

UNTERSUCHUNG VON MD-SIMULATIONEN

im Rahmen von Hybrid-Schemata
zur Beschreibung komplexer Flüssigkeiten



JOHANNES GUTENBERG
UNIVERSITÄT MAINZ

DISSERTATION

zur Erlangung des Grades
„Doktor der Naturwissenschaften“
am Fachbereich Physik, Mathematik und Informatik (FB 08)
der Johannes Gutenberg-Universität Mainz

vorgelegt von

Stefanie Stalter
geboren in Worms

Mainz, den 28. August 2018

Eingereicht am: 28. August 2018
Tag der mündlichen Prüfung: 26. November 2018

Stefanie Stalter
KOMET 331
Institut für Physik
Staudingerweg 9
Johannes Gutenberg-Universität D-55099 Mainz
ststalte@uni-mainz.de

ABSTRACT

Several types of simulations permit an insight into the dynamics and the intrinsic properties of a system, according to the underlying algorithms and simulated time and length scales. Depending on the type of simulation and the coarse graining of the physical system, we gain access to a specific set of dynamics, and therefore a specific set of physical properties. Each type of simulation has its respective advantages and disadvantages.

To profit from the advantages and simultaneously compensate the disadvantages of certain simulation types, the Heterogeneous Multiscale Method combines the molecular accuracy of a particle-based Molecular Dynamics simulation with the computational efficiency of a continuum-type method, Computational Fluid Dynamics. In the present thesis, we present a multiscale method to accurately describe the flow behaviour of soft matter fluids on large length scales.

For every time step of the continuum simulation, one or more time-intensive simulations at the micro scale are initialized to obtain the necessary information. Thus, the micro-scale simulations typically slow down the whole method, constituting a bottleneck to the hybrid scheme. To avoid this situation or at least to reduce the needed time to conduct such a simulation, the goal of this study is to optimize the necessary simulation parameters and underlying algorithms at the micro scale to accelerate the hybrid scheme calculations.

We investigate a variety of thermostat candidates for temperature control and their respective parameter space. To determine suitable values for the parameters, we study the pressure and the viscosity, which can be derived from the stress tensor. For the Lowe-Andersen thermostat, the parameter choice is critical, since it has a major influence on the dynamics of the system, the effects on the viscosity are surveyed in detail. With careful choice of the thermostat parameters, one can simulate a fluid with higher viscosity and thus control the dynamics, avoiding the use of a complex hydrodynamic simulation scheme, such as Multiple Particle Collision Dynamics.

The isokinetic thermostat is also studied in detail. With help of this temperature controlling algorithm, the impact and influence of simulational parameters such as the timestep and the simulation box length on pressure and viscosity are studied. We report that several small-scale simulations are preferable to a large-scale system with complex parallelisation.

Both simple Weeks-Chandler-Andersen particles and small polymer chains using a Kremer-Grest model are simulated. The latter differ in their viscous behaviour depending on the strain applied on the system, while the monomer system exhibit Newtonian behaviour. Parameters such as the monomer density of the polymer system as well as the chain length and their influence on the viscosity are examined.

Apart from the results of the investigations into the influence of the parameters, the procedures to refine the data and the fitting methods are presented. These algorithms are applied to the results extracted from the MD simulation. By application of these methods, the amount of required simulations can be reduced. In addition, first results based on the complete Heterogeneous Multiscale Method are shown.

ZUSAMMENFASSUNG

Verschiedene Simulationsarten erlauben einen Einblick in die Dynamik unterschiedlicher Größenskalen. Je nach Art der Simulation und der Vergrößerung eines physikalischen Systems wird Zugriff auf eine spezielle Dynamik und damit verbundene Größen ermöglicht. Jede Simulationsart hat entsprechende Vor- sowie Nachteile in der Anwendung und Umsetzung.

Um diese Nachteile auszugleichen und von den jeweiligen Vorteilen zu profitieren, kombiniert die Heterogene Multiskalen-Methode die molekulare Genauigkeit einer teilchenbasierten Simulation (Molekulardynamik) mit der numerischen Effizienz einer Kontinuumsmethode (numerische Strömungsmechanik). In den dieser Arbeit zugrundeliegenden Problemstellungen geht es dabei um die Beschreibung der Strömung von weicher Materie im Rahmen der Fluidodynamik mithilfe einer Multiskalenmethode.

Die teilchenbasierten Molekulardynamik-Simulationen auf der Mikroskala stellen bei einem solchen Multiskalen-Verfahren typischerweise einen Engpass dar, da in jedem Schritt der Kontinuumsmethode eine oder mehrere zeitaufwendige Mikroskalen-Simulationen gestartet werden müssen. Das Ziel dieser Arbeit ist es, die entsprechenden Parameter und zugrundeliegenden möglichen Algorithmen auf der Mikroskala zu optimieren. Auf diese Weise wird das gesamte Verfahren verbessert und beschleunigt.

Dabei wird sowohl auf eine geeignete Wahl des Thermostats eingegangen, mit dem die Temperatur des Systems kontrolliert wird, wie auch auf geeignete Parameter der jeweiligen Thermostate. Die dafür untersuchten Größen sind der Stresstensor und die davon abgeleiteten Observablen, der Druck und die dynamische Viskosität. Die Wahl der Parameter des Lowe-Andersen-Thermostats spielt dabei ebenso eine Rolle. Durch Kontrolle dieser Variablen kann eine Flüssigkeit mit höherer Viskosität simuliert werden, als die simulierten Teilchen eigentlich haben. Auf diese Weise kann eine ressourcenintensive hydrodynamische Vielteilchendynamik umgangen werden.

Das isokinetische Thermostat wird ebenfalls im Detail betrachtet und dient der Temperierung eines Systems von Newtonschen Fluiden. Die allgemeinen Simulationsparameter wie Zeitschritt und Simulationsboxgröße werden unter Zuhilfenahme dieses Thermostats untersucht. Dabei ist es unter dem Aspekt der Hybridsimulation bevorzugt, mehrere kleinere Systeme zu untersuchen, statt wenige große Systeme mit komplexer Parallelisierung zu realisieren.

Sowohl einfache Weeks-Chandler-Andersen-Teilchen, als auch kleine Polymerketten, die ein anderes viskoses Verhalten aufweisen als die Monomersysteme und unter Verwendung eines Kremer-Grest-Modells simuliert werden, werden untersucht. Hier werden Parameter wie Dichte und Kettenlänge und ihre Auswirkungen auf die Viskosität bestimmt.

Neben den Ergebnissen der Parameteruntersuchung sollen auch die Verfahren zur Datenverfeinerung und die Fitmethoden vorgestellt sowie auf die Simulationsdaten angewendet werden, um die notwendige Gesamtzahl der Simulationen so gering wie möglich zu halten. In diesem Zusammenhang werden auch erste Ergebnisse des hybriden Verfahrens präsentiert.

PUBLIKATIONEN

Einige Ergebnisse dieser Arbeit sind bereits in den folgenden Publikationen erschienen:

- [1] N. Emamy, M. Lukáčová-Medvid'ová, S. Stalter, P. Virnau and L. Yelash, *Reduced-order hybrid multiscale method combining the Molecular Dynamics and the Discontinuous-Galerkin method*, VII International Conference on Computational Methods for Coupled Problems in Science and Engineering Coupled Problems (Eds. Papadrakakis, Onate and Schrefle), p. 1–15 (2017)
- [2] S. Stalter, L. Yelash, N. Emamy, A. Statt, M. Hanke, M. Lukáčová-Medvid'ová and P. Virnau, *Molecular dynamics simulations in hybrid particle-continuum schemes: Pitfalls and caveats*, Computer Physics Communications, **224** (2018)

INHALTSVERZEICHNIS

i	EINLEITUNG	1
1	EINLEITUNG	3
ii	GRUNDLAGEN UND METHODEN	7
2	MOLEKULARDYNAMIK	9
2.1	Velocity-Verlet-Algorithmus	11
2.2	Lees-Edwards-Randbedingungen	13
2.2.1	SLLOD-Algorithmus	15
2.3	Virialdruck und Stresstensor in Simulationen	17
3	THERMOSTATE	21
3.1	Das Lowe-Andersen-Thermostat	22
3.2	Das isokinetische Thermostat	24
3.3	Das Langevin-Thermostat	24
4	MODELLSYSTEME	27
4.1	Harte Kugeln	27
4.1.1	Reskalierung	30
4.1.2	Abbilden des Modells	31
4.2	Polymerketten	32
5	CFD-SIMULATIONEN	35
5.1	Herleitung	35
5.1.1	Die Kontinuitätsgleichung	35
5.1.2	Die Impulsgleichungen	36
5.1.3	Die Navier-Stokes-Gleichungen	37
5.1.4	Verwendetes Integrationsschema	39
5.2	Diskontinuierliche Galerkin-Methode	41
5.2.1	Definitionen	42
5.2.2	Schwache Formulierung der Gleichungen	43
iii	UNTERSUCHUNGEN IM RAHMEN VON HYBRID-SCHEMATA	47
6	PARAMETEROPTIMIERUNG VON WCA-TEILCHEN	49
6.1	Vergleich verschiedener Thermostate	49
6.2	Parameterwahl für das isokinetische Thermostat	57
6.3	Fehleruntersuchung der Viskosität	63
7	PARAMETEROPTIMIERUNG DER SIMULATIONEN VON POLYMERKETTEN	67
7.1	Einschwingvorgang	67
7.2	Scherprofile	68
7.3	Temperaturverlauf der Polymerketten	70
7.4	Stresstensor unter Scherung	70
iv	KOPPLUNG	77
8	GRUNDLAGEN DER KOPPLUNG	79
8.1	Drehung der Simulationsboxen	80
8.2	Senkung der Parameterordnung	82

8.2.1	Datenverfeinerung	83
8.2.2	Erweiterung der Näherung auf ρ	84
8.2.3	Optimierungsalgorithmus	84
8.3	Eigenwert-Dekomposition	86
8.4	Rauschreduzierung	89
8.5	Chebyshev-Approximation mit Tikhonov-Regularisierung	91
8.6	Rücktransformation	95
9	UNTERSUCHTE SYSTEME	99
9.1	Couette-Strömungen mit Monomeren	99
9.2	Poiseuille-Fluss mit Monomeren	101
V	ZUSAMMENFASSUNG UND FAZIT	103
10	ZUSAMMENFASSUNG UND FAZIT	105
vi	APPENDIX	109
A	MODELLSYSTEM	111
A.1	Vergleich mit vorherigen Ergebnissen	111
A.2	Polymerketten	112
A.2.1	Dichte $\rho = 0.4$	112
A.2.2	Variation des Parameters k im FENE-Potential	115
B	NAVIER-STOKES DISKRETISIERUNG MIT GESTAFFELTEM GITTER	117
C	L-KURVEN DER TIKHONOV-REGULARISIERUNG	119

ABBILDUNGSVERZEICHNIS

Abb. 1.1	Skalenverhalten der Simulationsphysik.	3
Abb. 2.1	Momentaufnahme einer MD-Simulation.	9
Abb. 2.2	Phasenraumellipse.	12
Abb. 2.3	Definition der Scherrate $\dot{\gamma}$	13
Abb. 2.4	Lees-Edwards-Randbedingungen.	14
Abb. 2.5	Momentaufnahme der gescherten Simulationsboxen.	16
Abb. 3.1	Temperaturverlauf mit und ohne Thermostat.	21
Abb. 3.2	Bewegung d. Teilchen nach VV-Schritt und nach Thermostatschritt. 23	
Abb. 4.1	LJ- und WCA-Potential.	28
Abb. 4.2	Phasendiagramm der WCA-Kugeln.	29
Abb. 4.3	Kremer-Grest-Modell.	33
Abb. 5.1	Veranschaulichung der Definitionen.	42
Abb. 6.1	Scherprofile für die beiden Thermostate.	50
Abb. 6.2	Einschwingvorgang der Größe σ_{13}	51
Abb. 6.3	Vergleich Temperatur und Druck zwischen LAT und ISO.	52
Abb. 6.4	Vergleich der Viskosität zwischen LAT und ISO.	54
Abb. 6.5	Viskosität in Abhängigkeit der Parameter des LAT.	55
Abb. 6.6	Vergleich zwischen Langevin und ISO.	56
Abb. 6.7	Autokorrelationszeiten.	58
Abb. 6.8	Vergleich von p und η für versch. Δt	59
Abb. 6.9	Vergleich von p und η als Funktion von Δt	60
Abb. 6.10	Vergleich von p und η als Funktion der Boxlänge.	62
Abb. 6.11	Vgl. von p und η als Funktion der Boxlänge in (x, z) -Richtung. .	64
Abb. 6.12	Zeitentwicklung von σ_{13} und η	66
Abb. 7.1	Einschwingvorgang für Polymerketten.	67
Abb. 7.2	Autokorrelationsfunktion der Polymerketten.	68
Abb. 7.3	Scherprofile für Polymerketten.	69
Abb. 7.4	Temperaturverlauf der Polymerketten.	70
Abb. 7.5	p und σ_{13} für verschiedene Kettenlängen.	71
Abb. 7.6	η der Polymerkettensysteme.	73
Abb. 7.7	η in Abhängigkeit der inversen Kettenlänge.	74
Abb. 7.8	Momentaufnahme einer Polymerkette.	75
Abb. 8.1	Drehung der MD-Boxen.	79
Abb. 8.2	Datentransfer im Hybridsimulationsschema.	81
Abb. 8.3	σ_{ii}^{MD} gegen $\dot{\gamma}$	88
Abb. 8.4	$\tilde{\sigma}^{MD}$ gegen λ	90
Abb. 8.5	Singulärwerte und relativer Fehler.	92
Abb. 8.6	Tikhonov-Näherung mit Chebyshev-Polynomen.	93
Abb. 8.7	L-Kurven.	94
Abb. 8.8	Chebyshev-Näherung nach Rücktransformation.	96
Abb. 9.1	Darstellung der Couette-Strömung.	99
Abb. 9.2	Geschwindigkeitsprofil des Couette-Flusses.	100

Abb. 9.3	Darstellung des Poiseuille-Flusses.	101
Abb. 9.4	Geschwindigkeitsprofil des Poiseuille-Flusses.	102
Abb. A.1	Vergleich von η mit unabhängigen Quellen.	111
Abb. A.2	Temperaturverlauf der Polymerketten bei $\rho = 0.4$	112
Abb. A.3	p der Polymerkettensysteme für $\rho = 0.4$	113
Abb. A.4	σ_{13} und η der Polymerkettensysteme für $\rho = 0.4$	114
Abb. A.5	p der Polymerkettensysteme für versch. Federkonstanten.	115
Abb. A.6	σ_{13} der Polymerkettensysteme für versch. Federkonstanten.	116
Abb. A.7	η der Polymerkettensysteme für versch. Federkonstanten.	116
Abb. B.1	Das gestaffelte MAC-Gitter.	117
Abb. C.1	L-Kurven für $P_{\max} = 4$	120
Abb. C.2	L-Kurven für $P_{\max} = 8$	121

Teil I

EINLEITUNG

EINLEITUNG

Simulationsphysik bildet eine wichtige Verbindung zwischen experimenteller und theoretischer Physik. Bevor Theorien experimentell getestet werden, können durch Simulationen die dafür notwendigen Versuchsp Parameter des Experiments abgesteckt werden oder Theorien im Vorfeld überprüft werden.

Dabei ist es notwendig, dass sowohl in der Theorie als auch in der Simulation die einzelnen Bestandteile und angenommenen Näherungen verstanden werden. Eine aus Knete geformte Nachbildung einer Holzfigur hat andere Materialeigenschaften als das Original aus Holz. Dennoch wirft sie den gleichen Schatten. Die Modellierungsgrundlagen und verwendeten Theorien müssen also auch auf den zu untersuchenden Parameter abgestimmt werden.

Simulationen von komplexen Flüssigkeiten stellen in der Physik eine besondere Herausforderung dar. Für eine exakte Beschreibung eines realistischen Systems ist eine quantenbasierte Simulation denkbar. Mit dieser können aber, wie in Abbildung 1.1 dargestellt, nur kleine Zeit- und Längenskalen nachgestellt werden. Vergrößert man das System, indem ein „Teilchen“ der Simulation mehreren Atomen entspricht und quantenmechanische (oder später atomistische, etc.) Effekte vernachlässigt werden, können größere Zeit- und Längenskalen abgebildet werden. Zudem ist je nach Problemstellung lediglich die makroskopische Ordnung relevant oder interessant.

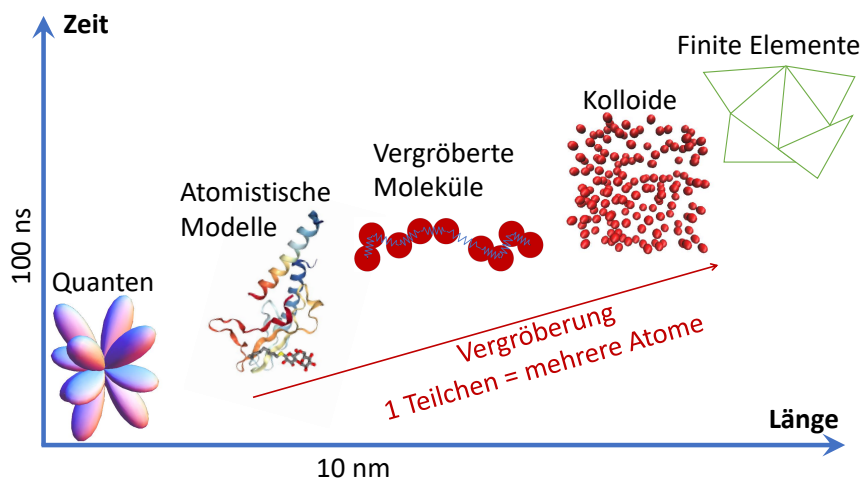


Abb. 1.1: Skalenverhalten in der Simulationsphysik. Das hier gezeigte Schema zeigt die Skalen, die mit verschiedenen Simulationsarten dargestellt werden können. Je nach Anzahl der Freiheitsgrade und zugrundeliegenden Simulationsart können verschiedene Zeit- und Längenskalen simuliert werden. Durch Vergrößerung des Modells werden intrinsische Eigenschaften vernachlässigt. Die auf diese Weise verlorenen Informationen sind je nach Problemstellung nicht relevant. Die einzelnen Skalen wurden dabei mithilfe der Programme Mathematica (Quanten) [3] und vmd (Atomistisches Modell, Kolloide) [4, 5] visualisiert, die Darstellung des atomistischen Modells entspricht dabei 3IOL [6]. Vergrößerte Moleküle und Finite Elemente sind mithilfe von PowerPoint entstanden.

