics* 73, 6 Pt 2 (2006), 066705.
- [71] GEIER, M., GREINER, A., AND KORVINK, J. G. Properties of the cascaded lattice boltzmann automaton. *International Journal of Modern Physics C* 18, 04 (2007), 455–462.
- [72] GEIER, M., GREINER, A., AND KORVINK, J. G. A factorized central moment lattice boltzmann method. *The European Physical Journal Special Topics* 171, 1 (2009), 55–61.

- [73] GEIER, M., AND SCHÖNHERR, M. Esoteric twist: An efficient in-place streaming algorithmus for the lattice boltzmann method on massively parallel hardware. *Computation* 5, 2 (2017), 19.
- [74] GEIER, M., SCHÖNHERR, M., PASQUALI, A., AND KRAFCZYK, M. The cumulant lattice boltzmann equation in three dimensions: Theory and validation. *Computers & Mathematics with Applications* 70, 4 (2015), 507–547.
- [75] GOLSE, F., AND SAINT-RAYMOND, L. The incompressible navier–stokes limit of the boltzmann equation for hard cutoff potentials. *Journal de Mathématiques Pures et Appliquées* 91, 5 (2009), 508–552.
- [76] GORBAN, A. N., AND PACKWOOD, D. J. Enhancement of the stability of lattice boltzmann methods by dissipation control. *Physica A: Statistical Mechanics and its Applications* 414 (2014), 285–299.
- [77] GUAN, H., AND WU, C. Large-eddy simulations of turbulent flows with lattice boltzmann dynamics and dynamical system sub-grid models. *Science in China Series E: Technological Sciences* 52, 3 (2009), 670–679.
- [78] GUO, Z., AND SHU, C. *Lattice Boltzmann method and its applications in engineering*, vol. v. 3 of *Advances in computational fluid dynamics*. World Scientific Pub. Co., Singapore and Hackensack, N.J, 2013.
- [79] GUO, Z., WANG, R., AND XU, K. Discrete unified gas kinetic scheme for all knudsen number flows. ii. thermal compressible case. *Physical review. E, Statistical, nonlinear, and soft matter physics* 91, 3 (2015), 033313.
- [80] GUO, Z., XU, K., AND WANG, R. Discrete unified gas kinetic scheme for all knudsen number flows: low-speed isothermal case. *Physical review. E, Statistical, nonlinear, and soft matter physics* 88, 3 (2013), 033305.
- [81] GUO, Z., AND ZHAO, T. Explicit finite-difference lattice boltzmann method for curvilinear coordinates. *Physical Review E* 67, 6 (2003), 066709.
- [82] GUO, Z., ZHAO, T. S., AND SHI, Y. Preconditioned lattice-boltzmann method for steady flows. *Physical review. E, Statistical, nonlinear, and soft matter physics* 70, 6 Pt 2 (2004), 066706.
- [83] GUZEL, G., AND KOC, I. Simulation of turbulent flows using a finite-volume based lattice boltzmann flow solver. *Communications in Computational Physics* 17, 01 (2015), 213–232.
- [84] HAFEZ, M. M., OSHIMA, K., AND KWAK, D. *Computational Fluid Dynamics Review 2010*. World Scientific, 2010.

- [85] HAN, S.-L., ZHU, P., AND LIN, Z.-Q. Two-dimensional interpolation-supplemented and taylor-series expansion-based lattice boltzmann method and its application. *Communications in Nonlinear Science and Numerical Simulation* 12, 7 (2007), 1162–1171.
- [86] HÄNEL, D. *Molekulare Gasdynamik: Einführung in die kinetische Theorie der Gase und Lattice-Boltzmann-Methoden*. Springer, Berlin [u.a.], 2004.
- [87] HARLOW, F. H., AND WELCH, J. E. Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface. *Physics of Fluids* 8, 12 (1965), 2182.
- [88] HASERT, M. *Multi-scale lattice boltzmann simulations on distributed octrees: Zugl.: Aachen, Techn. Hochsch. Diss., 2013*. Ingenieurwissenschaften. Hut, München, 2014.
- [89] HE, X. Error analysis for the interpolation-supplemented lattice-boltzmann equation scheme. *International Journal of Modern Physics C* 08, 04 (1997), 737–745.
- [90] HE, X., AND DOOLEN, G. Lattice boltzmann method on curvilinear coordinates system: Flow around a circular cylinder. *Journal of Computational Physics* 134, 2 (1997), 306–315.
- [91] HE, X., AND DOOLEN, G. D. Lattice boltzmann method on a curvilinear coordinate system: Vortex shedding behind a circular cylinder. *Physical Review E* 56, 1 (1997), 434–440.
- [92] HE, X., AND LUO, L.-S. Theory of the lattice boltzmann method: From the boltzmann equation to the lattice boltzmann equation. *Physical Review E* 56, 6 (1997), 6811–6817.
- [93] HE, X., LUO, L.-S., AND DEMBO, M. Some progress in lattice boltzmann method. part i. nonuniform mesh grids. *Journal of Computational Physics* 129, 2 (1996), 357–363.
- [94] HEGELE JR, L. A., MATTILA, K., AND PHILIPPI, P. C. Rectangular lattice-boltzmann schemes with BGK-collision operator. *Journal of Scientific Computing* 56, 2 (2013), 230–242.
- [95] HEJRAFAN, K., AND EZZATNESHAN, E. Implementation of a high-order compact finite-difference lattice boltzmann method in generalized curvilinear coordinates. *Journal of Computational Physics* 267 (2014), 28–49.
- [96] HEJRAFAN, K., AND GHAFARIAN, A. A spectral difference lattice boltzmann method for solution of inviscid compressible flows on structured grids. *Computers & Mathematics with Applications* 72, 5 (2016), 1341–1368.