Für Simulationen von Kolloiden, die von ihrer Größe und der typischen Zeitskala ihrer Wechselwirkungen auf der Mesoskala liegen, werden mithilfe von Molekulardynamik (MD) realisiert [7, 8], die in Kapitel 2 näher erläutert wird. Mit dieser Art der Simulation können jedoch noch immer nur relativ kleine Zeit- und Längenskalen in akzeptabler Simulationszeit realisiert werden. Um größere Zeit- und Längenskalen abdecken zu können, werden kontinuierliche Modelle verwendet, welche allerdings keinen Zugriff auf verschiedene interne Systemgrößen liefern. Ein Beispiel hierfür ist die in dieser Arbeit thematisierte numerischen Strömungsdynamik (engl. *Computational Fluid Dynamics*, CFD) [9], die in Kapitel 5 näher besprochen wird.

In der Literatur wird eine Kombination beider Skalen in Form einer Hybridmethode vorgeschlagen, wodurch die Vorteile beider Methoden genutzt und die Schwächen jeweils ausgeglichen werden. Die Heterogene Multiskalen Methode (HMM) wurde in der Vergangenheit bereits erfolgreich genutzt, um skalenseparierte Systeme zu beschreiben. Diese Systeme zeichnen sich dadurch aus, dass die Prozesse auf der Mikroebene die Prozesse auf der Makroebene nicht oder kaum beeinträchtigen, siehe auch [10, 11, 12, 13, 14, 15, 16].

In dieser Arbeit geht es um die Optimierung der Parameter der Molekulardynamikmethoden im Rahmen einer Hybridsimulation. Diese Hybridsimulation koppelt zwei Simulationsarten, die Zugang zu zwei unterschiedlichen Skalenbereichen erlauben. Während die Makroebene den gesamten zu simulierenden Bereich betrachtet und mithilfe der CFD-Simulation behandelt wird, erlaubt die MD-Simulation auf der Mikroebene eine detailliertere Betrachtung der physikalischen Vorgänge.

Da in jedem Schritt der CFD-Simulation der genaue Wert des Stresstensors σ für die dort herrschenden Randbedingungen nötig ist, müsste für jeden Quadraturpunkt dieser Simulation eine Mikrosimulation gestartet werden. Um diesen rechnerischen Aufwand zu minimieren, sind verschiedene Methoden in [1] und [2] vorgestellt worden, die die Datennahme optimieren. Diese werden in Kapitel 8 behandelt.

Die Kopplung selbst wird dabei zunächst mit einer „Offline“-Phase auf ihre Funktionalität geprüft. Dabei werden die beiden Skalen voneinander getrennt betrachtet; der Informationsaustausch erfolgt manuell durch Wertetabellen. Optimierungsmethoden dienen dabei dazu, die von der Mikroskala erhaltenen Daten derart vorzubereiten, dass möglichst wenige Mikrosimulationen gestartet werden müssen. Nach dieser intensiven Trainingsphase kann eine hybride Simulationsmethode entwickelt werden, die in Form einer Schnittstelle die beiden Simulationsprogramme anspricht und die Kommunikation der beiden erlaubt.

Der Schwerpunkt der Arbeit liegt auf den Untersuchungen der Molekulardynamik (MD). MD-Simulationen betrachten ein teilchenbasiertes System und lösen die dem System zugrundeliegenden Bewegungsgleichungen. Dabei wird jedes Molekül der zu simulierenden Flüssigkeit als stark vergrößerte Kugel aufgefasst. Die einzelnen Teilchen interagieren über ein Wechselwirkungspotential, welches eine Kraft auf die jeweils anderen Teilchen induziert. Auch externe Kräfte, wie elektromagnetische Felder, können eine Rolle spielen, werden in den hier verwendeten Simulationen aber nicht betrachtet.

Je nach Wahl des Potentials und der jeweiligen Potentialparameter können unterschiedliche physikalische Systeme von Idealen Gasen über Kolloide simuliert werden. Auch lange Molekülketten, die in grober Näherung nicht mehr als Kugeln aufgefasst werden können, lassen sich realisieren, indem man die einzelnen Molekülkettenteile als Kugeln realisiert und diese mit Federn miteinander verbindet (siehe auch Abschnitt 4.2) [17]. Dadurch können beliebig lange Molekülketten realisiert werden.

Die Arbeit ist wie folgt aufgeteilt. Im nächsten Teil wird auf die grundlegenden Algorithmen und verwendeten Methoden hingewiesen. Die Grundlagen der Molekulardynamik werden in Kapitel 2 behandelt. Neben der generellen Funktionsweise dieser Simulationemethode werden auf die hier verwendeten Algorithmen und Randbedingungen eingegangen, um ein lineares Scherprofil in den Simulationen zu erhalten. Auch die Herleitung des Stresstensors, der als Information für die Makro-Ebene benötigt wird, und die Berechnung des Tensors in der Simulation werden in diesem Kapitel behandelt.

Anschließend folgt in Kapitel 3 die theoretische Grundlage der in dieser Arbeit verwendeten Thermostate, welche die durch Scherung dem System zugeführte Wärme ableiten sollen, sodass die Temperatur des Systems konstant bleibt. Neben der Funktionsweise eines Thermostats werden das Lowe-Andersen-Thermostat und das isokinetische Thermostat erklärt. Auch auf das Langevin-Thermostat wird eingegangen, mit dem überprüft wird, ob die Aussage, dass das isokinetische Thermostat für den stabilen Zustand korrekte Ergebnisse liefert, zutrifft.

Die verwendeten Modellsysteme werden in Kapitel 4 thematisiert. Dabei wird das Hart-Kugel-Modell beschrieben und neben einer Reskalierung des Kugeldurchmessers auf die Abbildung des Modells auf reale Systeme eingegangen. Auch das den Polymerketten zugrundeliegende Modell wird hier erklärt.

Auf die CFD-Simulation wird ebenfalls im zweiten Teil der Arbeit in Kapitel 5 eingegangen. Dabei werden die Bewegungsgleichungen und das verwendete Integrationsschema hergeleitet. Die diskontinuierliche Galerkin-Methode erfordert, dass das Simulationsgebiet auf eine bestimmte Weise in Ort und Zeit diskretisiert wird. Die Definitionen und diese Diskretisierung finden sich ebenfalls in diesem Kapitel.

Im dritten Teil der Arbeit werden die Parameter der MD-Simulation des Harte-Kugeln-Systems zur Optimierung unter dem Augenmerk der Hybrid-Simulation in Kapitel 6 untersucht. Dabei werden Zeitschritt und Boxgröße für das isokinetische Thermostat, sowie Thermostat-Parameter für das Lowe-Andersen-Thermostat variiert und optimiert. Außerdem werden die beiden Thermostate miteinander verglichen. Ein Vergleich findet auch zwischen dem isokinetischen Thermostat mit dem Langevin-Thermostat statt.

In Kapitel 7 werden die Ergebnisse der Simulationen von Polymerketten betrachtet und mit den Ergebnissen aus dem vorherigen Kapitel 6 verglichen. Dabei ist der Effekt des sogenannten *Shear-Thinnings* besonders interessant.

Die Kopplung selbst wird in Kapitel 8 im vierten Teil der Arbeit besprochen. Hier werden die Methoden präsentiert, die die Simulationsergebnisse vereinfachen und die Kommunikation der beiden Programme erleichtern sollen.

In Kapitel 9 wird schließlich anhand eines Problems mit bekannter analytischer Lösung eine Demonstration der Hybrid-Simulation zeigen, dass diese die erwarteten Ergebnisse liefert. Dieses Anwendungskapitel zeigt die bisherigen Ergebnisse der Hybridsimulationen.

Abschließend werden die in den vorherigen Kapiteln erhaltenen Ergebnisse aus der Parameteroptimierung sowie den ersten Kopplungen der Systeme im fünften Teil in Kapitel 10 zusammengefasst und noch einmal erläutert. Ein Fazit und ein kleiner Ausblick beenden die Arbeit.

An dieser Stelle möchte ich mich für die Rechenzeit, die mir vom HPC Cluster Mogon der Johannes Gutenberg-Universität Mainz zur Verfügung gestellt wurde, bedanken. Die zeitaufwendigen Simulationen in dieser Arbeit wurden auf dem Cluster durchgeführt.

Teil II

GRUNDLAGEN UND METHODEN

Die Simulation eines Systems stellt stets eine Näherung der physikalischen Realität dar. Während zufallsbasierte Simulationsarten wie Monte-Carlo-Simulationen [18] den Phasenraum großflächig abdecken, aber keine dynamischen Untersuchungen eines Systems erlauben, werden bei Molekulardynamik-Simulationen (kurz: MD) die Newtonschen Bewegungsgleichungen der einzelnen Teilchen durch Integration über einen endlichen Zeitschritt Δt gelöst, wodurch nur ein kleiner Teil des Phasenraums abgedeckt werden kann. Dafür können in solchen Simulationen dynamische Eigenschaften und Einschwingvorgänge analysiert werden. Für die Zeitintegration der MD-Simulation können verschiedene Methoden verwendet werden. Auf eine kurze Herleitung und die hier verwendeten Algorithmen wird im Nachfolgenden eingegangen. Eine Momentaufnahme einer solchen teilchenbasierten Simulation ist in Abbildung 2.1 zu sehen.

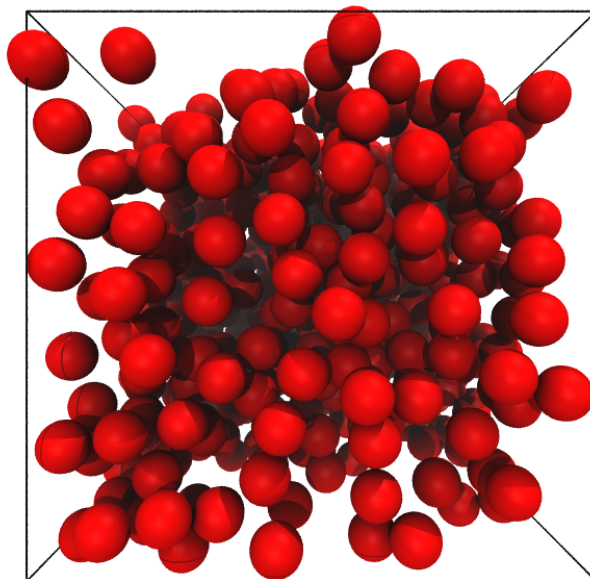


Abb. 2.1: Momentaufnahme einer Simulation, der Molekulardynamik zugrunde liegt. Der Simulationsbereich ist dabei in Form einer Box eingezeichnet. Bei den roten Kugeln handelt es sich um Kolloide, die als harte Kugeln realisiert sind, wie in Kapitel 4.1 vorgestellt.

Die zum Verständnis nötigen Grundlagen werden in den folgenden Kapiteln kurz erläutert. Dabei wird grob die Herleitung aus [7] nachvollzogen. Wichtig sind dabei vor allem die verwendeten Integratoren und Randbedingungen. Je nach Wahl des sogenannten Thermostats, welches die Temperatur konstant hält, unterscheiden sich die zu verwendenden

Integratoren. Die in dieser Arbeit angewandten Thermostate werden in Kapitel 3 vorgestellt.

Ein physikalisches Teilchen bewegt sich entsprechend den Newtonschen Bewegungsgleichungen. Diese berücksichtigen momentane Geschwindigkeiten und auf das Teilchen resultierende Kräfte. In einer MD-Simulation wird zunächst die Kraft auf ein Teilchen i ermittelt. Dazu wird der Gradient des Paarpotentials zwischen den einzelnen Teilchensorten bestimmt und die Kraft mithilfe des Abstands von Teilchen i zu allen übrigen Teilchen j berechnet. Anschließend werden entsprechend der Bewegungsgleichungen sowohl die neue Position als auch die neue Geschwindigkeit des Teilchens i für einen Zeitschritt Δt bestimmt und aktualisiert.

In dieser Arbeit werden Strömungen, also Systeme unter Scherung betrachtet. Durch diese Scherung wird dem System konstant Energie zugeführt, wodurch die Temperatur steigt. Um die Temperatur konstant zu halten, wird ein Thermostat eingeführt, siehe auch Kapitel 3. Nachdem die Positionen und Geschwindigkeiten der einzelnen Teilchen aktualisiert sind, wird ein aus dem Thermostat resultierender Algorithmus zur Kühlung des Systems angewandt.

Das System wird permanent durch z. B. eine äußere Kraft, die eine Scherung induziert, aus dem Gleichgewicht gebracht, wodurch das System nicht sein Äquilibrium erreichen kann. Dennoch ist man hier an einem stabilen Zustand interessiert. Da das System einer konstanten Scherung unterliegt, stellt sich nach einer gewissen Relaxationszeit ein lineares, konstantes Scherprofil ein. Anschließend ändert sich an den äußeren Randbedingungen und der Dynamik der Systeme nichts, man kann also von einem stabilen Zustand, einem *Fließgleichgewicht* sprechen, in welchem der Stresstensor gemessen wird, an dem man für die Makroebene der Simulation interessiert ist. Dieser stabile Zustand wird im Englischen als *Steady State* bezeichnet.

In den der Arbeit zugrunde liegenden Simulationen wird eine zufällige Startkonfiguration erzeugt. Um ein relaxiertes System zu erhalten, werden vor einer Messung einer Observablen A bereits eine bestimmte Anzahl an Schritten durchgeführt. Anschließend wird die Observable A im Abstand von mehreren Zeitschritten bestimmt und herausgeschrieben.

Aus diesen von der Simulationszeit abhängigen Größen $A(t)$ wird durch Mittelwertbildung der Wert der Observablen A berechnet. Diese Zeitmittelwerte entsprechen dabei für kurze Simulationen nicht notwendigerweise dem gesuchten Mittelwert im Gleichgewicht, bei denen man tatsächlich von Simulationen im mikrokanonischen Ensemble sprechen könnte. Sind die Simulationen jedoch lange genug gelaufen, geht man davon aus, dass der Zeitmittelwert $\overline{A(t)}$ dem Ensemblemittelwert $\langle A \rangle$ entspricht (Ergodenhypothese [7, 19]):

$$\overline{A(t)} = \lim_{t \rightarrow \infty} \frac{1}{t} \int_0^t A(t') dt' \stackrel{!}{=} \langle A \rangle = \sum_i \pi_i A_i, \quad (2.1)$$

wobei π_i die Wahrscheinlich ist, wonach der Zustand i eintritt. Dieser Wert ist durch die Wahrscheinlichkeitsverteilung des Systems gegeben. Dabei ist ein dynamisches System ergodisch (oder quasi-ergodisch), wenn die Trajektorie jedem Punkt im Phasenraum in endlicher Zeit beliebig nahekommt. Dies wird auch in [20] thematisiert.

2.1 VELOCITY-VERLET-ALGORITHMUS

In diesem Unterabschnitt wird das verwendete numerische Verfahren zur Integration des Zeitschritts vorgestellt. Für die implementierungsbasierte Beschreibung kann auf Frenkel und Smit [7] zurückgegriffen werden.

Ein System bestehe aus N Teilchen, die mit einem Paarpotential wechselwirken. Aus dem Paarpotential und dem Abstand der Teilchen lässt sich die Kraft, die auf Teilchen i wirkt, über

$$\mathbf{F}_i = \sum_{j=0, j \neq i}^N -\nabla U_{PP}(\mathbf{r}_{ij}) , \quad (2.2)$$

bestimmen, wobei U_{PP} das Paarpotential, \mathbf{r}_{ij} der Abstand von Teilchen i zu Teilchen j ist. Die Position eines Teilchens nach einer Zeit Δt lässt sich durch eine Taylor-Reihe entwickeln:

$$\mathbf{r}(t + \Delta t) = \mathbf{r}(t) + \mathbf{v}(t)\Delta t + \frac{\mathbf{F}(t)}{2m} (\Delta t)^2 + \frac{(\Delta t)^3}{3!} \ddot{\mathbf{r}}(t) + \mathcal{O}((\Delta t)^4) . \quad (2.3)$$

Da der Ort $\mathbf{r}(t)$ und die Geschwindigkeit $\mathbf{v}(t)$ bereits aus dem vorherigen Schritt bekannt sind und die Kräfte aus den Potentialen berechnet werden können, sind die neuen Positionen der Teilchen $\mathbf{r}(t + \Delta t)$ bereits bis einschließlich zweiter Ordnung bekannt. Die Geschwindigkeit $\mathbf{v}(t + \Delta t)$, welche für den nächsten Integrationsschritt benötigt werden, können analog berechnet werden, allerdings ist dort die Genauigkeit nur bis zur ersten Ordnung gegeben, wenn keine weiteren Terme zur Bestimmung hinzugezogen werden sollen.

Da dieser Algorithmus mit der numerischen Integration nach Euler übereinstimmt, wird er auch als Euleralgorithmus bezeichnet. In den meisten Fällen führt er jedoch durch ungenügende Genauigkeit zu einer Überschätzung des Ergebnisses und damit unter anderem zu einem Energiedrift [7].

Eine weitere und verbesserte Methode ist der Verlet-Algorithmus, bei dem auch $\mathbf{r}(t - \Delta t)$ in eine Taylor-Reihe entwickelt wird und anschließend zu Gleichung (2.3) addiert wird. Für die neuen Positionen $\mathbf{r}(t + \Delta t)$ ergibt sich damit:

$$\mathbf{r}(t + \Delta t) = 2\mathbf{r}(t) - \mathbf{r}(t - \Delta t) + \frac{\mathbf{F}(t)}{m} (\Delta t)^2 + \mathcal{O}((\Delta t)^4) . \quad (2.4)$$

Die neuen Positionen sind nun genauer bestimmt als mit dem vorherigen Algorithmus und nicht mehr von der Geschwindigkeit abhängig. Die Geschwindigkeit kann aber weiterhin bis einschließlich erster Ordnung aus den Positionen berechnet werden. Wendet man den Verlet-Algorithmus statt auf die Positionen \mathbf{r} auf die Geschwindigkeiten \mathbf{v} an, erhält man den sogenannten Velocity-Verlet algorithm (deutsch: *Geschwindigkeits-Verlet-Algorithmus*):

$$\mathbf{r}(t + \Delta t) = \mathbf{r}(t) + \mathbf{v}(t)\Delta t + \frac{\mathbf{F}(t)}{2m} (\Delta t)^2 + \mathcal{O}((\Delta t)^3) , \quad (2.5)$$

$$\mathbf{v}(t + \Delta t) = \mathbf{v}(t) + \frac{\mathbf{F}(t) + \mathbf{F}(t + \Delta t)}{2m} \Delta t + \mathcal{O}((\Delta t)^3) . \quad (2.6)$$

Beispielhaft wird in Abbildung 2.2 gezeigt, wie eine Kreisbahn angenähert werden könnte. Um zu zeigen, dass die Bahnen über- bzw. unterschätzt werden können, erhalten die verwendeten Geraden eine höhere Steigung, als wenn sie durch die Extremwerte gelegt werden sollten. Die Geraden genügen der Geradengleichung $y = mx + b$ mit Steigung

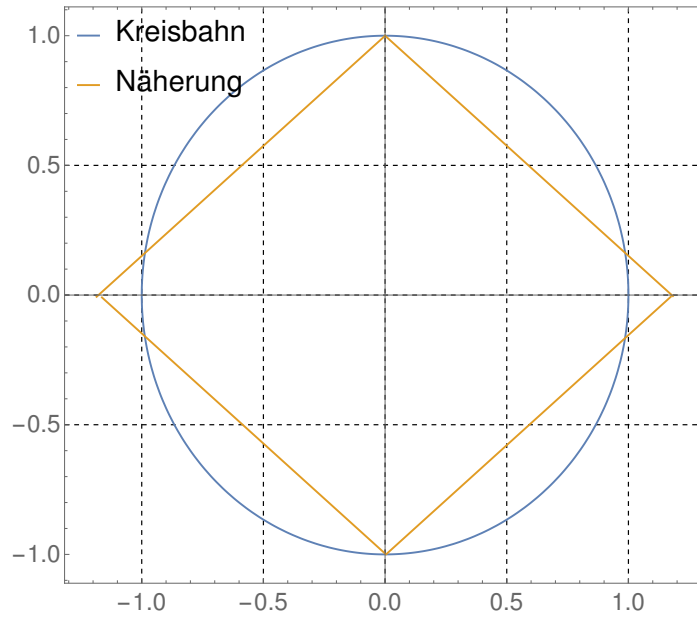


Abb. 2.2: Phasenraumellipse eines Teilchens auf einer Kreisbahn und eines Teilchens mit linearer Näherung des Ortes. Der Sinus wurde dabei mit einer Gerade mit Steigung $m = 0.75$ angenähert. Diese hat bei den Extremstellen einen höheren Betrag in der y -Komponente, wodurch die ideale Kreisbahn nicht erreicht wird. Wie auch in den Simulationen kann die Phasenraumellipse des tatsächlichen Teilchens nicht exakt beschrieben werden, die Phasenraumellipse wird an einigen Stellen überschätzt, an den meisten unterschätzt.

$m = 0.75$, wobei der y -Achsenabschnitt b derart gewählt wird, dass die Nulldurchgänge mit dem Sinus übereinstimmen. Auf diese Weise wird die Phasenraumellipse über- bzw. unterschätzt, wie in der Abbildung dargestellt. In den Simulationen sind diese Abschätzungen sehr viel feiner und schwingen nur minimal um die Phasenraumellipse. Je besser der Algorithmus, desto genauer wird der Phasenraum abgetastet.

Insgesamt wird eine MD-Simulation nach direktem Einsetzen des Geschwindigkeits-halbschritts aus wiederholtem Anwenden nachfolgender Schritte bestehen:

- Schritt 1: Berechne die neue Position \mathbf{r} der Teilchen:

$$\mathbf{r}(t + \Delta t) = \mathbf{r}(t) + \mathbf{v}(t)\Delta t + \frac{\mathbf{F}(t)}{2m}\Delta t^2 .$$
- Schritt 2: Bestimme die Kraft \mathbf{F} für den nächsten Zeitschritt mit den neuen Positionen $\mathbf{r}(t + \Delta t)$:

$$\mathbf{F}(t + \Delta t) = -\nabla U(|\mathbf{r}(t + \Delta t)|) .$$
- Schritt 3: Berechne die Geschwindigkeit $\mathbf{v}(t + \Delta t)$ aus der Kraftberechnung aus dem vorherigen Schritt:

$$\mathbf{v}(t + \Delta t) = \mathbf{v}(t) + \frac{1}{2} \left(\frac{\mathbf{F}(t) + \mathbf{F}(t + \Delta t)}{m} \right) \Delta t .$$
- (Schritt 4: Anwenden der Thermostat-Algorithmen zur Kühlung des Systems.)

Schritt 4 tritt dabei nur beim isokinetischen und beim Lowe-Andersen-Thermostat auf. Beim ebenfalls in dieser Arbeit betrachteten Langevin-Thermostat wird der Algorithmus direkt an die Bewegungsgleichungen gekoppelt.

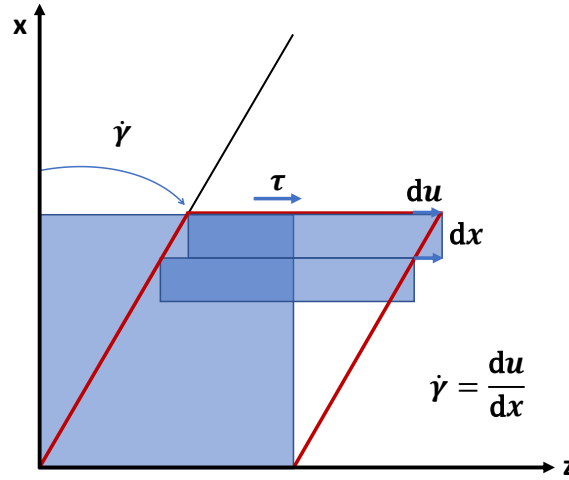


Abb. 2.3: Definition der Scherrate $\dot{\gamma}$. Damit wird die Änderung des Winkels bezeichnet, mit dem eine Form deformiert wird. Das Fluidvolumen wird in diesem Fallbeispiel analog zu der in dieser Arbeit gebrauchten Grundlagen in z -Richtung geschert. Exemplarisch wurden hier zwei Schichten eingezeichnet, die eine Dicke dx haben. Die beiden Schichten bewegen sich mit einem Geschwindigkeitsunterschied von du zueinander. Die Scherrate ist dementsprechend $\dot{\gamma} = \frac{du}{dx}$.

2.2 LEES-EDWARDS-RANDBEDINGUNGEN

Aufgrund begrenzter Rechenzeitressourcen ist es immer nur möglich, eine verhältnismäßig kleine Anzahl von Teilchen zu simulieren. Um Abweichungen aufgrund der kleinen Systemgröße zu vermeiden, verwendet man periodische Randbedingungen [8]. Bei diesen wird die Simulationsbox virtuell weitergeführt, sodass das Simulationsgebiet am oberen Rand mit dem unteren Rand fortgeführt wird. Die Simulationsbox wird also in jede Raumrichtung kopiert. Das heißt, dass ein Teilchen am Rand des Gebiets entsprechend der sogenannten *Minimum Image Convention* [8] mit Teilchen am entgegengesetzten Rand wechselwirkt. Diese Konvention besagt, dass ein Teilchen immer mit der Kopie desjenigen Teilchens wechselwirkt, das den kürzesten Abstand besitzt. Verlässt ein Teilchen das Simulationsgebiet, wird es auf der entgegengesetzten Seite wieder einsortiert. Hierdurch täuscht man dem System vor, dass es in ein unendlich großes System eingebettet ist.

Dies funktioniert insbesondere, wenn typische Korrelationslängen r deutlich kleiner sind als die lineare Größe der Simulationsbox, andernfalls wechselwirkt ein Partikel über den Rand hinaus mit sich selbst. Dies kann zu unerwünschten Nebeneffekten, sogenannten *Finite-Size-Effekten*, führen. In den hier vorliegenden Simulationen werden Potentiale mit sehr kurzer Reichweite verwendet, welche nach einem gewissen Abstand abgeschnitten werden (wie sie in Kapitel 4 vorgestellt werden), sodass die Boxgröße relativ klein gewählt werden kann, siehe hierzu auch Kapitel 6.2.

In den für das Hybrid-Schema nötigen Simulationen werden gescherte Systeme betrachtet. Die Stärke dieser Scherung wird durch die in dieser Arbeit als Scherrate bezeichnete Schergeschwindigkeit $\dot{\gamma}$ angegeben. Die Scherrate ist dabei das Verhältnis von Geschwindigkeitsunterschied zweier Flüssigkeitsschichten du zur Schichtdicke dx :

$$\dot{\gamma} = \frac{du}{dx} . \quad (2.7)$$

Dies ist in Abbildung 2.3 exemplarisch gezeigt.

Die Scherrate stellt außerdem die inverse Steigung des (linearen) Geschwindigkeitsprofils bei angelegter Scherung dar. Bei mehrdimensionalen Spannungen ist die Gesamtscherrate entsprechend die Summe der einzelnen Scherraten [21]. Der Verzerrungsgeschwindigkeitstensor \mathbf{E} enthält die Information der Scherrate. In dieser Arbeit wird lediglich die in der Literatur als einfache Scherung (engl.: *simple shear*) bezeichnete Art der Scherung untersucht. Dabei handelt es sich um eine nur durch die Änderung des Winkels einer Achse induzierte Spannung, wie in Abbildung 2.3 angedeutet, die in einem linearen Geschwindigkeitsprofil resultiert.

Durch die angelegte Scherung entsteht ein Geschwindigkeitsprofil in der simulierten Flüssigkeit. Um diese in der Simulation zu realisieren, werden die periodischen Randbedingungen abgeändert. Auch bei den hier verwendeten Lees-Edwards-Randbedingungen [22] wird die Simulationsbox virtuell in alle Raumrichtungen kopiert, jedoch werden die Boxen, die in x -Richtung hinzugefügt werden, zusätzlich in z -Richtung verschoben, wie in Abbildung 2.4 dargestellt.

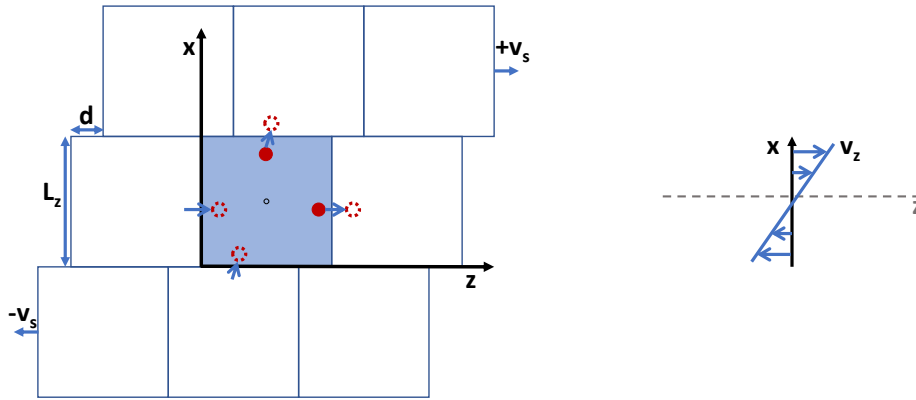


Abb. 2.4: Lees-Edwards-Randbedingungen. In z - und in der nicht dargestellten y -Richtung werden periodische Randbedingungen verwendet. Verlässt ein Teilchen in x -Richtung die Box, wird es wieder einsortiert und erhält eine Verschiebung um d in z -Richtung. Zudem wird die Geschwindigkeit eines Teilchens dem Geschwindigkeitsprofil angepasst. Der schwarze Punkt in der Mitte der schraffierten Simulationsbox markiert den Ursprung. Die unschraffierten Boxen zeigen die virtuellen Kopien der Simulationsbox, wobei die obere und die untere Reihe entsprechend der Scherung mit der Flussgeschwindigkeit v_s relativ zum Ursprung bewegt werden.

Die Randbedingungen dienen dazu, bei der Zeitentwicklung die Teilchen innerhalb des Simulationsgebietes zu behalten. Entsprechend der Minimum Image Convention wird ein Teilchen, das die Box in y -Richtung verlässt, wie in den periodischen Randbedingungen an der gegenüberliegenden Seite wieder einsortiert. In x -Richtung erfährt ein Teilchen eine Verschiebung um einen Abstand d in z -Richtung, wobei $d = \dot{\gamma} \cdot L_z \cdot t_{MD}$. Dabei ist L_z die Breite der Simulationsbox in z -Richtung; t_{MD} bezeichnet die aktuelle Simulationszeit in Simulationseinheiten.

Zusätzlich zur Verschiebung wird das Scherprofil angepasst, d. h. einem Teilchen wird die Schergeschwindigkeit hinzugefügt oder abgezogen, je nachdem, ob es das Simulationsgebiet in positiver oder negativer x -Richtung verlässt. Dabei ist $v_s = \dot{\gamma} \cdot L_z$ die Scherge-

schwindigkeit und damit die Geschwindigkeit, mit der sich die in x -Richtung hinzugefügten Boxen bewegen. Der Ursprung der Simulation wird auf die Boxmitte gesetzt.

Die Randbedingungen für eine Simulationsbox mit den Ausmaßen $L_x \times L_y \times L_z$ und dem Ursprung in der Mitte der Box können entsprechend der *Minimum Image Convention* wie folgt als Algorithmus dargestellt werden:

$$\begin{aligned}
 x > \frac{L_x}{2} : \quad x &-= L_x, z &-= d, \quad v_z &-= v_s \\
 x < -\frac{L_x}{2} : \quad x &+= L_x, z &+= d, \quad v_z &+= v_s \\
 y > \frac{L_y}{2} : \quad y &-= L_y, \\
 y < -\frac{L_y}{2} : \quad y &+= L_y, \\
 z > \frac{L_z}{2} : \quad z &-= L_z, \\
 z < -\frac{L_z}{2} : \quad z &+= L_z.
 \end{aligned}$$

Dabei ist es besonders wichtig, innerhalb des Programms die Reihenfolge korrekt zu wählen, da bei einer Überschreitung der x -Grenze auch die z -Komponente des Teilchens korrigiert wird. Außerdem ist – anders als bei den periodischen Randbedingungen – in der z -Komponente eine mehrfache Abfrage nötig, da durch die Verschiebung d die z -Komponente um mehr als $2L_z$ anwachsen kann.

Durch Anwenden der Lees-Edwards-Randbedingungen stellt sich bei der Simulation eines Fluids mit kleiner Reynolds-Zahl ein lineares Geschwindigkeitsprofil ein. Da dieses Profil lediglich durch die Randbedingungen vorgegeben wird, wird es nur auf das System angewendet, wenn ein Teilchen das Simulationsgebiet in x -Richtung verlässt. Dadurch ist die Relaxationszeit dieser Systeme, bis sich das lineare Geschwindigkeitsprofil erfolgreich eingestellt hat, höher als in ungescherten Systemen. Zur Beschleunigung dieser Einschwingprozesse wird im Unterabschnitt 2.2.1 der SLLOD-Algorithmus vorgestellt, der das lineare Scherprofil direkt an die Bewegungsgleichungen koppelt und damit die Relaxationszeit reduzieren soll. Dieser Algorithmus wird (bislang) nur auf die Simulationen angewandt, die mit dem isokinetischen Thermostat und dem Langevin-Thermostat temperiert werden.

2.2.1 SLLOD-Algorithmus

Um ein System zu scheren, werden die Lees-Edwards-Randbedingungen angewendet, vgl. dazu Abschnitt 2.2. Zusätzlich dazu kann das Scherprofil an die Bewegungsgleichungen gekoppelt werden, sodass sich das Scherprofil schneller und gleichmäßiger einstellt. Erste Ideen hierzu wurden von Hoover et al. in [23, 24] vorgestellt. Der in dieser Arbeit verwendete Algorithmus, der sogenannte SLLOD-Algorithmus [16, 24], ist eine Weiterentwicklung dieser Ideen und wird im Folgenden erläutert.