- [97] HEJRANFAR, K., AND HAJIHASSANPOUR, M. Chebyshev collocation spectral lattice boltzmann method for simulation of low-speed flows. *Physical review. E, Statistical, nonlinear, and soft matter physics* 91, 1 (2015), 013301.
- [98] HEJRANFAR, K., AND HAJIHASSANPOUR, M. Chebyshev collocation spectral lattice boltzmann method in generalized curvilinear coordinates. *Computers & Fluids* 146 (2017), 154–173.
- [99] HEJRANFAR, K., SAADAT, M. H., AND TAHERI, S. High-order weighted essentially nonoscillatory finite-difference formulation of the lattice boltzmann method in generalized curvilinear coordinates. *Physical review. E* 95, 2-1 (2017), 023314.
- [100] HEROUX, M. A., PHIPPS, E. T., SALINGER, A. G., THORNQUIST, H. K., TUMINARO, R. S., WILLENBRING, J. M., WILLIAMS, A., STANLEY, K. S., BARTLETT, R. A., HOWLE, V. E., HOEKSTRA, R. J., HU, J. J., KOLDA, T. G., LEHOUCQ, R. B., LONG, K. R., AND PAWLOWSKI, R. P. An overview of the trilinos project. *ACM Transactions on Mathematical Software* 31, 3 (2005), 397–423.
- [101] HESTHAVEN, J. S., AND WARBURTON, T. *Nodal discontinuous Galerkin methods: Algorithms, analysis, and applications*, vol. 54 of *Texts in applied mathematics*. Springer, New York, 2008.
- [102] HEUVELINE, V., AND LATT, J. The OpenLB project: An open source and object oriented implementation of lattice boltzmann methods. *International Journal of Modern Physics C* 18, 04 (2007), 627–634.
- [103] HIGUERA, F. J., AND JIMÉNEZ, J. Boltzmann approach to lattice gas simulations. *Europhysics Letters (EPL)* 9, 7 (1989), 663–668.
- [104] HIGUERA, F. J., SUCCI, S., AND BENZI, R. Lattice gas dynamics with enhanced collisions. *EPL (Europhysics Letters)* 9, 4 (1989), 345.
- [105] HIRSCHL, E. H., AND KRAUSE, E., Eds. *100 Volumes of 'Notes on Numerical Fluid Mechanics': 40 Years of Numerical Fluid Mechanics and Aerodynamics in Retrospect*, 1. Aufl. ed., vol. v.v. 100 of *Notes on Numerical and Fluid Dynamics, 100*. Springer-Verlag, s.l., 2009.
- [106] HOCHBRUCK, M., LUBICH, C., AND SELHOFER, H. Exponential integrators for large systems of differential equations. *SIAM Journal on Scientific Computing* 19, 5 (1998), 1552–1574.
- [107] HOLDYCH, D. J., NOBLE, D. R., GEORGIADIS, J. G., AND BUCKIUS, R. O. Truncation error analysis of lattice boltzmann methods. *Journal of Computational Physics* 193, 2 (2004), 595–619.