Die abgeänderten Bewegungsgleichungen lauten:

$$\dot{\mathbf{q}} = \mathbf{p} + \nabla \mathbf{u} \cdot \mathbf{q}, \quad (2.8)$$

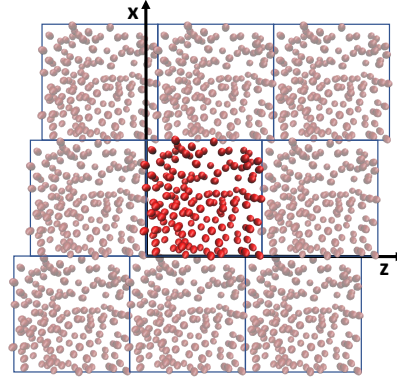


Abb. 2.5: Momentaufnahme einer Konfiguration, welche mithilfe des isokinetischen Thermostats und dem SLLOD-Algorithmus erzeugt wurde. Die eigentliche Simulationsbox in der Mitte wird von den virtuellen Boxen umrahmt, visuell durch eine leichte Transparenz der Teilchen hervorgehoben. Die in x -Richtung hinzugefügten Boxen werden gemäß den Randbedingungen in z -Richtung verschoben. Aus Gründen der Übersichtlichkeit ist die y -Richtung nicht mit eingeblendet.

$$\dot{\mathbf{p}} = \frac{\mathbf{f}}{m} - \nabla \mathbf{u} \cdot \mathbf{p} . \quad (2.9)$$

In diesem Gleichungssatz ist $\nabla \mathbf{u}$ die Matrix-Repräsentation der angelegten Spannung. Da in der xy -Ebene in z -Richtung gesichert wird, nimmt $\nabla \mathbf{u}$ die folgende Form an:

$$\nabla \mathbf{u} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ \dot{\gamma} & 0 & 0 \end{pmatrix} . \quad (2.10)$$

Die Bewegungsgleichungen werden in der Implementierung um einen Halbschritt und die entsprechenden neuen Terme des SLLOD-Algorithmus erweitert. Im Nachfolgenden wird dies an einem Gleichungssatz der generalisierten Koordinaten $\mathbf{r}(t + 1/2\Delta t)$ gezeigt. Dieser Halbschritt wird anschließend in Gleichungen des ganzen Schrittes eingesetzt. Für das gegebene $\Delta \mathbf{u}$, sieht der Halbschritt wie folgt aus:

$$\mathbf{r}\left(t + \frac{1}{2}\Delta t\right) = \mathbf{r}(t) + \frac{1}{2}\Delta t (\mathbf{v}(t) + \dot{\gamma} (r_x(t) \cdot \hat{e}_z)) , \quad (2.11)$$

$$\mathbf{v}\left(t + \frac{1}{2}\Delta t\right) = \mathbf{v}(t) + \frac{1}{2}\Delta t \left(\frac{\mathbf{F}(t)}{m} - \dot{\gamma} (v_x(t) \cdot \hat{e}_z) \right) . \quad (2.12)$$

Der momentane Zeitschritt in den Gleichungen wird mit t gekennzeichnet, die Ebene des Halbschrittes ist demnach $t + 1/2\Delta t$, die des ganzen Schrittes $t + \Delta t$. Der Index x bei Position \mathbf{r} und Geschwindigkeit \mathbf{v} bezeichnet die x -Koordinate des jeweiligen Vektors.

Analog zur Anpassung des Halbschrittes müssen die Bewegungsgleichungen für die generalisierten Koordinaten \mathbf{r} und Geschwindigkeiten \mathbf{v} um den Halbschritt $\mathbf{r}(t + 1/2\Delta t)$ erweitert werden. Entsprechend dem Zeitniveau muss auch $\mathbf{v}(t + 1/2\Delta t)$ verwendet werden und man erhält:

$$\mathbf{r}(t + \Delta t) = \mathbf{r}(t) + \Delta t \left(\mathbf{v} \left(t + \frac{1}{2} \Delta t \right) + \dot{\gamma} \left(\mathbf{r}_x(t + \frac{1}{2} \Delta t) \cdot \hat{\mathbf{e}}_z \right) \right), \quad (2.13)$$

$$\mathbf{F}(t + \Delta t) = -\nabla U(|\mathbf{r}(t + \Delta t)|), \quad (2.14)$$

$$\mathbf{v}(t + \Delta t) = \mathbf{v} \left(t + \frac{1}{2} \Delta t \right) + \frac{1}{2} \Delta t \left(\frac{\mathbf{F}(t + \Delta t)}{m} - \dot{\gamma} (\mathbf{v}_x(t + \Delta t) \cdot \hat{\mathbf{e}}_z) \right). \quad (2.15)$$

Die Methode macht sich das Crank-Nicolson-Verfahren zunutze. Wie in [25] gezeigt, erlaubt diese Methode größere Zeitschritte und führt zu einem um eine Größenordnung kleineren Fehler.

Der SLLOD-Algorithmus ist exakt, kann aber nicht aus den Hamiltonschen Gleichungen hergeleitet werden. Dieser Umstand wird auch in [24] diskutiert.

2.3 VIRIALDRUCK UND STRESSTENSOR IN SIMULATIONEN

In Schersimulationen ist man vor allem an den einzelnen Komponenten des Drucktensors interessiert. In diesem Abschnitt soll zunächst in grober Anlehnung an [26] über den Virialsatz ein Ausdruck für den Virialdruck hergeleitet werden, im Anschluss wird dieser Ausdruck tensoriell verallgemeinert.

Gegeben ist ein System mit Volumen V , indem sich N Teilchen befinden. Damit ergibt sich eine durchschnittliche Teilchendichte von $\rho = N/V$. Zwei Teilchen, deren Abstand mit \mathbf{r}_{ij} beschrieben wird, wechselwirken über ein Paarpotential $U(\mathbf{r})$.

Der Ausdruck, der hergeleitet werden soll, ist dabei

$$p = k_B T \rho + \frac{1}{3V} \left\langle \sum_i \sum_{i < j} (\mathbf{r}_i - \mathbf{r}_j) \cdot \mathbf{f}_{ij} \right\rangle, \quad (2.16)$$

wobei $\mathbf{r}_i - \mathbf{r}_j$ der Abstand des i -ten zum j -ten Teilchen ist. \mathbf{f}_{ij} bezeichnet die Kraft, welche das j -te Teilchen auf das i -te auswirkt und resultiert aus dem Paarpotential: $\mathbf{f}_i = \mathbf{f}(\mathbf{x}_i) = -\partial U / \partial \mathbf{x}_i$. Zudem gilt Newtons drittes Axiom, sodass $\mathbf{f}_{ij} = -\mathbf{f}_{ji}$. Diese Symmetrie sowie die Symmetrie der Teilchenabstände wird durch die Einschränkung der zweiten Summe berücksichtigt, sodass die Kraft zweier Teilchen aufeinander immer nur einmal beitragen kann.

Die Herleitung des Drucks einer Flüssigkeit wird nun über das Virialtheorem von Clausius [27] vorgenommen, indem ein generalisiertes *Äquipartitionstheorem* berücksichtigt wird. Dies gilt auch für Potentialkräfte mit Reibungsterm, sofern sich ein Fließgleichgewicht einstellt, siehe [28]. Letzteres führt zu folgenden Relationen:

$$\left\langle \mathbf{x}_i \frac{\partial H}{\partial \mathbf{x}_i} \right\rangle = k_B T, \quad (2.17)$$

$$\left\langle \mathbf{p}_i \frac{\partial H}{\partial \mathbf{p}_i} \right\rangle = k_B T, \quad (2.18)$$

wobei \mathbf{x}_i und \mathbf{p}_i Koordinaten und zugehörigen Impuls beschreiben. Die Hamiltonfunktion H des Systems ist über die Summe aus einem kinetischen und einem potentiellen Energieteil gegeben, sodass $H = H_{\text{kin}} + V(\mathbf{r})$. $V(\mathbf{r})$ wird durch die Teilchenwechselwirkung untereinander und der Wechselwirkung mit äußeren Kräften bestimmt. Damit folgt aus (2.18):

$$\frac{1}{2} \left\langle \mathbf{p}_i \frac{\partial H}{\partial \mathbf{p}_i} \right\rangle = \left\langle \frac{\mathbf{p}_i^2}{2m} \right\rangle = \frac{1}{2} k_B T. \quad (2.19)$$

Für jedes Teilchen trägt jeder Freiheitsgrad $\frac{1}{2}k_B T$ zur Bewegungsenergie bei. Unter Verwendung von (2.17) führt dies zu:

$$\langle H_{kin} \rangle = \left\langle \sum_i \frac{p_i^2}{2m} \right\rangle = \frac{1}{2} \sum_i \left\langle \mathbf{x}_i \frac{\partial V}{\partial \mathbf{x}_i} \right\rangle . \quad (2.20)$$

Das Potential V lässt sich als Summe eines externen Potentials W und eines internen Wechselwirkungspotentials U darstellen, welches nur die Wirkung der Teilchen aufeinander berücksichtigt und zuvor vorgestellt wurde, sodass man für $V = U + W$ schreiben kann und es ergibt sich:

$$\langle H_{kin} \rangle = \frac{1}{2} \sum_i \left\langle \mathbf{x}_i \frac{\partial (U + W)}{\partial \mathbf{x}_i} \right\rangle \quad (2.21)$$

$$= \frac{1}{2} \left(\sum_i \left\langle \mathbf{x}_i \frac{\partial U}{\partial \mathbf{x}_i} \right\rangle + 3pV \right) . \quad (2.22)$$

Den letzten Term erhält man über die Annahme eines Wandpotentials der Form:

$$W = W_{inf} \sum_i^N \{ \Theta(x_{i,1} - L_1) + \Theta(x_{i,2} - L_2) + \Theta(x_{i,3} - L_3) \} , \quad (2.23)$$

wobei N die Anzahl der Teilchen, W_{inf} die Größe der Barriere, die die Wand darstellt, und L_i die Dimensionen der Box. In den hier vorgenommenen Simulationen wird eine Erweiterung der periodischen Randbedingungen vorgenommen, auf diese Weise wird dem System vorgetäuscht, in einem größeren System eingebettet zu sein. Diese Wände sind also hypothetisch, da sie in großer Entfernung zur Simulationsbox sind.

Aufgrund der Heaviside-Funktion Θ gilt:

$$\frac{\partial W}{\partial L_1} = -W_{inf} \sum_i \delta(x_{i,1} - L_1) , \quad (2.24)$$

und deshalb

$$\left\langle x_{i,1} \frac{\partial W}{\partial x_{i,1}} \right\rangle = \left\langle \sum_i W_{inf} \delta(x_{i,1} - L_1) \right\rangle = \dots = \left\langle L_1 \frac{\partial W}{\partial L_1} \right\rangle . \quad (2.25)$$

Mit der Einsteinschen Summenkonvention kann diese Ableitung auf das Volumen ausgeweitet werden, sodass gilt

$$pV = \frac{1}{3} \left\langle \sum_{i,\alpha} x_{i,\alpha} \frac{\partial W}{\partial x_{i,\alpha}} \right\rangle . \quad (2.26)$$

Dabei beschreibt p den Druck des Systems.

Die Ableitung des Potentials aus Gleichung (2.22) kann durch die Kraft ersetzt werden, die Summe über die paarweise Wechselwirkung kann also als $\mathbf{f}_i = \mathbf{f}(\mathbf{x}_i) = -\partial U / \partial \mathbf{x}_i$ geschrieben werden. Da das Potential vom Abstand zweier Teilchen abhängt, kann $\mathbf{x}_i = \mathbf{r}_i$ als Ortsvektor des i -ten Teilchens gesetzt werden.

Dies führt zu:

$$\sum_i \langle \mathbf{r}_i \mathbf{f}_i \rangle = \sum_i \sum_{j \neq i} \langle \mathbf{r}_i \mathbf{f}_{ij} \rangle = \sum_i \sum_{j \neq i} \langle \mathbf{r}_i \cdot \mathbf{f}_{ij} \rangle + \langle \mathbf{r}_j \cdot \mathbf{f}_{ji} \rangle \quad (2.27)$$

$$= \frac{1}{2} \sum_i \sum_{j \neq i} \langle (\mathbf{r}_i - \mathbf{r}_j) \cdot \mathbf{f}_{ij} \rangle = \sum_i \sum_{j > i} \langle (\mathbf{r}_i - \mathbf{r}_j) \cdot \mathbf{f}_{ij} \rangle . \quad (2.28)$$

Hierbei wird Newtons drittes Axiom $\mathbf{f}_{ij} = -\mathbf{f}_{ji}$ angewendet. Im letzten Schritt werden die Summengrenzen angepasst, indem die Symmetrie des Teilchenabstands benutzt wird. Damit kann der Druck wie folgt ausgedrückt werden:

$$\langle H_{\text{kin}} \rangle = \frac{1}{2} \sum_i \left\langle \mathbf{x}_i \frac{\partial U}{\partial \mathbf{x}_i} \right\rangle + \frac{3}{2} p V \quad (2.29)$$

$$\frac{1}{2} N k_B T = -\frac{1}{2} \sum_i \sum_{j>i} \langle (\mathbf{r}_i - \mathbf{r}_j) \cdot \mathbf{f}_{ij} \rangle + \frac{3}{2} p V \quad (2.30)$$

$$\rightarrow p = k_B T \rho + \frac{1}{3V} \left\langle \sum_i \sum_{j>i} (\mathbf{r}_i - \mathbf{r}_j) \cdot \mathbf{f}(\mathbf{r}_i - \mathbf{r}_j) \right\rangle. \quad (2.31)$$

In inhomogenen Systemen hängt der Druck für gewöhnlich von der Raumrichtung und der Position \mathbf{r} ab, an der er bestimmt werden soll. In diesem Fall handelt es sich um einen Drucktensor, der nun hergeleitet werden soll.

Über den Zusammenhang aus Temperatur und kinetischer Energie, wie in Gleichung (3.1) notiert, ergibt sich in drei Dimensionen:

$$\begin{aligned} \frac{3}{2} N k_B T &= \frac{1}{2} \sum_i^N m_i v_i^2 \quad | \div 3V \\ \frac{N}{V} k_B T &= \frac{1}{3V} \sum_i^N m_i v_i^2 \\ \Rightarrow p &= \frac{1}{3V} \sum_i^N m_i v_i^2 + \frac{1}{3V} \left\langle \sum_i \sum_{j>i} (\mathbf{r}_i - \mathbf{r}_j) \cdot \mathbf{f}(\mathbf{r}_i - \mathbf{r}_j) \right\rangle \end{aligned} \quad (2.32)$$

Um die Schreibweise zu vereinfachen, wird im Nachfolgenden die aus der Normierung folgende 3 mit dem Volumen verrechnet.

Fasst man nun die Vektoren als Tensoren auf, die Vektorprodukte als dyadisches Produkt und definiert den Stresstensor als negativ, erhält man den in dieser Arbeit verwendeten Ausdruck des Stresstensors

$$\boldsymbol{\sigma} = -\frac{1}{V} \left(\sum_i^N (m_i \mathbf{v}_i \otimes \mathbf{v}_i) + \sum_i^N \sum_{j>i}^N (\mathbf{r}_{ij} \otimes \mathbf{F}_{ij}) \right), \quad (2.33)$$

bzw. in Komponentenschreibweise

$$\sigma_{\alpha\beta} = -\frac{1}{V} \left(\sum_i^N (m_i v_{i,\alpha} v_{i,\beta}) + \sum_i^N \sum_{j>i}^N (r_{ij,\alpha} F_{ij,\beta}) \right). \quad (2.34)$$

Für den Stresstensor $\boldsymbol{\sigma}$ gilt außerdem folgende Relation zwischen dem Druck eines Systems p und dem zähen Spannungstensor $\boldsymbol{\tau}$:

$$\boldsymbol{\sigma} = p \mathbf{I} + \boldsymbol{\tau}. \quad (2.35)$$

Entsprechend lässt sich der Druck des Systems über die Spur des Stresstensors bestimmen als

$$p = \frac{1}{3} \text{Sp}(\boldsymbol{\sigma}). \quad (2.36)$$

Das Virialtheorem kann nicht zur Druckbestimmung in räumlich inhomogenen Systemen angewandt werden. Außerdem kann der Virialausdruck nur angewendet werden,

wenn die Wechselwirkung durch ein differenzierbares und paarweises Potential beschrieben werden kann. Entsprechend werden harte Kugeln über solch ein Potential angenähert, wie in Kapitel 4 näher erläutert.

Eine wichtige Größe, die aus dem Stresstensor berechnet wird, ist die (dynamische) Viskosität eines Systems η . Diese ist ein Maß für die Zähflüssigkeit eines Fluids und beschreibt das Verhalten des Fluids unter Einwirkung einer äußeren Kraft, die das System schert. Je niedriger die Viskosität, desto dünnflüssiger ist die Flüssigkeit oder das Gas. Wird auf ein Fluid die Spannung τ ausgeübt, so ist die Viskosität der Proportionalitätsfaktor zwischen der Spannung und der Scherrate aus (2.7) und es gilt:

$$\tau = \eta \cdot \dot{\gamma} . \quad (2.37)$$

Die an das System angelegte Spannung ist in den hier zugrundeliegenden Simulationen die σ_{13} -Komponente des Cauchy-Stress- bzw. Cauchy-Spannungstensors σ , die Viskosität lässt sich also bestimmen über:

$$\eta = \frac{\sigma_{13}}{\dot{\gamma}} . \quad (2.38)$$

Handelt es sich um ein Newtonsches Fluid, ist die Viskosität konstant und unabhängig von der angelegten Scherrate. In dieser Arbeit werden sowohl Newtonsche Fluide behandelt, als auch Systeme, die bei Anlegen einer Spannung ausdünnen. Die Ergebnisse für Fluide mit Newtonschem Verhalten sind in Kapitel 6 aufgeführt, die Untersuchung zu ausdünnenden Systemen ist in Kapitel 7 zu finden.

THERMOSTATE

Einem System werde Energie in Form einer äußeren Einwirkung zugefügt. Durch die daraus resultierende Reibung erhöht sich die Temperatur des Systems. Um diese bis auf Fluktuationen der Ordnung $\mathcal{O}(1/N)$ konstant zu halten, wird ein Thermostat eingeführt.

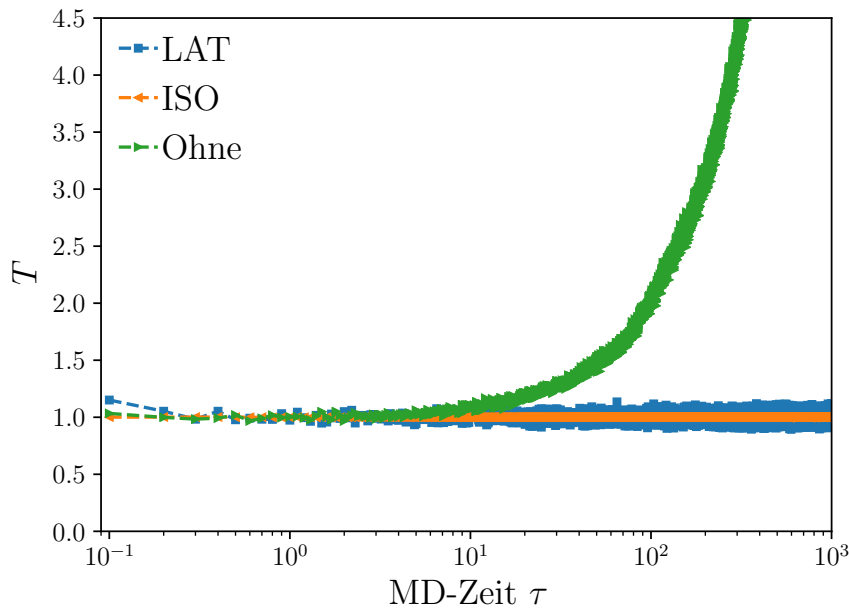


Abb. 3.1: Temperaturverlauf mit und ohne Thermostate. Durch die konstante Energiezufuhr erhöht sich die kinetische Energie und damit die Temperatur eines gescherten Systems bereits nach kurzer Simulationszeit τ . Der entsprechende Datenverlauf ist in grün eingezeichnet. Für die mit einem der in diesem Kapitel vorgestellten Thermostate gekühlten Systeme (blaue und gelbe Kurve) bleibt die Temperatur abgesehen von kleinen Fluktuationen konstant. Das Lowe-Andersen-Thermostat (LAT, blau) unterliegt dabei höheren Schwankungen als das isokinetische Thermostat (gelb), da letzteres lediglich die Geschwindigkeiten und damit die Temperatur reskaliert.