- [108] HSU, A. T., YANG, T., LOPEZ, I., AND ECER, A. A review of lattice boltzmann models for compressible flows. In *Parallel Computational Fluid Dynamics 2003*, B. Chetverushkin, J. Periaux, and N. Satofuka, Eds. Elsevier professional, s.l., 2004, pp. 19–28.
- [109] HUANG, J., YANG, C., AND CAI, X.-C. A fully implicit method for lattice boltzmann equations. *SIAM Journal on Scientific Computing* 37, 5 (2015), S291–S313.
- [110] HÜBNER, T., AND TUREK, S. Efficient monolithic simulation techniques for the stationary lattice boltzmann equation on general meshes. *Computing and Visualization in Science* 13, 3 (2010), 129–143.
- [111] JOHN, V. *Large Eddy Simulation of Turbulent Incompressible Flows: Analytical and Numerical Results for a Class of LES Models*, vol. 34 of *Lecture Notes in Computational Science and Engineering*. Springer, Berlin and Heidelberg, 2004.
- [112] JOPPICH, W., AND POTT, S. Results obtained with a semi-lagrangian mass-integrating transport algorithm by using the GME grid. In *Large-Scale Scientific Computing: 6th International Conference, LSSC 2007, Sozopol, Bulgaria, June 5-9, 2007. Revised Papers*, I. Lirkov, S. Margenov, and J. Waśniewski, Eds. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008, pp. 417–424.
- [113] JUNK, M., KLAR, A., AND LUO, L.-S. Asymptotic analysis of the lattice boltzmann equation. *Journal of Computational Physics* 210, 2 (2005), 676–704.
- [114] JUNK, M., AND YANG, Z. Convergence of lattice boltzmann methods for navier–stokes flows in periodic and bounded domains. *Numerische Mathematik* 112, 1 (2009), 65–87.
- [115] KARLIN, I., BÖSCH, F., CHIKATAMARLA, S., AND SUCCI, S. Entropy-assisted computing of low-dissipative systems. *Entropy* 17, 12 (2015), 8099–8110.
- [116] KARLIN, I. V. Persönliche Kommunikation.
- [117] KARLIN, I. V., BOSCH, F., AND CHIKATAMARLA, S. S. Gibbs’ principle for the lattice-kinetic theory of fluid dynamics. *Physical Review E, Statistical, nonlinear, and soft matter physics* 90, 3 (2014), 031302.
- [118] KARLIN, I. V., FERRANTE, A., AND ÖTTINGER, H. C. Perfect entropy functions of the lattice boltzmann method. *Europhysics Letters (EPL)* 47, 2 (1999), 182–188.
- [119] KARLIN, I. V., AND SUCCI, S. Equilibria for discrete kinetic equations. *Physical Review E* 58, 4 (1998), R4053–R4056.

- [120] KARLIN, I. V., SUCCI, S., AND CHIKATAMARLA, S. S. Comment on "numerics of the lattice boltzmann method: effects of collision models on the lattice boltzmann simulations". *Physical review. E, Statistical, nonlinear, and soft matter physics* 84, 6 Pt 2 (2011), 068701.
- [121] KARLIN, I. V., SUCCI, S., AND ORSZAG, S. Lattice boltzmann method for irregular grids. *Physical Review Letters* 82, 26 (1999), 5245–5248.
- [122] KATAOKA, T., AND TSUTAHARA, M. Lattice boltzmann model for the compressible navier-stokes equations with flexible specific-heat ratio. *Physical review. E, Statistical, nonlinear, and soft matter physics* 69, 3 Pt 2 (2004), 035701.
- [123] KIM, J., MOIN, P., AND MOSER, R. Turbulence statistics in fully developed channel flow at low reynolds number. *Journal of Fluid Mechanics* 177, -1 (1987), 133.
- [124] KRAFCZYK, M., TÖLKE, J., AND LUO, L.-S. Large-eddy simulations with a multiple-relaxation-time lbe model. *International Journal of Modern Physics B* 17, 01n02 (2003), 33–39.
- [125] KRÄMER, A., KÜLLMER, K., REITH, D., JOPPICH, W., AND FOYSI, H. Semi-lagrangian off-lattice boltzmann method for weakly compressible flows. *Physical Review E* 95, 2 (2017).
- [126] KRÄMER, A., WILDE, D., KÜLLMER, K., REITH, D., FOYSI, H., AND JOPPICH, W. NATrium: an extensible framework for lattice boltzmann simulations on irregular grids. *Eingereicht bei: Computers and Mathematics with Applications*. (2017).
- [127] KRÄMER, A., WILDE, D., KÜLLMER, K., REITH, D., JOPPICH, W., AND FOYSI, H. Stability-enhanced and efficient collision models for the lattice boltzmann method. *Eingereicht bei: Journal of Computational Physics* (2017).
- [128] KRONBICHLER, M., AND WALL, W. A. A performance comparison of continuous and discontinuous galerkin methods with fast multigrid solvers, 2016.
- [129] KRÜGER, T., KUSUMAATMAJA, H., KUZMIN, A., SHARDT, O., SILVA, G., AND VIGGEN, E. M. *The Lattice Boltzmann Method*. Springer International Publishing, Cham, 2017.
- [130] KÜLLMER, K., KRÄMER, A., REITH, D., JOPPICH, W., AND FOYSI, H. Numerical optimisation of the pseudopotential-based lattice boltzmann method. *Journal of Computational Science* 17 (2016), 384–393.
- [131] KUNDU, P. K., AND COHEN, I. M. *Fluid mechanics*, 4th ed. ed. Academic Press, Amsterdam and Boston, 2008.

- [132] KWAK, D., AND KIRIS, C. C. Artificial compressibility method. In *Computation of viscous incompressible flows*, D. Kwak and C. C. Kiris, Eds., Scientific Computation. Springer, Dordrecht u.a., 2011, pp. 41–77.
- [133] LADD, ANTHONY J. C. Numerical simulations of particulate suspensions via a discretized boltzmann equation. part 2. numerical results. *Journal of Fluid Mechanics* 271 (1994), 311.
- [134] LAGRAVA, D., MALASPINAS, O., LATT, J., AND CHOPARD, B. Advances in multi-domain lattice boltzmann grid refinement. *Journal of Computational Physics* 231, 14 (2012), 4808–4822.
- [135] LALLEMAND, P., AND LUO, L.-S. Theory of the lattice boltzmann method: Dispersion, dissipation, isotropy, galilean invariance, and stability. *Physical Review E* 61, 6 (2000), 6546–6562.
- [136] LAMBERT, J. D. *Numerical methods for ordinary differential systems: The initial value problem*. Wiley, Chichester u.a., 1991.
- [137] LAMMERS, P., BERONOV, K. N., VOLKERT, R., BRENNER, G., AND DURST, F. Lattice BGK direct numerical simulation of fully developed turbulence in incompressible plane channel flow. *Computers & Fluids* 35, 10 (2006), 1137–1153.
- [138] LATT, J. *Hydrodynamic Limit of lattice Boltzmann equations*. Dissertation, Université de Genève, Genf, 2007.
- [139] LATT, J., AND CHOPARD, B. Lattice boltzmann method with regularized pre-collision distribution functions. *Mathematics and Computers in Simulation* 72, 2-6 (2006), 165–168.
- [140] LATT, J., CHOPARD, B., MALASPINAS, O., DEVILLE, M., AND MICHLER, A. Straight velocity boundaries in the lattice boltzmann method. *Physical Review E* 77, 5 (2008), 056703.
- [141] LEE, T., AND LIN, C.-L. A characteristic galerkin method for discrete boltzmann equation. *Journal of Computational Physics* 171, 1 (2001), 336–356.
- [142] LEE, T., AND LIN, C.-L. An eulerian description of the streaming process in the lattice boltzmann equation. *Journal of Computational Physics* 185, 2 (2003), 445–471.
- [143] LETALLEUR, N., PLOURABOUÉ, F., AND PRAT, M. Average flow model of rough surface lubrication: Flow factors for sinusoidal surfaces. *Journal of Tribology* 124, 3 (2002), 539.

- [144] LI, W., AND LUO, L.-S. An implicit block lu-sgs finite-volume lattice-boltzmann scheme for steady flows on arbitrary unstructured meshes. *Journal of Computational Physics* 327 (2016), 503–518.
- [145] LI, Y., LEBOEUF, E., AND BASU, P. Least-squares finite-element scheme for the lattice boltzmann method on an unstructured mesh. *Physical Review E* 72, 4 (2005).
- [146] LI, Y., LEBOEUF, E. J., AND BASU, P. K. Least-squares finite-element lattice boltzmann method. *Physical review. E, Statistical, nonlinear, and soft matter physics* 69, 6 Pt 2 (2004), 065701.
- [147] LI, Y., LEBOEUF, E. J., AND BASU, P. K. Use of a least squares finite element lattice boltzmann method to study fluid flow and mass transfer processes. In *Computational science - ICCS 2005*, V. S. Sunderam, G. D. van Albada, P. M. A. Sloot, and J. J. Dongarra, Eds., vol. 3514 of *Lecture Notes in Computer Science*. Springer, Berlin, 2005, pp. 172–179.
- [148] LINTERMANN, A., EITEL-AMOR, G., MEINKE, M., AND SCHRÖDER, W. Lattice-boltzmann solutions with local grid refinement for nasal cavity flows. In *New Results in Numerical and Experimental Fluid Mechanics VIII*, A. Dillmann, G. Heller, H.-P. Kreplin, W. Nitsche, and I. Peltzer, Eds., vol. 121 of *Notes on Numerical Fluid Mechanics and Multidisciplinary Design*. Springer, Berlin and Heidelberg, 2013, pp. 583–590.
- [149] LIPAVSKII, M. V., TOLSTYKH, A. I., AND CHIGIREV, E. N. Numerical simulation of shear layer instability using a scheme with ninth-order multioperator approximations. *Computational Mathematics and Mathematical Physics* 53, 3 (2013), 296–310.
- [150] LUO, L.-S. Reply to "comment on 'numerics of the lattice boltzmann method: Effects of collision models on the lattice boltzmann simulations'". *Physical review. E, Statistical, nonlinear, and soft matter physics* 86, 4 Pt 2 (2012), 048701.
- [151] LUO, L.-S., LIAO, W., CHEN, X., PENG, Y., AND ZHANG, W. Numerics of the lattice boltzmann method: effects of collision models on the lattice boltzmann simulations. *Physical review. E, Statistical, nonlinear, and soft matter physics* 83, 5 Pt 2 (2011), 056710.
- [152] MALASPINAS, O., AND SAGAUT, P. Consistent subgrid scale modelling for lattice boltzmann methods. *Journal of Fluid Mechanics* 700 (2012), 514–542.
- [153] MALEVSKY, A. V., AND THOMAS, S. J. Parallel algorithms for semi-lagrangian advection. *International Journal for Numerical Methods in Fluids* 25, 4 (1997), 455–473.