In den dieser Arbeit zugrundeliegenden Problemstellungen wird das System konstant geschert. Durch diese äußere Einwirkung wird dem System permanent Energie zugeführt, sodass die Temperatur ohne ein Thermostat stetig anwächst, wie in Abbildung 3.1 dargestellt. Die grüne Kurve stellt die Temperatur des gescherten Systems ohne Thermostat dar. Wie nicht anders zu erwarten, wächst sie kontinuierlich als Funktion der Zeit. Die anderen beiden Kurven sind Systemen entnommen, die mit den in den nachfolgenden Abschnitten vorgestellten Thermostaten gekühlt werden. Das Lowe-Andersen-Thermostat (LAT), das ein physikalisches Thermostat darstellt, weist dabei höhere Schwankungen auf als das isokinetische Thermostat, welches lediglich die Geschwindigkeiten der einzelnen Teilchen reskaliert.

Durch Anwenden eines Thermostats wird physikalisch motiviert – wie beim Lowe-Andersen-Thermostat – oder künstlich – wie beim isokinetischen Thermostat – die Temperatur kontrolliert. Die Temperatur hängt mit der kinetischen Energie wie folgt zusammen:

$$\frac{\Phi}{2} N k_B T = \frac{1}{2} \sum_i^N m_i v_i^2, \quad (3.1)$$

wobei Φ die Anzahl der Freiheitsgrade des Systems anzeigt. In drei Dimensionen ist $\Phi = 3$, in zwei Dimensionen gilt $\Phi = 2$. Damit lässt sich die momentane Temperatur T eines Systems mit N Teilchen bestimmen, wobei Teilchen i eine Geschwindigkeit v_i und Masse m_i besitzt. Wenn die durchschnittliche kinetische Energie durch Scherung erhöht wird, erhöht sich damit auch die Temperatur.

Im Anschluss werden zwei mögliche Thermostate behandelt, die in dieser Arbeit auf ihre Funktionalität für ein Hybridschema getestet werden. Das Lowe-Andersen-Thermostat stellt dabei ein Galilei-invariantes Thermostat dar, das zusätzlich den Gesamtimpuls des Systems erhält. Ein anderes, sehr einfaches Thermostat ist das isokinetische, welches die Geschwindigkeiten der Teilchen und damit die Temperatur lediglich reskaliert und dadurch keinem physikalischen Thermostat entspricht. In der Literatur wird letzteres auch als Gaußsches Thermostat betitelt [29].

Auch das Langevin-Thermostat [30] wird vorgestellt, mit dem später in Abschnitt 6.1 die Ergebnisse des isokinetischen Thermostats verifiziert werden.

3.1 DAS LOWE-ANDERSEN-THERMOSTAT

Das Lowe-Andersen-Thermostat (LAT) [31] ist Galilei-invariant und im Gegensatz zum Andersen-Thermostat [31, 32] impulserhaltend. In der Simulation wird der Thermostatalgorithmus durchgeführt, nachdem die Bewegungsgleichungen für einen Zeitschritt mithilfe des Velocity-Verlet-Algorithmus gelöst sind, siehe Abschnitt 2.1. Dabei wechselwirken die Simulationsteilchen mit einem angeschlossenen Wärmebad. Um diese Wechselwirkung physikalisch realistisch zu simulieren, geschieht eine solche Kollision und somit ein Austausch der Geschwindigkeiten zwischen „Badteilchen“ und Simulationsteilchen mit Kollisionsfrequenz Γ und damit einer Kollisionswahrscheinlichkeit der Form $w = \Gamma \Delta t$ (mit $0 < w \leq 1$), wobei Δt den Zeitschritt der Simulation darstellt.

Anders als das Andersen-Thermostat wirkt das Lowe-Andersen-Thermostat stets paarweise auf zwei Teilchen, die sich in einem Interaktionsradius $r_{ij} < r_C$ zueinander befinden. Die beiden Teilchen erhalten einen Stoß entlang ihrer Verbindungslinie, der einer Maxwell-Boltzmann-Verteilung entnommen wird. Der Stoß \mathbf{v}'_{ij} wird wie folgt berechnet:

$$\mathbf{v}'_{ij} = \xi_{ij} \sqrt{2k_B T} \hat{\mathbf{r}}_{ij}, \quad (3.2)$$

wobei ξ_{ij} eine Zufallszahl aus einer Normalverteilung mit einer Varianz von 1 ist und $\hat{\mathbf{r}}_{ij}$ der normalisierte Abstand von Teilchen i zu Teilchen j . Daraus ergeben sich die neuen Geschwindigkeiten von Teilchen i und j nach der Badkollision zu:

$$\mathbf{v}'_i = \mathbf{v}_i + \Delta_{ij} \quad (3.3)$$

und

$$\mathbf{v}'_j = \mathbf{v}_j - \Delta_{ij}, \quad (3.4)$$

wobei

$$2\Delta_{ij} = \hat{\mathbf{r}}_{ij} (\mathbf{v}'_{ij} - \mathbf{v}_{ij}) \cdot \hat{\mathbf{r}}_{ij}. \quad (3.5)$$

Da die beiden Teilchen einen entgegengesetzten Kick bekommen, bleibt der Impuls bei dieser Operation erhalten.

Die beiden zu untersuchenden Parameter im Lowe-Andersen-Thermostat sind die Badkollisionsfrequenz Γ und der Interaktionsradius r_C . Die Kollisionsfrequenz bestimmt dabei die Wahrscheinlichkeit, mit der ein Simulationsteilchen mit dem Wärmebad interagiert, also die Anzahl an Kollisionen eines Teilchens mit seinen Nachbarn, welche sich innerhalb des Interaktionsradius befinden. Diese Wahrscheinlichkeit entspricht dem Produkt aus Badkollisionsfrequenz und Zeitschritt $w = \Gamma \Delta t$.

Im Programmcode wird dies darüber realisiert, dass für jedes Teilchen i eine Liste der Nachbarn bestimmt wird, welche sich innerhalb des Interaktionsradius r_C befinden. Diese Liste an m Nachbarn wird anschließend in einer zufälligen Reihenfolge sortiert. Das Teilchen i wechselwirkt dann nur mit den ersten $m \cdot w$ Nachbarn, die sich in der Liste befinden. Die Wahrscheinlichkeit ist also so zu wählen, dass sie nicht größer als $w = 0.5$ ist, da jedes Teilchen auch in der Nachbarliste jedes Nachbarn steht.

Da in jedem Schritt durch den Stoß nachträglich die Geschwindigkeiten geändert werden, bewegt sich ein Teilchen i an eine andere Position, als vorher durch die Bewegungsgleichungen bestimmt. Im Rahmen der Simulation ist diese Abweichung minimal, dennoch „behindern“ sich die Teilchen unter Umständen oder nähern sich stärker, als laut Bewegungsgleichungen vorgesehen, wie in Abbildung 3.2 angedeutet. Dort sind die aus den momentanen Geschwindigkeiten resultierenden Positionen nach dem Velocity-Verlet-Schritt angedeutet. Durch den Kick ändern sich die Geschwindigkeiten, wodurch sich auch die Positionen im nächsten Zeitschritt ändern.

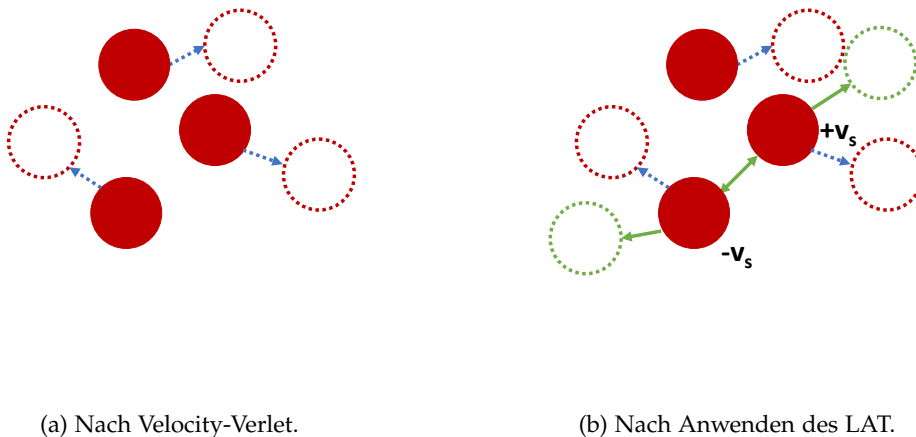


Abb. 3.2: Bewegung der Teilchen (a) nach Velocity-Verlet-Schritt und (b) nach Anwendung des Lowe-Andersen-Thermostats. Die roten Kugeln sind die Teilchen zu einem Zeitschritt n . Die blauen Pfeile entsprechen den jeweiligen Geschwindigkeiten zum Zeitschritt n . Da die Teilchen durch Anwenden des Thermostats in (b) zum Teil einen nicht durch die Bewegungsgleichungen vorgegebenen Geschwindigkeitsschub (grüner Pfeil zwischen den Teilchen) erhalten, können sie im nächsten Zeitschritt der Bewegung eines anderen Teilchens in die Quere kommen oder sich stärker annähern, als vorgesehen. Die resultierenden Geschwindigkeiten sind als grüne Pfeile dargestellt, die neuen Positionen im nächsten Schritt als grüne Kugeln. Dadurch erhöht sich die Viskosität, da die mittlere freie Weglänge vermindert wird. In den Simulationen ist dieser Kick wesentlich niedriger; das hier gezeigte Schaubild dient der Veranschaulichung des Effekts.

Durch diesen Effekt erhöht sich die mittlere freie Weglänge des Fluids. Dies hat zur Folge, dass sich die dynamische Viskosität ändert, genauer gesagt erhöht. Dieser Umstand ist auch in [31] beschrieben. Der Zusammenhang zwischen zusätzlicher Viskosität η_t und der Badkollisionsfrequenz Γ , sowie dem Kollisionsradius r_C lautet

$$\eta_t \sim \frac{\pi \rho^2 r_C^5 \Gamma}{75 m} \quad (3.6)$$

und wird in [31] hergeleitet. Sofern dieser Zusatzterm der Viskosität verglichen zur Viskosität des nicht thermostierten System groß ist, verlangsamt die erhöhte Gesamtviskosität die Dynamik des Systems, wie auch in [32] erklärt. Der Zusammenhang zwischen Viskosität und den Parametern des Lowe-Andersen-Thermostats wird im Abschnitt 6.1 gezeigt.

3.2 DAS ISOKINETISCHE THERMOSTAT

Das zweite verwendete Thermostat ist das isokinetische Thermostat (ISO) [16], welches in der Literatur auch als Gaußsches Thermostat bezeichnet wird [29]. Es findet häufig bei Hybridsimulationsschemata Verwendung, da die Einschwingvorgänge für diese Art der Problemstellung irrelevant sind und man nur am stabilen Zustand (engl.: *Steady State*), dem sogenannten Fließgleichgewicht [33] interessiert ist.

Bei diesem Verfahren wird nach jedem Velocity-Verlet-Integrationsschritt die momentane Temperatur des Systems unter Zuhilfenahme der kinetischen Energie berechnet:

$$\frac{3}{2} N k_B T = \sum_i^N m_i v_i^2. \quad (3.7)$$

Anschließend wird die Geschwindigkeit reskaliert, sodass die Durchschnittstemperatur der vorher eingestellten Temperatur T_0 entspricht. Dieser Faktor berechnet sich aus

$$\lambda = \sqrt{\frac{T_0}{T}}. \quad (3.8)$$

Das Thermostat entspricht also einem Abbremsen der Simulationsteilchen, alle Geschwindigkeiten werden reskaliert. Der Impuls ist dabei nicht erhalten.

Das lineare Scherprofil wird vor Anwendung des Thermostats abgezogen und nach der Reskalierung wieder aufaddiert. Zudem muss vor Beginn der Simulation, also nach der Initialisierung der Startkonfiguration, die Schwerpunktgeschwindigkeit abgezogen bzw. auf Null gesetzt werden, da sonst Effekte wie der sogenannte *Flying Ice Cube* auftreten, der 1998 zum ersten Mal von Harvey et al. vorgestellt wurde [34].

In den hier aufgeführten Simulationen ist die eingestellte Temperatur $T_0 = 1 k_B T$, sofern nicht anders gekennzeichnet.

3.3 DAS LANGEVIN-THERMOSTAT

Das Lowe-Andersen-Thermostat ändert die Viskosität des Systems, indem es je nach Wahl der Parameter stark in die Dynamik des Systems eingreift. In [2] wird argumentiert, dass für die Hybridsimulationen lediglich der stabile Zustand des Systems relevant ist, sodass das isokinetische Thermostat Anwendung finden kann. Um zu überprüfen, ob diese Wahl trotz einer unphysikalischen Grundlage zutreffend ist, wird in Abschnitt 6.1 untersucht, ob auch mithilfe des sogenannten Langevin-Thermostats [30] vergleichbare Ergebnisse

erzielt werden können. Die Grundlagen dieses Thermostats werden hier beschrieben und folgen der Beschreibung in [7].

Anders als bei den vorher vorgestellten Thermostaten koppelt das Langevin-Thermostat direkt an die Bewegungsgleichungen, deswegen spricht man von der Langevin-Dynamik im Gegensatz zur Newtonschen Dynamik. Die natürliche Reibung eines Stoffes mit Luft oder den Bestandteilen der Lösung, in der die Teilchen sich befinden, wird berücksichtigt. Dabei können bestimmte Effekte nicht beachtet werden, hierzu zählen unter anderem hydrophobes Verhalten oder auch hydrodynamische Effekte bei dichten Lösungen.

Die Newtonschen Bewegungsgleichungen werden durch einen Reibungsterm ergänzt, der das Abbremsen der simulierten Teilchen durch Reibungsverluste berücksichtigt. Gleichzeitig wird die zufällige Interaktion mit Lösungsteilchen durch einen zufallsbasierten Term \mathbf{F}_Z realisiert, der die kinetische Energie sowohl erhöhen, wie auch abschwächen kann. Insgesamt ergibt sich die Langevin-Gleichung eines Teilchens zu:

$$m\ddot{\mathbf{r}} = -\nabla U(\mathbf{r}) - m\gamma\dot{\mathbf{r}} + \mathbf{F}_Z . \quad (3.9)$$

Der positive Reibungskoeffizient γ ist ein zuvor festzulegender Parameter, der die Stärke der Reibung festlegt und damit die Temperatur des Systems senkt.

Die zufällige Kraft \mathbf{F}_Z hängt sowohl von der Systemtemperatur T_0 als auch dem Reibungskoeffizienten γ ab, es gilt:

$$\mathbf{F}_Z = \sqrt{2\gamma k_B T_0} \mathbf{R}(t) . \quad (3.10)$$

Dabei entspricht $\mathbf{R}(t)$ einem Vektor aus zufälligen Werten, die aus einer Gaußverteilung gezogen werden, die wiederum das Fluktuations-Dissipationsgesetz erfüllt, welches in [30] vorgestellt wird, für die Komponenten i und j von $\mathbf{R}(t)$ gilt:

$$\langle R_i(t) R_j(t') \rangle = \delta_{ij} \delta(t - t') , \quad (3.11)$$

mit $\delta(t)$ der Delta-Distribution und δ_{ij} dem Kronecker-Delta.

Je nach Wahl des Reibungskoeffizienten kann unterschiedliches physikalisches Verhalten simuliert werden. Dabei können Simulationen vom „inertialen“ Regime bei niedrigen Werten für γ , bei dem viskose Effekte vernachlässigt werden können, bis hin zum diffusen Regime bei hohen Werten für γ , welches auch mithilfe der Brownschen Dynamik beschrieben werden kann, simuliert werden.

Zur Implementierung des Langevin-Thermostats werden die einzelnen Schritte der MD-Simulation wie in Abschnitt 2.1 dargestellt angewendet. Dabei wird der gesamte Algorithmus zur Integration der Bewegungsgleichungen in ein zweischrittiges Verfahren aufgeteilt, sodass die einzelnen Methoden für verschiedene Situationen verwendet werden können. So wird der nachfolgende erste Teil der Positionsbestimmung auch für das Lowe-Andersen-Thermostat verwendet. Beim isokinetischen Thermostat sind die Bewegungsgleichungen zusätzlich über den SLLOD-Algorithmus direkt mit dem Scherprofil verknüpft, wie in Abschnitt 2.2.1 beschrieben. Der zweite Schritt unterscheidet sich lediglich dahingehend, dass Kräfte und Geschwindigkeiten bezüglich der Newtonschen Bewegungsgleichungen bestimmt werden (wobei diese wieder für das isokinetische Thermostat um den SLLOD-Algorithmus erweitert werden müssen).

Im ersten Teil des Verfahrens wird die Verschiebung der virtuellen Boxen für die Lees-Edwards-Randbedingungen berechnet und die Zelllisten werden falls nötig aktualisiert. Anschließend wird die Position eines Teilchens bezüglich des neuen Zeitschritts bestimmt:

$$\mathbf{r}(t + \Delta t) = \mathbf{r}(t) + \mathbf{v}(t)\Delta t + \frac{\Delta t^2}{2m} \mathbf{F}_{\text{Ges}}(t) . \quad (3.12)$$

Das Teilchen wird mithilfe der Lees-Edwards-Randbedingungen wieder in das Simulationsgebiet verschoben, sofern es die Box verlassen hat. Dabei ist $F_{\text{Ges}}(t)$ die Gesamtkraft, die auf ein Teilchen wirkt.

Anschließend werden die Kräfte entsprechend den Lees-Edwards-Randbedingungen aus dem Gradienten des Potentials berechnet. Die Gesamtkraft auf ein Teilchen hängt zudem von der Zufallskraft F_Z sowie der momentanen Geschwindigkeit in Form eines Reibungsterms ab. Für die Zufallskraft werden drei zufallsverteilte Größen aus der Verteilung gezogen und mit dem Rauschterm gewichtet. Um den Reibungsterm berücksichtigen zu können, muss folgende Gleichung gelöst werden:

$$\mathbf{v}(t + \Delta t) = \mathbf{v}(t) + \frac{\mathbf{F}_{\text{Ges}}(t) + \mathbf{F}(t + \Delta t) + \mathbf{F}_Z(t + \Delta t) - m\gamma \mathbf{v}(t + \Delta t)}{2m} \Delta t, \quad (3.13)$$

$$= \frac{\mathbf{v}(t) + (\mathbf{F}_{\text{Ges}}(t) + \mathbf{F}(t + \Delta t) + \mathbf{F}_Z(t + \Delta t)) / 2m}{1 + (\gamma \cdot \Delta t) / 2}. \quad (3.14)$$

Dabei sind alle Kräfte des vorhergehenden Zeitschritts in $\mathbf{F}_{\text{Ges}}(t) = \mathbf{F}(t) + \mathbf{F}_Z(t) - m\gamma \mathbf{v}(t)$ zusammengefasst. Auch das Scherprofil muss bei der Berechnung der Geschwindigkeiten berücksichtigt werden. Vorläufig wird dies über eine Implementierung des SLLOD-Algorithmus gewährleistet.

Der Algorithmus ist ursprünglich von Benjamin Trefz in seiner Dissertation [35] in den von Alexander Winkler entwickelten MD-Code [36] eingearbeitet worden und verwendet den so genannten BBK-Integrator [37], der anschließend in die Positions-Geschwindigkeits-Darstellung [38] umgeschrieben wird und weiter zum vollen Zeitintegrationsschritt vereinfacht werden kann. Für diese Arbeit wurde der bereits implementierte Algorithmus auf Scherung erweitert.

Bei den anderen implementierten Thermostaten werden im zweiten Schritt die im vorherigen Kapitel in Gleichungen (2.5) und (2.6) bzw. (2.13) bis (2.15) vorgestellten Bewegungsgleichungen gelöst und anschließend ein Thermostatschritt durchgeführt.

Für die Untersuchungen, die mittels Hybridsimulationen vorgenommen werden können, sind verschiedene physikalische Modelle interessant. Eine Änderung des mikroskopischen Modells führt zu Änderungen der Erwartungswerte im Stresstensor bei gleicher Dichte und einer neuen Wertetabelle, mit der die makroskopische Simulation rechnet.

Aus den den Modellen zugrunde liegenden Wechselwirkungspotentialen zwischen einzelnen Teilchen kann über

$$\mathbf{F}_{ij} = -\nabla U(\mathbf{r}_{ij}) \quad (4.1)$$

die Kraft von Teilchen j auf Teilchen i berechnet werden, wobei $\mathbf{r}_{ij} = \mathbf{r}_i - \mathbf{r}_j$ den Abstand der beiden zueinander bezeichnet. Auf diese Weise lässt sich die Beschleunigung auf ein Teilchen berechnen, mit der die Bewegungsgleichungen aufgestellt werden können, wie in Abschnitt 2.1 hergeleitet wird.

In diesem Kapitel sollen die Modellsysteme beschrieben werden, welche in den MD-Simulationen realisiert werden konnten. Dabei werden neben den physikalisch zugrundeliegenden Modellen die zur Realisierung verwendeten Potentiale beschrieben, aus denen die auf die Teilchen wirkenden Kräfte bestimmt werden können.

4.1 HARTE KUGELN

Eines der am häufigsten verwendeten Potentiale, um die Paarwechselwirkung zweier Teilchen zu beschreiben, ist das Lennard-Jones-Potential (LJ-Potential):

$$U_{LJ}(r) = 4\epsilon \left[\left(\frac{d}{r} \right)^{12} - \left(\frac{d}{r} \right)^6 \right]. \quad (4.2)$$

Hierbei ist d der Durchmesser der Kugeln, ϵ beschreibt die Tiefe des Potentials und damit die Stärke der Attraktion, bzw. Repulsion. In Abbildung 4.1 ist der Verlauf zu sehen. Je nach Wahl der Parameter können mit diesem Potential Edelgase oder auch kolloidale Teilchen beschrieben werden (wenn nur der repulsive Teil berücksichtigt wird). Das Lennard-Jones-Potential findet auch Anwendung als generisches Potential in komplexen Kraftfeldern, z. B. bei der Simulation von Polymeren oder Biomolekülen.

Das Potential modelliert die Wechselwirkung zweier ungebundener, neutraler Teilchen und berücksichtigt sowohl eine langreichweitige Attraktion für große r als auch eine kurzreichweitige Repulsion von Teilchen bei $r < \sqrt[6]{2}d$. Es handelt sich dabei um Kugeln mit einem Durchmesser von d , die sich nicht oder nur marginal überlappen können. Der Durchmesser d beschreibt also die Größenordnung des Systems.

Zwei Teilchen nähern sich einander an, bis ihr Abstand zu gering wird, sie also den repulsiven Teil des Potentials erfahren. Je nach Wahl der Potentialparameter tritt dies ein, wenn die beiden sich berühren oder einen effektiven Abstand zueinander haben. Da das Potential langsam ansteigt, besteht für die Kugeln die Möglichkeit, leicht miteinander zu überlappen, bzw. in die durch z. B. Materialeigenschaften entstehende *Verbotene Zone* einzudringen, in der die Repulsion des anderen Teilchens zu spüren ist. Durch die Wahl

von ϵ kann die Stärke der Abstoßung bzw. auch der Anziehung eingestellt werden. Die Potentialtiefe entspricht der Energieskala des Systems.

Das Lennard-Jones-Potential besitzt eine unendliche Reichweite. Dadurch ist es für MD-Simulationen ressourcenintensiv, da die Kraft zwischen jedem Teilchenpaar berechnet werden muss.

In der Literatur gibt es verschiedene Lösungsansätze, das Potential in variierender Form zu nutzen. Eine Möglichkeit besteht darin, das Potential minimal nach oben zu verschieben und an der daraus resultierenden zweiten Nullstelle für $r \gg d$ das Potential auf Null zu setzen. Dadurch werden jene Kräfte bei großen Abständen vernachlässigt, die zur Gesamtsumme der Kraft kaum beitragen und numerisch vernachlässigt werden können, eine langreichweitige Attraktion wird dadurch abgestellt.

Um realistische harte Kugeln zu simulieren, genügt der rein repulsive Part des Lennard-Jones-Potentials. Damit das Potential kontinuierlich ist und keine Sprungstelle besitzt, wird es entsprechend nach oben verschoben. Dieses Potential wird auch Weeks-Chandler-Anderson-Potential genannt [39].

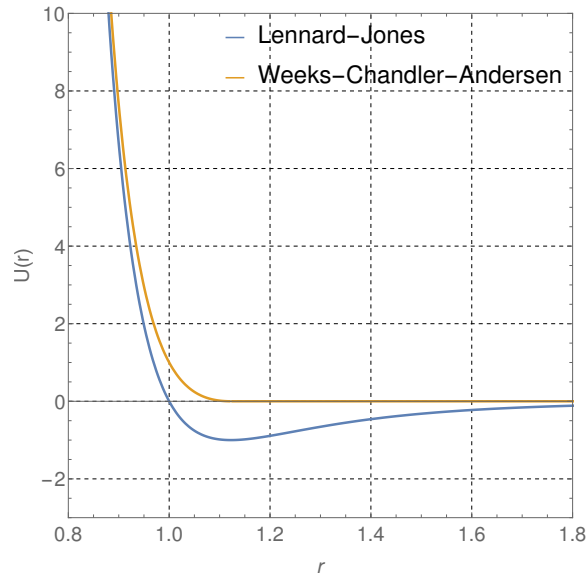


Abb. 4.1: Lennard-Jones- und Weeks-Chandler-Anderson-Potential. Das LJ-Potential hat den Nulldurchgang bei $d = 1$. Das Minimum liegt bei $U_{LJ}(r = \sqrt[6]{2}) = -\epsilon$. Das WCA-Potential ist um ϵ nach oben verschoben und bei $r = \sqrt[6]{2}$ abgeschnitten und auf 0 gesetzt.

Das Minimum des Potentialtopfs bei einem Wert von $-\epsilon$ stellt die Nullstelle der Kraft dar und damit denjenigen Abstand, an dem Repulsion und Attraktion im Gleichgewicht sind. Ab diesem Punkt wird bei dem WCA-Potential die Energie auf Null gesetzt, um die Attraktion abzuschalten. Um keine Sprungstelle zu erhalten, wird das LJ-Potential entsprechend um ϵ nach oben verschoben und an der neuen Nullstelle abgeschnitten:

$$U_{WCA}(r) = \begin{cases} 4\epsilon \left[\left(\frac{d}{r} \right)^{12} - \left(\frac{d}{r} \right)^6 \right] + \epsilon, & r \leq \sqrt[6]{2}d \\ 0, & r > \sqrt[6]{2}d \end{cases} \quad (4.3)$$

wobei $r = \sqrt[6]{2}d$ die Reichweite der repulsiven Wechselwirkung darstellt. Auch hier ist das Potential abhängig von der Potentialoptiefe ϵ und dem Durchmesser d der Teilchen. In den hier vorliegenden Simulationen wird der Durchmesser auf $d = 1\sigma$ gesetzt, da er die Längenskala des Systems darstellt. Jede Größe wird in Einheiten des Teilchendurchmessers angegeben.

Im realistischen Fall des Hartkugelsystems würden die einzelnen Kugeln nicht voneinander „spüren“, bis sie einander berühren. Das Potential, sobald sie sich berühren, wäre unendlich, da sie sich nicht überlappen dürfen, entspricht also einer Stufenfunktion. Dadurch ist das Potential nicht differenzierbar und die Kraft auf ein Teilchen würde unendlich ansteigen, sobald es ein anderes berührt. Damit würde im nächsten Integrationschritt das Teilchen unendlich beschleunigt werden. Um dieses unphysikalische Verhalten zu verhindern, sind die Kugeln leicht elastisch, d.h. das Potential erlaubt durch ein kontinuierliches Ansteigen eine leichte Überlappung.

In den hier vorliegenden Simulationen wird die Tiefe des Potentials, welche auch den Anstieg und damit die „Härte“ des Kugelmaterials anzeigt, auf $\varepsilon = 1k_B T$ gesetzt. Die Kugeln sind somit relativ weich. Je höher ε , desto steiler das Potential und somit die Härte der simulierten Kugeln.

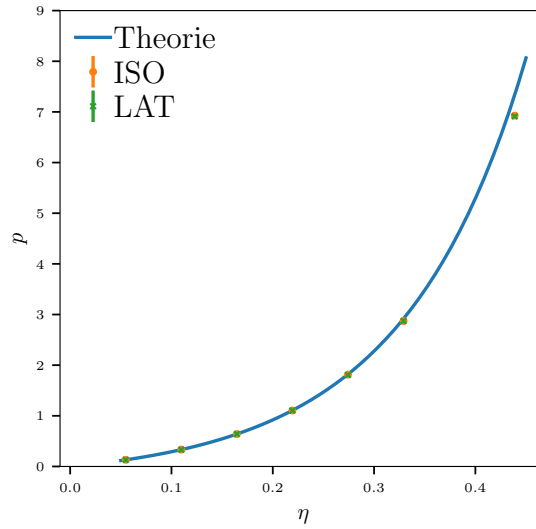


Abb. 4.2: Phasendiagramm der WCA-Kugeln. Hierbei werden die durch das Lowe-Andersen- und das isokinetische Thermostat gekühlten WCA-Kugel-Systeme mit der Zustandsgleichung eines Hartkugelsystems verglichen. Die Datenpunkte liegen auf der in blau eingezeichneten Theoriekurve aus der Carnahan-Starling-Zustandsgleichung [40]. Als Equilibrierungszeit wurde eine MD-Zeit von 100 angenommen, erst danach beginnt die Datennahme über einen Zeitraum von $t_{MD} = 10000$. Bei niedrigen Dichten stimmen die Theoriewerte mit den Simulationen gut überein, bei hohen Dichten weichen die Datenpunkte leicht ab. Die Simulationen werden durch den Barker-Henderson-Radius $d_{BH} \approx 1.01561$ korrigiert (siehe hierzu Abschnitt 4.1.1).

Um zu überprüfen, ob das Potential die Zustandsgleichungen erfüllt, wird das Phasendiagramm der WCA-Kugeln mit dem eines Hartkugelsystems verglichen. Die Scherrate ist dabei $\dot{\gamma} = 0$. Sowohl das Lowe-Andersen-Thermostat als auch das isokinetische Thermostat werden bei den Simulationen verwendet. Die Werte stimmen bei niedrigen Dichten gut miteinander überein. Die Theoriekurve wurde mithilfe der Carnahan-Starling-Gleichung [40] berechnet:

$$p(\eta_P) = \frac{6}{\pi} \eta_P \frac{1 + \eta_P + \eta_P^2 - \eta_P^3}{(1 - \eta_P)^3}. \quad (4.4)$$

Dabei entspricht η_P der Packungsdichte, also dem Verhältnis zwischen von den Kugeln eingenommenen Volumen zum Gesamtvolumen. Die Packungsdichte ist proportional zur Dichte:

$$\eta_P = \frac{N \cdot V_{\text{Kugel}}}{V_{\text{Ges}}} = \rho \cdot V_{\text{Kugel}} . \quad (4.5)$$

Bei hohen Dichten weichen die Simulationsergebnisse von der Theoriekurve ab, was zu erwarten ist. Die Carnahan-Starling-Gleichung beschreibt Systeme mit niedriger Packungsdichte.

4.1.1 Reskalierung

Um die Simulationsergebnisse mit den theoretischen Werten vergleichen zu können, muss der Radius (bzw. der Durchmesser σ) der Kugel reskaliert werden. Das WCA-Potential modelliert Teilchen, deren Verhalten echten harten Kugeln mit einem geringfügig kleineren Radius entspricht, da die Kugeln sich leicht überlappen können. Der effektive Radius einer harten Kugel variiert also von dem eingestellten σ . Um die beiden Ergebnisse miteinander vergleichen zu können, wird davon ausgegangen, dass das Potential eine Störung auf dem idealen Kastenpotential der harten Kugeln darstellt, siehe hierzu auch [41, 42]. Mithilfe der Gleichung

$$d = \int_0^\infty 1 - e^{-\frac{u_{\text{WCA}}(r)}{k_B T}} \quad (4.6)$$

lässt sich ein „tatsächlicher“ Hartkugelradius bestimmen, mit dem die Ergebnisse reskaliert werden müssen. Da ab $r = \sqrt[6]{2}$ das Potential auf 0 gesetzt wird, können beim Integral diese Beiträge vernachlässigt werden, die obere Grenze des Integrals kann also auf $\sqrt[6]{2}$ gesetzt werden. Durch Integration erhält man für $k_B T = 1$ einen Wert von

$$d_{\text{BH}} \approx 1.01561 \quad (4.7)$$

für den sogenannten Barker-Henderson-Radius. Dies ist der korrespondierende Radius einer harten Kugel in Einheiten von σ und entsprechend der Umrechnungsfaktor. Mit diesem muss die Packungsdichte modifiziert werden:

$$\eta'_P = \frac{4}{3} \pi \left(\frac{d_{\text{BH}}}{2} \right)^3 \cdot \rho . \quad (4.8)$$

Ähnlich wird der Druck modifiziert. Die Einheiten des Drucks sind

$$[p] = \frac{[E]}{[m]^3} , \quad (4.9)$$

$$p' = p \cdot \frac{d_{\text{BH}}^3}{k_B T} . \quad (4.10)$$

Um den dimensionslosen und damit mit den Theoriewerten vergleichbaren Druck zu erhalten, muss also mit dem kubischen Barker-Henderson-Radius d_{BH}^3 multipliziert und durch die in der Simulation verwendete Temperatur dividiert werden. Sofern nicht anders angegeben, ist die Temperatur in den nachfolgenden Simulationen $\varepsilon = k_B T = 1$. Die Korrektur des Drucks wird im Nachfolgenden nur vorgenommen, wenn tatsächlich mit anderen, unabhängigen Simulationen verglichen wird und ist entsprechend vermerkt.

4.1.2 Abbilden des Modells

In diesem Abschnitt wird auf die Abbildung der in der Simulation verwendeten Teilchen auf realistische Teilchen eingegangen. Dieser Abschnitt wurde auch in [2] behandelt und dient der Vollständigkeit der theoretischen Beschreibung des Systems.

Dazu wird zunächst angenommen, dass es sich bei den verwendeten Simulationsteilchen um Kolloide handelt. Die typische Größenordnung eines kolloidalen Teilchens liegt auf der Mesoskala zwischen 100 nm und 10 μm . Für die folgenden Rechnungen wird eine Größe von 1 μm angenommen. Damit ist dies die in den Simulationen verwendete typische Längeneinheit und der Durchmesser der Teilchen $\sigma = 10^{-6} \text{ m}$.