- [154] MARUSIC, I., MATHIS, R., AND HUTCHINS, N. High reynolds number effects in wall turbulence. *International Journal of Heat and Fluid Flow* 31, 3 (2010), 418–428.
- [155] MATTILA, K. K., HEGELE, JR, L. A., AND PHILIPPI, P. C. Investigation of an entropic stabilizer for the lattice-boltzmann method. *Physical review. E, Statistical, nonlinear, and soft matter physics* 91, 6 (2015), 063010.
- [156] MATTILA, K. K., PHILIPPI, P. C., AND HEGELE, L. A. High-order regularization in lattice-boltzmann equations. *Physics of Fluids* 29, 4 (2017), 046103.
- [157] MAZLOOMI M, A., CHIKATAMARLA, S. S., AND KARLIN, I. V. Entropic lattice boltzmann method for multiphase flows. *Physical Review Letters* 114, 17 (2015), 174502.
- [158] MAZZEO, M. D. Fast discontinuous galerkin lattice-boltzmann simulations on GPUs via maximal kernel fusion. *Computer Physics Communications* 184, 3 (2013), 537–549.
- [159] MCNAMARA, G., AND ZANETTI, G. Use of the boltzmann equation to simulate lattice-gas automata. *Physical Review Letters* 61, 20 (1988), 2332–2335.
- [160] MEI, R., AND SHYY, W. On the finite difference-based lattice boltzmann method in curvilinear coordinates. *Journal of Computational Physics* 143, 2 (1998), 426–448.
- [161] MIN, M., AND LEE, T. A spectral-element discontinuous galerkin lattice boltzmann method for nearly incompressible flows. *Journal of Computational Physics* 230, 1 (2011), 245–259.
- [162] MINION, M. L., AND BROWN, D. L. Performance of under-resolved two-dimensional incompressible flow simulations, ii. *Journal of Computational Physics* 138, 2 (1997), 734–765.
- [163] MIRZAEI, M., AND POOZESH, A. Simulation of fluid flow in a body-fitted grid system using the lattice boltzmann method. *Physical review. E, Statistical, nonlinear, and soft matter physics* 87, 6 (2013), 063312.
- [164] MISZTAL, M. K., HERNANDEZ-GARCIA, A., MATIN, R., MÜTER, D., JHA, D., SØRENSEN, H. O., AND MATHIESEN, J. Simulating anomalous dispersion in porous media using the unstructured lattice boltzmann method. *Frontiers in Physics* 3 (2015).
- [165] MISZTAL, M. K., HERNANDEZ-GARCIA, A., MATIN, R., SØRENSEN, H. O., AND MATHIESEN, J. *Detailed analysis of the lattice Boltzmann method on unstructured grids*. 2014.

- [166] MONTESSORI, A., FALCUCCI, G., PRESTININZI, P., LA ROCCA, M., AND SUCCI, S. Regularized lattice bhatnagar-gross-krook model for two- and three-dimensional cavity flow simulations. *Physical review. E, Statistical, nonlinear, and soft matter physics* 89, 5 (2014), 053317.
- [167] MONTESSORI, A., LA ROCCA, M., FALCUCCI, G., AND SUCCI, S. Regularized lattice BGK versus highly accurate spectral methods for cavity flow simulations. *International Journal of Modern Physics C* 25, 12 (2014), 1441003.
- [168] MOSER, R. D., KIM, J., AND MANSOUR, N. N. Direct numerical simulation of turbulent channel flow up to $Re_\tau = 590$. *Physics of Fluids* 11, 4 (1999), 943.
- [169] MUSAVI, S. H., AND ASHRAFIZAADEH, M. Meshless lattice boltzmann method for the simulation of fluid flows. *Physical review. E, Statistical, nonlinear, and soft matter physics* 91, 2 (2015), 023310.
- [170] NAMBURI, M., KRITHIVASAN, S., AND ANSUMALI, S. Crystallographic lattice boltzmann method. *Scientific Reports* 6 (2016), 27172.
- [171] NANNELLI, F., AND SUCCI, S. The lattice boltzmann equation on irregular lattices. *Journal of Statistical Physics* 68, 3-4 (1992), 401–407.
- [172] NIU, X. D., SHU, C., CHEW, Y. T., AND PENG, Y. A momentum exchange-based immersed boundary-lattice boltzmann method for simulating incompressible viscous flows. *Physics Letters A* 354, 3 (2006), 173–182.
- [173] PANTANO, C., AND SARKAR, S. A study of compressibility effects in the high-speed turbulent shear layer using direct simulation. *Journal of Fluid Mechanics* 451 (2002).
- [174] PARESCHI, G., FRAPOLLI, N., CHIKATAMARLA, S. S., AND KARLIN, I. V. Conjugate heat transfer with the entropic lattice boltzmann method. *Physical review. E* 94, 1-1 (2016), 013305.
- [175] PATEL, S., AND LEE, T. A new splitting scheme to the discrete boltzmann equation for non-ideal gases on non-uniform meshes. *Journal of Computational Physics* 327 (2016), 799–809.
- [176] PATEL, S., MIN, M., AND LEE, T. A spectral-element discontinuous galerkin thermal lattice boltzmann method for conjugate heat transfer applications. *International Journal for Numerical Methods in Fluids* 82, 12 (2016), 932–952.
- [177] PATEL, S. S., MIN, M., UGA, K. C., AND LEE, T. A spectral-element discontinuous galerkin lattice boltzmann method for simulating natural convection heat transfer in a horizontal concentric annulus. *Computers & Fluids* 95 (2014), 197–209.