Zur Bestimmung der Zeitskala wird die Simulationszeit t_{MD} herangezogen. Bei einem gewählten Zeitschritt von $\Delta t = 10^{-4}$ in Simulationseinheiten entspricht diese 10^4 Zeitschritten. Zwischen Zeit- und Längeneinheit gilt die Relation:

$$t_{\text{MD}} = \sigma \cdot \sqrt{\frac{m}{\varepsilon}}, \quad (4.11)$$

Dabei ist ε zum einen die Tiefe des dem WCA-Potentials zugrundeliegenden Lennard-Jones-Potentials, zum anderen handelt es sich dabei um ein Maß für die thermische Energie des Systems. m ist die Masse eines Kolloidteilchens. In diesen Betrachtungen wird Polystyrol als Material verwendet. Es wird angenommen, dass die Simulationen bei Raumtemperatur stattfinden. Entsprechend gilt:

$$\begin{aligned} \varepsilon &= 1 k_B T, \\ &= 1.3 \cdot 10^{-23} \frac{\text{J}}{\text{K}} \cdot 300 \text{ K}. \end{aligned} \quad (4.12)$$

Die Masse der Polystyrolkugeln kann über die Dichte von Polystyrol $\rho = 1.05 \frac{\text{t}}{\text{m}^3}$ bestimmt werden. Durch Einsetzen in die Formel der Masse einer Kugel erhält man folgenden Wert für t_{MD} :

$$\begin{aligned} m &= \frac{4}{3} \pi \left(\frac{\sigma}{2} \right)^3 \cdot \rho \\ &= \frac{4}{3} \pi (10^{-6} \text{ m})^3 \cdot \frac{1}{8} \cdot 1.05 \frac{\text{t}}{\text{m}^3} \\ &= 0.55 \cdot 10^{-18} \text{ t} = 5.5 \cdot 10^{-16} \text{ kg}, \end{aligned} \quad (4.13)$$

$$\begin{aligned} \Rightarrow t_{\text{MD}} &= 10^{-6} \text{ m} \sqrt{\frac{5.5 \cdot 10^{-16} \text{ kgs}^2}{3.9 \cdot 10^{-21} \text{ kgm}^2}} \\ &= 3.755 \cdot 10^{-4} \text{ s}. \end{aligned} \quad (4.14)$$

Dabei entspricht t_{MD} der strukturellen Relaxationszeit, die eine Polystyrolkugel benötigt, um sich den gegebenen Bedingungen anzupassen. Gleichzeitig benötigt eine Kugel durchschnittlich eine Zeit von t_{MD} , um einen Abstand von σ hinter sich zu bringen.

Diese beiden Werte hängen mit der Diffusion zusammen, welche ebenfalls ein Maß dafür ist, wie lange ein Teilchen benötigt, um sich um eine Strecke seines eigenen Durchmessers σ fortzubewegen. Eine andere Methode, die Relaxationszeit zu bestimmen, verwendet entsprechend die Diffusion der Teilchen. Durch experimentelle Bestimmung der Diffusionskonstante D_0 ergibt sich:

$$t = \frac{\sigma^2}{6D} \approx 10^{-2} - 10^{-4} \text{ s} \quad (4.15)$$

in Experimenten, die mit der Simulation vergleichbar sind, wie in [43] durchgeführt. Die Zeitskala der Simulationen von kolloidalen Teilchen entspricht also in etwa der experimentellen Zeitskala.

Zum Vergleich soll an dieser Stelle die Rechnung für ein atomistisches System wiederholt werden. Als Beispiel dient Argon, welches durch ein Lennard-Jones-Potential beschrieben werden kann. Die Potentialtiefe ist in diesem Fall $\varepsilon = 1.65 \cdot 10^{-21}$ J. Der Durchmesser von Argon entspricht $\sigma = 3.4 \cdot 10^{-10}$ m, die Dichte $\rho = 1.784 \text{ kgm}^{-3}$. Diese Werte können nun in die obige Formel (4.11) eingesetzt werden und man erhält:

$$t = \sqrt{\frac{m\sigma^2}{\varepsilon}} = 2.17 \cdot 10^{-12} \text{ s} . \quad (4.16)$$

Die physikalischen Zeitskalen hängen also stark von dem System ab, das durch die verwendeten Potentiale beschrieben wird.

4.2 POLYMERKETTEN

Zur Analyse von Polymeren ist ein an die Problemstellung angepasstes, zugrundeliegendes Modell sinnvoll. Da das Polymer stark vergrößert wird, um Rechenzeit einzusparen, verliert man Informationen über die Beschaffenheit. So kann man das Polymer beispielsweise beim Asakura-Osawa-Modell [44, 45] als Kugel auffassen. Im Folgenden wird ein wesentlich detaillierteres Modell verwendet, bei dem mehrere Atome zu einer WCA-Kugel zusammengefasst werden.

Ein Standardmodell zur Simulation von vergrößerten Polymeren ist das Kremer-Grest-Modell [17, 46]. Die Modell-Monomere, die je nach Vergrößerung für ein oder mehrere reale Monomere des Polymers stehen können, werden mithilfe von nicht-linearen Federn miteinander verbunden. Die Federn werden in dieser Arbeit mit dem sogenannten FENE-Potential (FENE= finitely extensible nonlinear elastic) modelliert:

$$U_{\text{FENE}} = \begin{cases} -0.5 k R_0^2 \ln \left[1 - \left(\frac{r_{ij}}{R_0} \right)^2 \right] , & r_{ij} < R_0 \\ \infty , & r_{ij} \geq R_0 \end{cases} , \quad (4.17)$$

wobei R_0 die maximale Auslenkung der Federn ist (in unserem Fall ist $R_0 = 1.5 \sigma$) und r_{ij} der Abstand von zwei aufeinander folgenden Kettenmonomere i und j . k ist dabei die Federkonstante und in dieser Arbeit wird $k = 30 \frac{\varepsilon}{\sigma^2}$ gewählt.

Aufgrund der Abstoßung des WCA-Potentials (welches zusätzlich zum FENE-Potential zwischen zwei benachbarten Monomeren wirkt), können sich zwei aufeinanderfolgende Kugeln nicht zu sehr annähern. Wenn die Kugeln ihren Abstand zu stark ausdehnen, ziehen die Federn sie wieder zurück. Bei einer übermäßigen Streckung der Federn würden diese zerbrechen, dargestellt durch das unendliche Potential bei einem Abstand $r > R_0$. Das Auseinanderbrechen der Ketten kann mit dem zugrundeliegenden Programmcode nicht realisiert werden. Eine vereinfachte Darstellung des vereinfachten Modells ist in Abbildung 4.3 zu sehen. Durch die Federn ist die Überlappung zweier benachbarter Teilchen wesentlich höher. Auf diese Art können höhere Monomerdichten realisiert werden, ohne eine kristalline Phase zu simulieren.

Mit diesem Polymermodell lassen sich Polymerlösungen mit Ketten einer bestimmten Länge unterschiedlicher Dichte simulieren. Um komplexe Strukturen wie Ringpolymere oder Sternmoleküle darzustellen, müssen die einzelnen Monomere durch die Federn zu einem entsprechend komplexen Konstrukt aufgebaut werden.

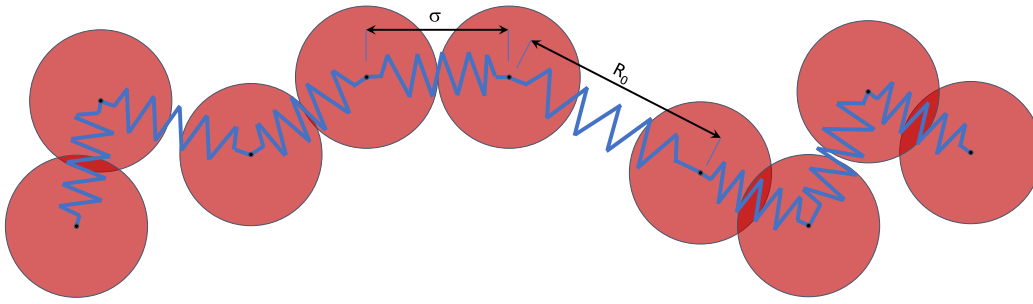


Abb. 4.3: Kremer-Grest-Modell mit neun Monomeren, die über eine FENE-Feder verbunden sind. Die einzelnen Kugeln stoßen sich wie im zuvor beschriebenen Modell wie harte Kugeln ab. Zwei aufeinanderfolgende Monomere sind durch eine (in blau dargestellte) Feder verbunden. Diese kann sich bis zu einem Maximalwert von R_0 ausdehnen, wie im Bild angedeutet. Der ideale Abstand zweier aufeinanderfolgender Kugeln ist dabei bei $r = \sigma$. Sind zwei Kugeln zu nah beieinander, werden sie durch das WCA-Potential abgestoßen.

Die grundlegende Idee hinter der Simulation von Polymerketten für die der Arbeit zugrundeliegenden Hybridsimulationen ist die, dass es bei starker Scherung zu einer Ausdünnung der Lösung kommt, dem sogenannten *Shear Thinning*. Die Viskosität der Flüssigkeit sinkt dabei im Vergleich zur Viskosität einer Newtonschen Flüssigkeit, die konstant bleibt. Die Viskosität ist ein Maß für die Reaktion einer Flüssigkeit auf eine Krafteinwirkung. Es handelt sich dabei um eine dynamische Größe, da sie von der Stärke der Scherung abhängen kann.

Wichtige Parameter des FENE-Potentials sind die maximale Auslenkung R_0 und die Federkonstante k . Simulationsergebnisse mit variierender Federkonstante k sind im Anhang, Abschnitt A.2.2 zu finden. In dieser Arbeit wird mit $k = 30 \frac{\epsilon}{\sigma^2}$ gearbeitet.

In dieser Arbeit werden Polymerketten mit $N_p = 2, 5, 10$ und 15 zusammenhängenden Monomeren simuliert. Bei Ketten kann nicht mehr von der Teilchendichte im zuvor verwendeten Zusammenhang gesprochen werden, da jede einzelne Kette als ein Polymer, also ein Teilchen gilt. Stattdessen wird ρ nun als Monomerdichte behandelt, bezogen auf die Anzahl an einzelnen Monomeren, die sich im simulierten Volumen befinden. Alternativ kann man zum Vergleich zwischen Polymerketten und einzelnen Kugeln die jeweilige Packungsdichte η_p angeben.

Auch hier können die einzelnen Monomere wie im vorherigen Abschnitt 4.1.2 beschrieben auf physikalische Systeme abgebildet werden.

Die Ergebnisse der Analyse der Polymerkettensysteme sind in Kapitel 7 vorgestellt.

Bei der numerischen Strömungsmechanik (englisch: *Computational Fluid Dynamics*, CFD [9]) werden mithilfe von numerischen Methoden die Navier-Stokes-Gleichungen gelöst, die die Dynamik von Fluiden beschreiben. Dabei können verschiedene Verfahren als Lösungsmethoden verwendet werden, z. B. die Finite-Elemente-Methode oder die in dieser Arbeit verwendete diskontinuierliche Galerkin-Methode.

Das nachfolgende Kapitel liefert eine kurze Einführung in die zu lösenden Navier-Stokes-Gleichungen und den in [1, 2] vorgestellten Ergebnissen zugrundeliegenden CFD-Simulationen.

5.1 HERLEITUNG

Um eine (inkompressible) Flüssigkeit zu beschreiben, sollen in den nachfolgenden Unterkapiteln die Bewegungsgleichungen eines Volumenelementes δV in einem Strömungsprofil betrachtet werden. Dazu wird zunächst die Kontinuitätsgleichung und anschließend die Impulsgleichung hergeleitet. Die Herleitung lehnt sich dabei an die Beschreibung aus [47] an. Aus diesen Gleichungen werden die Navier-Stokes-Gleichungen abgeleitet.

Das Integrationsschema, das im Rahmen der von Dr. Nehzat Emamy und Dr. Leonid Yelash entwickelten CFD-Simulation mithilfe des in Darmstadt entwickelten BoSSS-Frameworks [48, 49] zum Einsatz kommt, wird im Anschluss beschrieben.

5.1.1 Die Kontinuitätsgleichung

Aus der Massenerhaltung lässt sich die Kontinuitätsgleichung herleiten. Sie verknüpft die zeitliche Änderung der Dichte ρ einer Flüssigkeit in einem Volumenelement mit der räumlichen Änderung der Massenstromdichte $\mathbf{j} = \rho \mathbf{u}$, wobei \mathbf{u} die Geschwindigkeit eines Volumenelementes ist. Anschaulich bedeutet die Kontinuitätsgleichung, dass aus einem Volumen δV ebenso viel Masse herausfließt, wie eindringt. Die Gleichung soll in diesem Abschnitt im Detail betrachtet werden.

Ein Volumenelement δV einer inkompressiblen Flüssigkeit bewege sich mit der Strömungsgeschwindigkeit \mathbf{u} . Dabei herrscht ein kontinuierlicher Massenstrom mit der Massenstromdichte $\mathbf{j} = \rho \mathbf{u}$, wobei ρ die (konstante) Dichte der Flüssigkeit sei.

Zur Zeit t entspricht die Masse M des Volumens V also

$$M = \int_V \rho dV. \quad (5.1)$$

Durch ein infinitesimales Oberflächenelement $d\mathbf{A}$ fließt pro Zeiteinheit die Flüssigkeitsmenge $\rho \mathbf{u} d\mathbf{A}$. Dabei entspricht $d\mathbf{A}$ dem Betrag dieses infinitesimalen Oberflächenelements

entlang der äußeren Normalen \hat{n} . Unter Zuhilfenahme des Gaußschen Integralsatzes ergibt sich damit:

$$\begin{aligned}\frac{d}{dt}m(t) &= \frac{d}{dt} \int_V \rho dV = - \int_{\partial V} \rho \mathbf{u} \cdot \hat{n} dA, \\ &= - \int_V \nabla \cdot (\rho \mathbf{u}) dV, \\ \Rightarrow \quad 0 &= \frac{d}{dt} \int_V \rho dV + \int_V \nabla \cdot (\rho \mathbf{u}) dV.\end{aligned}\quad (5.2)$$

Ist ρ hinreichend glatt, kann man über den Satz von Lebesgue die beiden Integrale zu einem zusammenfassen. Da dieser Zusammenhang für beliebige Kontrollvolumina gilt, muss der Integrand gleich Null sein. Damit ergibt sich die Kontinuitätsgleichung zu

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0. \quad (5.3)$$

Unter Verwendung der totalen Zeitableitung (Gleichung (5.8)) im ersten Term und Anwendung der Kettenregel im zweiten Term ergibt sich daraus:

$$\frac{d\rho}{dt} + \rho \nabla \cdot \mathbf{u} = 0. \quad (5.4)$$

Für inkompressible Flüssigkeiten gilt $\rho(t, \mathbf{r}) = \text{konst}$ und somit vereinfacht sich die Kontinuitätsgleichung für diesen Fall zu

$$\nabla \cdot \mathbf{u} = 0. \quad (5.5)$$

5.1.2 Die Impulsgleichungen

In diesem Unterkapitel werden die Bewegungsgleichungen eines Volumenelementes δV einer (inkompressiblen) Flüssigkeit oder eines Gases betrachtet, dass sich in einem Strömungsprofil bewegt. Dieses Volumenelement hat dabei verschiedene Eigenschaften: Zum Zeitpunkt t befindet es sich am Ort $\mathbf{r}(t)$ mit einer Strömungsgeschwindigkeit

$$\mathbf{u}(\mathbf{r}, t) = \frac{d\mathbf{r}}{dt}. \quad (5.6)$$

Die Masse des Volumenelementes ist dabei $\delta m = \rho \delta V$.

Für eine feste Zeit t beschreibt $\mathbf{u}(\mathbf{r}, t)$ das Strömungsfeld. Wenn dieses zeitunabhängig ist, also $\mathbf{u}(\mathbf{r}, t) = \mathbf{u}(\mathbf{r})$, handelt es sich um ein stationäres Strömungsfeld. Sind dabei die Reibungskräfte vernachlässigbar, bezeichnet man dies als eine ideale Flüssigkeit. Ist die Reibungskraft hingegen groß im Vergleich zu anderen Kräften, handelt es sich um eine viskose oder zähe Flüssigkeit. Ist das Strömungsfeld wirbelfrei, findet also keine Impulsübertragung orthogonal zur Strömung statt, spricht man von laminarer Strömung. Die Bewegung der gesamten Flüssigkeit ist bekannt, wenn $\mathbf{u}(\mathbf{r}, t)$ an jedem Ort \mathbf{r} und zu jeder Zeit t gegeben ist.