- [178] PATIL, D. V. Chapman–enskog analysis for finite-volume formulation of lattice boltzmann equation. *Physica A: Statistical Mechanics and its Applications* 392, 12 (2013), 2701–2712.
- [179] PATIL, D. V., AND LAKSHMISHA, K. N. Finite volume tvd formulation of lattice boltzmann simulation on unstructured mesh. *Journal of Computational Physics* 228, 14 (2009), 5262–5279.
- [180] PENG, C., MIN, H., GUO, Z., AND WANG, L.-P. A hydrodynamically-consistent MRT lattice boltzmann model on a 2d rectangular grid. *Journal of Computational Physics* 326 (2016), 893–912.
- [181] PENG, G., XI, H., AND CHOU, S.-H. On boundary conditions in the finite volume lattice boltzmann method on unstructured meshes. *International Journal of Modern Physics C* 10, 06 (1999), 1003–1016.
- [182] PENG, G., XI, H., DUNCAN, C., AND CHOU, S.-H. Finite volume scheme for the lattice boltzmann method on unstructured meshes. *Physical Review E* 59, 4 (1999), 4675–4682.
- [183] PESKIN, C. S. Flow patterns around heart valves: A numerical method. *Journal of Computational Physics* 10, 2 (1972), 252–271.
- [184] PREMNATH, K. N., PATTISON, M. J., AND BANERJEE, S. Dynamic subgrid scale modeling of turbulent flows using lattice-boltzmann method. *Physica A: Statistical Mechanics and its Applications* 388, 13 (2009), 2640–2658.
- [185] PREMNATH, K. N., PATTISON, M. J., AND BANERJEE, S. Generalized lattice boltzmann equation with forcing term for computation of wall-bounded turbulent flows. *Physical review. E, Statistical, nonlinear, and soft matter physics* 79, 2 Pt 2 (2009), 026703.
- [186] QIAN, Y. H., D’HUMIÈRES, D., AND LALLEMAND, P. Lattice BGK models for navier-stokes equation. *EPL (Europhysics Letters)* 17, 6 (1992), 479.
- [187] QU, K. U., SHU, C., AND CHEW, Y. T. An isoparametric transformation-based interpolation-supplemented lattice boltzmann method and its application. *Modern Physics Letters B* 24, 13 (2010), 1315–1318.
- [188] REIDER, M. B., AND STERLING, J. D. Accuracy of discrete-velocity BGK models for the simulation of the incompressible navier-stokes equations. *Computers & Fluids* 24, 4 (1995), 459–467.
- [189] ROHDE, M., KANDHAI, D., DERKSEN, J. J., AND VAN DEN AKKER, H. E. A. A generic, mass conservative local grid refinement technique for lattice-boltzmann schemes. *International Journal for Numerical Methods in Fluids* 51, 4 (2006), 439–468.

- [190] SAGAUT, P. Toward advanced subgrid models for lattice-boltzmann-based large-eddy simulation: Theoretical formulations. *Computers & Mathematics with Applications* 59, 7 (2010), 2194–2199.
- [191] SAINT-RAYMOND, L. From the BGK model to the navier–stokes equations. *Annales Scientifiques de l'École Normale Supérieure* 36, 2 (2003), 271–317.
- [192] SCHÄLING, B. *Die Boost C++ Bibliotheken: [eine Einführung in 50 Bibliotheken mit über 300 Beispielen]*. XML Press, Laguna Hills, Calif., 2011.
- [193] SCHRANNER, F. S., DOMARADZKI, J. A., HICKEL, S., AND ADAMS, N. A. Assessing the numerical dissipation rate and viscosity in numerical simulations of fluid flows. *Computers & Fluids* 114 (2015), 84–97.
- [194] SHAN, X., AND CHEN, H. Lattice boltzmann model for simulating flows with multiple phases and components. *Physical Review E* 47, 3 (1993), 1815–1819.
- [195] SHI, X., LIN, J., AND YU, Z. Discontinuous galerkin spectral element lattice boltzmann method on triangular element. *International Journal for Numerical Methods in Fluids* 42, 11 (2003), 1249–1261.
- [196] SHU, C., CHEW, Y., AND NIU, X. Least-squares-based lattice boltzmann method: A meshless approach for simulation of flows with complex geometry. *Physical Review E* 64, 4 (2001), 045701.
- [197] SHU, C., NIU, X. D., AND CHEW, Y. T. Taylor-series expansion and least-squares-based lattice boltzmann method: Two-dimensional formulation and its applications. *Physical Review E* 65, 3 Pt 2B (2002), 036708.
- [198] SI, H., AND SHI, Y. Study on lattice boltzmann method/large eddy simulation and its application at high reynolds number flow. *Advances in Mechanical Engineering* 7, 3 (2015), 168781401557382.
- [199] SIDJE, R. B. Expokit: A software package for computing matrix exponentials. *ACM Transactions on Mathematical Software* 24, 1 (1998), 130–156.
- [200] SIEBERT, D. N., PHILIPPI, P. C., AND MATTILA, K. K. Consistent lattice boltzmann equations for phase transitions. *Physical review. E, Statistical, nonlinear, and soft matter physics* 90, 5-1 (2014), 053310.
- [201] SKORDOS, P. A. Initial and boundary conditions for the lattice boltzmann method. *Physical Review E* 48, 6 (1993), 4823–4842.
- [202] SOFONEA, V. Discontinuous galerkin schemes for isothermal lattice boltzmann models in one dimension. *International Journal of Modern Physics C* 19, 05 (2008), 677–688.

- [203] SOFONEA, V., AND SEKERKA, R. F. Viscosity of finite difference lattice boltzmann models. *Journal of Computational Physics* 184, 2 (2003), 422–434.
- [204] STANFORTH, A., AND CÔTÉ, J. Semi-lagrangian integration schemes for atmospheric models—a review. *Monthly Weather Review* 119, 9 (1991), 2206–2223.
- [205] STERLING, J. D., AND CHEN, S. Stability analysis of lattice boltzmann methods. *Journal of Computational Physics* 123, 1 (1996), 196–206.
- [206] STROUSTRUP, B. *The C++ programming language: [C++ 11]*, 4. ed. ed. Addison-Wesley, Upper Saddle River, NJ, 2013.
- [207] SUCCI, S. *The lattice Boltzmann equation for fluid dynamics and beyond*. Numerical mathematics and scientific computation. Clarendon Press and Oxford University Press, Oxford, New York, 2001.
- [208] SUCCI, S. Lattice boltzmann 2038. *EPL (Europhysics Letters)* 109, 5 (2015), 50001.
- [209] SUCCI, S., AND BENZI, R. Lattice boltzmann equation for quantum mechanics. *Physica D: Nonlinear Phenomena* 69, 3-4 (1993), 327–332.
- [210] SUN, W.-Y., AND YEH, K.-S. A general semi-lagrangian advection scheme employing forward trajectories. *Quarterly Journal of the Royal Meteorological Society* 123, 544 (1997), 2463–2476.
- [211] SWIFT, OSBORN, AND YEOMANS. Lattice boltzmann simulation of nonideal fluids. *Physical review letters* 75, 5 (1995), 830–833.
- [212] TAYLOR, G. I., AND GREEN, A. E. Mechanism of the production of small eddies from large ones. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 158, 895 (1937), 499–521.
- [213] TOSCHI, F., AND SUCCI, S. Lattice boltzmann method at finite knudsen numbers. *Europhysics Letters (EPL)* 69, 4 (2005), 549–555.
- [214] TSUKAHARA, T., SEKI, Y., KAWAMURA, H., AND TOCHIO, D. Dns of turbulent channel flow at very low reynolds numbers. *arXiv:1406.0248 [physics.flu-dyn]* (2014).
- [215] TSUTAHARA, M. The finite-difference lattice boltzmann method and its application in computational aero-acoustics. *Fluid Dynamics Research* 44, 4 (2012), 045507.
- [216] UBERTINI, S., AND SUCCI, S. A generalised lattice boltzmann equation on unstructured grids. *Communications in Computational Physics* 3, 2 (2008), 342–356.

- [217] UGA, K. C., MIN, M., LEE, T., AND FISCHER, P. F. Spectral-element discontinuous galerkin lattice boltzmann simulation of flow past two cylinders in tandem with an exponential time integrator. *Computers & Mathematics with Applications* 65, 2 (2013), 239–251.
- [218] VILLANI, C. A review of mathematical topics in collisional kinetic theory. In *Handbook of mathematical fluid dynamics*, S. Friedlander and D. Serre, Eds., vol. 1 of *Handbook of Mathematical Fluid Dynamics*. Elsevier, Amsterdam and Boston, 2002, pp. 71–74.
- [219] WANG, L.-P. Persönliche Kommunikation.
- [220] WANG, L.-P., AND MAXEY, M. R. Settling velocity and concentration distribution of heavy particles in homogeneous isotropic turbulence. *Journal of Fluid Mechanics* 256, -1 (1993), 27.
- [221] WANG, L.-P., MIN, H., PENG, C., GENEVA, N., AND GUO, Z. A lattice-boltzmann scheme of the navier–stokes equation on a three-dimensional cuboid lattice. *Computers & Mathematics with Applications* (2016).
- [222] WANG, P., AND GUO, Z. A semi-implicit gas-kinetic scheme for smooth flows. *Computer Physics Communications* 205 (2016), 22–31.
- [223] WANG, P., ZHU, L., GUO, Z., AND XU, K. A comparative study of LBE and DUGKS methods for nearly incompressible flows. *Communications in Computational Physics* 17, 03 (2015), 657–681.
- [224] WARDLE, K. E., AND LEE, T. Finite element lattice boltzmann simulations of free surface flow in a concentric cylinder. *Computers & Mathematics with Applications* 65, 2 (2013), 230–238.
- [225] WARSHEL, A., AND LEVITT, M. Theoretical studies of enzymic reactions: dielectric, electrostatic and steric stabilization of the carbonium ion in the reaction of lysozyme. *Journal of molecular biology* 103, 2 (1976), 227–249.
- [226] WEIDENDORFER, J. Sequential performance analysis with callgrind and kcachegrind. In *Tools for High Performance Computing*, M. Resch, R. Keller, and V. Himmler, Eds. Springer-Verlag, s.l., 2008, pp. 93–113.
- [227] WESSELING, P. *Principles of Computational Fluid Dynamics*, vol. 29 of *Springer Series in Computational Mathematics*. Springer-Verlag Berlin Heidelberg, Berlin Heidelberg, 2010.
- [228] WILSON, T. *Stabilization, Extension and Unification of the Lattice Boltzmann Method Using Information Theory*. Dissertation, University of Toronto, Toronto, 2016.

- [229] WOLF-GLADROW, D. A. *Lattice-gas cellular automata and lattice Boltzmann models: An introduction*, vol. 1725 of *Lecture notes in mathematics*. Springer, New York, 2000.
- [230] WU, J., AND SHU, C. Implicit velocity correction-based immersed boundary-lattice boltzmann method and its applications. *Journal of Computational Physics* 228, 6 (2009), 1963–1979.
- [231] XI, H., PENG, G., AND CHOU, S.-H. Finite-volume lattice boltzmann method. *Physical Review E* 59, 5 (1999), 6202–6205.
- [232] XI, H., PENG, G., AND CHOU, S.-H. Finite-volume lattice boltzmann schemes in two and three dimensions. *Physical Review E* 60, 3 (1999), 3380–3388.
- [233] XU, A. Two-dimensional finite-difference lattice boltzmann method for the complete navier-stokes equations of binary fluids. *EPL (Europhysics Letters)* 69, 2 (2004), 214.
- [234] YAN-BIAO, G., AI-GUO, X., GUANG-CAI, Z., PING, Z., LEI, Z., AND YING-JUN, L. Finite-difference lattice boltzmann scheme for high-speed compressible flow: Two-dimensional case. *Communications in Theoretical Physics* 50, 1 (2008), 201.
- [235] ZADEHGOL, A., ASHRAFIZAADEH, M., AND MUSAVI, S. H. A nodal discontinuous galerkin lattice boltzmann method for fluid flow problems. *Computers & Fluids* 105 (2014), 58–65.
- [236] ZHANG, J., JOHNSON, P. C., AND POPEL, A. S. An immersed boundary lattice boltzmann approach to simulate deformable liquid capsules and its application to microscopic blood flows. *Physical biology* 4, 4 (2007), 285–295.
- [237] ZHANG, R., CHEN, H., QIAN, Y., AND CHEN, S. Effective volumetric lattice boltzmann scheme. *Physical Review E* 63, 5 (2001).
- [238] ZHOU, J. G. Rectangular lattice boltzmann method. *Physical review. E, Statistical, nonlinear, and soft matter physics* 81, 2 Pt 2 (2010), 026705.
- [239] ZHOU, J. G. MRT rectangular lattice boltzmann method. *International Journal of Modern Physics C* 23, 05 (2012), 1250040.
- [240] ZHOU, Z., AND MA, J. Lattice boltzmann methods for solving partial differential equations of exotic option pricing. *Frontiers of Mathematics in China* 11, 1 (2016), 237–254.
- [241] ZHU, L., WANG, P., AND GUO, Z. Performance evaluation of the general characteristics based off-lattice boltzmann scheme and dugks for low speed continuum flows. *Journal of Computational Physics* 333 (2017), 227–246.

-
- [242] ZHUO, C., AND ZHONG, C. LES-based filter-matrix lattice boltzmann model for simulating fully developed turbulent channel flow. *International Journal of Computational Fluid Dynamics* 30, 7-10 (2016), 543–553.
- [243] ZONG, Y., PENG, C., GUO, Z., AND WANG, L.-P. Designing correct fluid hydrodynamics on a rectangular grid using MRT lattice boltzmann approach. *Computers & Mathematics with Applications* 72, 2 (2016), 288–310.
- [244] ZOU, Q., AND HE, X. On pressure and velocity boundary conditions for the lattice boltzmann BGK model. *Physics of Fluids* 9, 6 (1997), 1591.