Nach einem infinitesimalen Zeitschritt dt hat sich die Strömungsgeschwindigkeit des Volumenelementes δV wie folgt verändert:

$$\mathbf{u} + d\mathbf{u} = \mathbf{u}(\mathbf{r} + \mathbf{u} dt, t + dt). \quad (5.7)$$

Zu $d\mathbf{u}$ zählen dabei sowohl zeitliche Änderungen $\frac{\partial \mathbf{u}}{\partial t}$, die lokale Änderung genannt wird, wie auch räumliche $\frac{\partial \mathbf{u}}{\partial \mathbf{r}} \frac{\partial \mathbf{r}}{\partial t}$, also die Änderung der Strömung, wenn das Volumenelement

einen anderen Ort erreicht hat. Diese räumliche Änderung wird auch als Konvektionsbeschleunigung bezeichnet und lässt sich umschreiben zu $\frac{\partial \mathbf{u}}{\partial \mathbf{r}} \frac{\partial \mathbf{r}}{\partial t} = (\mathbf{u} \cdot \nabla) \mathbf{u}$, insgesamt erhält man:

$$\frac{d\mathbf{u}}{dt} = \frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} . \quad (5.8)$$

Die Konvektionsbeschleunigung ist dabei das Skalarprodukt zwischen dem Vektor \mathbf{u} und dem Tensor

$$\nabla \mathbf{u} = \begin{pmatrix} \frac{\partial u_x}{\partial x} & \frac{\partial u_x}{\partial y} & \frac{\partial u_x}{\partial z} \\ \frac{\partial u_y}{\partial x} & \frac{\partial u_y}{\partial y} & \frac{\partial u_y}{\partial z} \\ \frac{\partial u_z}{\partial x} & \frac{\partial u_z}{\partial y} & \frac{\partial u_z}{\partial z} \end{pmatrix} . \quad (5.9)$$

Um die Bewegung der Flüssigkeit, also die Strömungsgeschwindigkeit, zu beschreiben, müssen alle auf das Volumenelement wirkenden Kräfte berücksichtigt werden. Dazu zählen:

1. Die Druckdifferenzen des isotropen Drucks p an verschiedenen Positionen \mathbf{r} in Form der Änderung des Drucks ∇p im Volumenelement

$$\mathbf{F}_P = -\nabla p \delta V . \quad (5.10)$$

2. Die Schwerkraft $\mathbf{F}_g = \delta m \mathbf{g} = \rho \delta V \mathbf{g}$, später wird mit der speziellen Volumenkraft \mathbf{f} gearbeitet, die dieser äußeren Kraft pro Volumen entspricht, also $\mathbf{f} = \mathbf{F}_g / \delta V$.
3. Außerdem sorgen die durch Zähigkeit des Fluids erzeugten irreversiblen Prozesse für innere Spannungen, wie z. B. die inneren Reibungen zwischen zwei sich unterschiedlich schnell bewegenden Schichten. Diese Prozesse werden durch den zähen Spannungstensor $\boldsymbol{\tau}$ beschrieben:

$$\mathbf{F}_R = \nabla \cdot \boldsymbol{\tau} \delta V . \quad (5.11)$$

Insgesamt wirkt auf das Volumenelement δV also eine resultierende Kraft von:

$$\mathbf{F}_{\text{Ges}} = \mathbf{F}_P + \mathbf{F}_g + \mathbf{F}_R = \delta m \ddot{\mathbf{r}} = \rho \delta V \frac{d\mathbf{u}}{dt} . \quad (5.12)$$

Durch Einsetzen der Beschleunigung aus Gleichung (5.8) und den Ausdrücken für die Kräfte erhält man folgenden Satz an partiellen Differentialgleichungen:

$$\rho \left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} \right) = -\nabla p + \nabla \cdot \boldsymbol{\tau} + \mathbf{f} . \quad (5.13)$$

Mit einem allgemeinen Spannungstensor $\boldsymbol{\sigma}$ der Form $\boldsymbol{\sigma} = -p\mathbf{I} + \boldsymbol{\tau}$ lassen sich die Gleichungen zu den Impulsgleichungen für Strömungen mit konstanter Dichte zusammenfassen:

$$\rho \left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} \right) = \nabla \cdot \boldsymbol{\sigma} + \mathbf{f} . \quad (5.14)$$

5.1.3 Die Navier-Stokes-Gleichungen

In der makroskopischen Ebene der Simulationen werden inkompressible Flüssigkeiten betrachtet. Diese werden durch die Kontinuitäts- und die Impulsgleichungen beschrieben, die in den vorherigen Abschnitten 5.1.1 und 5.1.2 hergeleitet wurden. In diesem Abschnitt sollen die Gleichungen entdimensionalisiert werden.

Die beiden Gleichungen lauten wie folgt:

$$\begin{aligned}\rho \left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} \right) &= \nabla \cdot \boldsymbol{\sigma} + \mathbf{f} , \\ \nabla \cdot \mathbf{u} &= 0 .\end{aligned}\tag{5.15}$$

Dabei ist \mathbf{u} der Geschwindigkeitsvektor und $\boldsymbol{\sigma}$ der Cauchy-Stresstensor. Äußere Kräfte werden mit \mathbf{f} zusammengefasst. Dabei kann es sich um jedwede von außen angelegte Kraft handeln, wie z. B. die Gravitationskraft oder auch elektromagnetische Kräfte. Mithilfe des symmetrischen Verzerrungsgeschwindigkeitstensors \mathbf{E} lässt sich der Stresstensor für Newtonsche Fluide umschreiben zu:

$$\boldsymbol{\tau} = \mu (\nabla \mathbf{u} + \nabla \mathbf{u}^T) = 2\mu \mathbf{E} .\tag{5.16}$$

Damit und über die zuvor schon verwendete Relation aus Gleichung (2.35) können die Gleichungen umgeschrieben werden. Dadurch erhält man die Navier-Stokes-Gleichungen:

$$\begin{aligned}\rho \left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} \right) &= -\nabla p + \mu \Delta \mathbf{u} + \mathbf{f} , \\ \nabla \cdot \mathbf{u} &= 0 .\end{aligned}\tag{5.17}$$

Um Fluide mit ähnlichem Strömungsverhalten allgemein und unabhängig von ihrer Skala behandeln zu können, werden dimensionslose Größen eingeführt, vgl. hierzu [50]. Dafür werden zunächst charakteristische Größen eingeführt, die mit einer Tilde gekennzeichnet sind. Dazu zählen

- die charakteristische Länge \tilde{L} , die als Simulationsboxgröße oder als Teilchengröße definiert sein kann,
- die charakteristische Geschwindigkeit \tilde{u} , z. B. der Betrag der mittleren Flussgeschwindigkeit,
- die charakteristische Dichte $\tilde{\rho}$ und
- die charakteristische dynamische Viskosität $\tilde{\mu}$ der Strömung.

Mithilfe dieser Größen lassen sich die Simulationsgrößen reskalieren und normieren. Zusätzlich wird die Reynolds-Zahl Re eingeführt. Dabei handelt es sich um das Verhältnis von spezifischer Impulskonvention zur Impulsdiffusion in einem physikalischen System und es gilt:

$$Re = \frac{\tilde{\rho} \tilde{u} \tilde{L}}{\tilde{\mu}} .\tag{5.18}$$

Damit und mit den dimensionslosen Variablen

- Geschwindigkeitsvektor $\mathbf{u}^* = \frac{\mathbf{u}}{\tilde{u}}$,
- Ortsvektor $\mathbf{r}^* = \frac{\mathbf{r}}{\tilde{L}}$,
- Zeit $t^* = \frac{\tilde{u} t}{\tilde{L}}$,
- Druck $p^* = \frac{p}{\tilde{\rho} \tilde{u}^2}$,
- Dichte $\rho^* = \frac{\rho}{\tilde{\rho}}$,

- dynamische Viskosität $\mu^* = \frac{\mu}{\tilde{\mu}}$ und
- Spannungs- bzw. Stresstensor $\tau^* = \frac{\tau \tilde{\mu}}{\tilde{\mu}}$

sowie den Operatoren $\nabla^* = \tilde{\mathbb{L}} \nabla$ und $\Delta^* = \tilde{\mathbb{L}}^2 \Delta$ werden die Gleichungen (5.13) modifiziert zu:

$$\begin{aligned} \rho^* \left(\frac{\partial \mathbf{u}^*}{\partial t^*} + (\mathbf{u}^* \cdot \nabla^*) \mathbf{u}^* \right) &= -\nabla^* p^* + \frac{\mu^*}{\text{Re}} \Delta^* \mathbf{u}^* + \mathbf{f}^* , \\ \nabla^* \cdot \mathbf{u}^* &= 0 . \end{aligned} \quad (5.19)$$

Um die Gleichungen übersichtlicher zu gestalten, wird im folgenden $\{\mathbf{u}, \mathbf{r}, t, \rho, p, \mu, \nabla, \Delta\}$ geschrieben statt $\{\mathbf{u}^*, \mathbf{r}^*, t^*, \rho^*, p^*, \mu^*, \nabla^*, \Delta^*\}$. Statt \mathbf{f}^* wird die Notation \mathbf{g} verwendet. In \mathbf{g} werden alle äußeren Kräfte, wie Gravitationskraft, zusammengefasst. Mit dieser Notation erhalten die Navier-Stokes-Gleichungen die endgültige Form:

$$\begin{aligned} \rho \left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} \right) &= -\nabla p + \frac{\mu}{\text{Re}} \Delta \mathbf{u} + \mathbf{g} , \\ \nabla \cdot \mathbf{u} &= 0 . \end{aligned} \quad (5.20)$$

5.1.4 Verwendetes Integrationsschema

Bei der Zeitintegration der Kontinuitätsgleichung und der Impulsgleichungen aus Gleichung (5.20) wird die nachfolgend erklärte Projektionsmethode verwendet. In dieser wird durch eine mehrschrittige Projektion die Randbedingung aus Gleichung (5.5) direkt an die Gleichungen gekoppelt, wie schon bei [51], welche aber zu einem „Splitting error“ (deutsch: Unterteilungsfehler) der Größenordnung $\mathcal{O}(\Delta t)$ führte. Verschiedene Methoden wurden entwickelt, um dieses Verfahren zu verbessern und eine höhere Genauigkeit zu erzielen. In dieser Arbeit wird [52] weiterentwickelt und die auch schon in [53] erfolgreich angewendete Methode verwendet. Die Erklärung dieses Abschnitts folgt dabei [2].

Um die entdimensionalisierten Gleichungen aus (5.20) in einem Simulationsgebiet Ω mit Rand $\partial\Omega$ zu lösen, wird eine mehrschrittige Projektionsmethode angewendet, die sog. Chorin-Projektionsmethode. Die Superskripte n und $n+1$ in den Gleichungen stehen dabei für die jeweilige Zeitebene der entsprechend gekennzeichneten Größe.

Der erste Schritt dieses Verfahrens ist die Zeitintegration der zuvor hergeleiteten Navier-Stokes-Gleichungen mithilfe der Euler-Methode. Vom n -ten Zeitschritt ausgehend erhält man einen Hilfstern \mathbf{u}' , der nach Euler-Methode dem $n+1$ -ten Zeitschritt entsprechen würde. Im zweiten Schritt der Methode wird die Änderung des Durchschnittsdrucks \bar{p} entlang der Normalen $\hat{\mathbf{n}}$ in der Zeitebene $n+1$ berechnet, aus der ein Korrekturterm ermittelt wird, um den der vorherige Term \mathbf{u}' vermindert wird. Dieser Korrekturterm folgt als „Strafterm“ aus dem numerischen Verfahren. Im dritten Schritt des Verfahrens wird schließlich die Zeitebene $n+1$ für die Geschwindigkeit \mathbf{u} berechnet:

$$\text{I} \quad \mathbf{u}' = \Delta t \left(-\mathbf{u}^n \cdot \nabla \mathbf{u}^n + \frac{1}{\text{Re}} \nabla \cdot \frac{\boldsymbol{\sigma}^n}{\rho} + \frac{\mathbf{g}^n}{\rho} \right), \quad \text{in } \Omega \quad (5.21a)$$

$$\text{II} \quad \Delta \bar{p}^{n+1} = \frac{\nabla \cdot \mathbf{u}'}{\Delta t}, \quad \text{in } \Omega \quad (5.21b)$$

$$\frac{\partial \bar{p}^{n+1}}{\partial \hat{n}} = \frac{\hat{n} \cdot \mathbf{u}'}{\Delta t}, \quad \text{auf } \partial \Omega \quad (5.21c)$$

$$\mathbf{u}^* = \mathbf{u}' - \Delta t \nabla \bar{p}^{n+1}, \quad \text{in } \Omega \quad (5.21d)$$

$$\text{III} \quad \mathbf{u}^{n+1} = \mathbf{u}^n + \mathbf{u}^*, \quad \text{in } \Omega. \quad (5.21e)$$

Druck p und Stresstensor $\boldsymbol{\tau}$ sind wieder zum Spannungstensor $\boldsymbol{\sigma}$ zusammengefasst. Die Ableitung in Normalenrichtung aus Gleichung (5.21c) ist anders ausgedrückt auch

$$\frac{\partial \bar{p}^{n+1}}{\partial \hat{n}} = \hat{n} \cdot \nabla \bar{p}^{n+1}. \quad (5.22)$$

Für den Zeitschritt $n + 1$ gilt dabei für den durchschnittlichen Druck \bar{p} :

$$\bar{p}^{n+1} = \frac{1}{\Delta t} \int_{t^n}^{t^{n+1}} \frac{p'}{\rho} dt. \quad (5.23)$$

Dabei muss das Integral des Drucks über das Gebiet Ω verschwinden, also

$$\int_{\Omega} \bar{p}^{n+1} = 0, \quad (5.24)$$

um eine eindeutige Lösung der Gleichungen (5.21b)-(5.21c) zu gewährleisten. An dieser Stelle sei angemerkt, dass p' eine Druckkorrektur ist, um divergenzfreie Bedingungen für die Geschwindigkeit zu sichern. Der Druck p ist dabei in den Gleichungen wie oben angemerkt in Form des Spannungstensors $\boldsymbol{\sigma}$ enthalten, siehe dazu auch Gleichung (2.35).

Die zweite Ordnung des Integrationsschritts der Geschwindigkeit \mathbf{u} wird mithilfe der Adams-Bashforth-Methode [54] im Schritt I durchgeführt. Dabei sind die Koeffizienten $\beta_0 = 3/2$ und $\beta_1 = -1/2$ gewählt:

$$\mathbf{u}'_{AB} = \Delta t \sum_{j=0}^{J-1} \beta_j \left(-\mathbf{u}^{n-j} \cdot \nabla \mathbf{u}^{n-j} + \frac{1}{\text{Re}} \nabla \cdot \frac{\boldsymbol{\sigma}^{n-j}}{\rho} + \frac{\mathbf{g}^{n-j}}{\rho} \right). \quad (5.25)$$

Dennoch ist der effektive Druck \bar{p}^{n+1} bereits mit der ersten Ordnung in der Zeit gut bestimmt. Für eine Genauigkeit in höherer Ordnung des Drucks kann in [55, 56] nachgelesen werden.

Im oben beschriebenen Projektionsschema verschwindet die Ableitung $\nabla \cdot \mathbf{u}^{n-j} = 0$ für alle $j \geq 0$. Entsprechend können alle Terme im Integrationsschritt III integriert werden. Durch Ersetzen der mittleren Geschwindigkeit aus dem ersten Schritt wird die rechte Seite der Poisson-Gleichung im Schritt II zeitunabhängig. Auf diese Weise wird eine numerische Instabilität verhindert, wie sie etwa in [57, 58, 59] unter Verwendung der diskontinuierlichen Galerkin-Methode beobachtet wurde.

5.2 DISKONTINUIERLICHE GALERKIN-METHODE

Die Galerkin-Methode ist ein wichtiges Instrument der Numerik. Sie dient als Methode zur Lösung von Differentialgleichungen, denen kein natürliches Extremalprinzip zugrunde liegt. Sie ist nicht nur Grundlage der hier verwendeten diskontinuierlichen Galerkin-Methode, sondern auch der Finite-Elemente-Methode und dehnt deren Anwendbarkeit auf Kontinuumsprobleme aus, wie z. B. Strömungsberechnungen, die die Basis der dieser Arbeit zugrundeliegenden Problemstellung bildet.

Bei der Galerkin-Methode wird das kontinuierliche Problem auf einen geeigneten diskreten Raum projiziert. Dabei erhält man ein System für die Koeffizienten der diskreten Lösung. Nachfolgend soll die Vorgehensweise der Galerkin-Methode erklärt werden, um anschließend eine schwache Formulierung der Ortsdiskretisierung der der CFD-Simulation zugrundeliegenden Gleichungen in der diskontinuierlichen Galerkin-Methode zu erhalten.

GALERKIN-METHODE Zur Herleitung der Galerkin-Methode betrachtet man die folgende Differenzialgleichung für $u(x)$:

$$\mathbf{D}(u(x)) + f(x) = 0, \quad (5.26)$$

mit Differenzialoperator \mathbf{D} und vorgegebener Funktion $f(x)$. Außerdem erfüllt u die vorgegebenen Anfangsbedingungen $u(x_0) = u_0$. Als Lösung für u wird nun eine Linearkombination von Basisfunktionen Φ_i aus dem Funktionenraum V aufgestellt. Da es sich dabei nur um eine Näherung handelt, wird sie als u_N bezeichnet:

$$u_N(x) = \sum_i^N c_i \Phi_i(x), \quad (5.27)$$

wobei c_i die zu bestimmenden Koeffizienten sind. Eingesetzt in obige Differenzialgleichung (5.26) kann diese Approximation um ein Residuum $R(x)$ abweichen:

$$\mathbf{D}(u_N(x)) + f(x) = R(x). \quad (5.28)$$

Dieses Residuum kann minimiert werden, wodurch ein Minimierungsproblem geschaffen wird. Nun wird eine Lösung gesucht, für die

$$\langle w, r \rangle = \int w(x) R(x) dx = 0 \quad (5.29)$$

für beliebige Testfunktionen $w(x)$ aus dem diskreten Raum mit den Basisfunktionen $\{\Phi_1, \dots, \Phi_N\}$.

Speziell bei der Galerkin-Methode ist, dass die Gewichtsfunktionen (Testfunktionen) die Basisfunktionen Φ_j sind, sodass sich für c_j folgendes Gleichungssystem ergibt:

$$\left\langle \Phi_j, \mathbf{D} \left(\sum_{i=1}^N c_i \Phi_i \right) + f \right\rangle = 0, \quad j = 1, \dots, N. \quad (5.30)$$

In dem der Arbeit zugrundeliegenden Problem handelt es sich um ein mehrdimensionales Problem, sodass u nicht nur für x , sondern auch für y und (im dreidimensionalen Fall) auch für z gelöst werden muss.

5.2.1 Definitionen

Im nachfolgenden Abschnitt werden einige wichtige Definitionen eingeführt, die im Anschluss zur Erläuterung der Ortsdiskretisierung des Gitters, auf dem die Navier-Stokes-Gleichungen gelöst werden, notwendig sind. Die Erklärung folgt dabei [1, 60].

Das Simulationsgebiet Ω der CFD-Simulation wird mithilfe eines Geometrie konformen Gitternetzes \mathcal{T}_h aufgeteilt. Die zugehörige Gitterweite beträgt dabei h .

Jedem Gitterelement T kann auf dem Rand ∂T ein nach außen gerichteter Normaleneinheitsvektor \mathbf{n}_T zugeordnet werden. Gitteroberflächen \mathcal{F}_h können innere Zwischenflächen zwischen benachbarten Flächen \mathcal{F}_h^i oder Randflächen \mathcal{F}_h^b sein. Der Oberflächennormalenvektor \mathbf{n}_F zeigt von einem willkürlich gewählten (aber festen) Element T_1 zu einem Element T_2 . Entsprechend ist $\mathbf{n}_F = \mathbf{n}_{T_1} = -\mathbf{n}_{T_2}$ auf jeder Fläche $\mathcal{F} \in \mathcal{F}_h^i$. An den Rändern zeigt die Flächennormale aus dem Gebiet Ω heraus.

Zur Veranschaulichung sind diese Definitionen noch einmal in Abbildung 5.1 aufgezeigt und markiert.

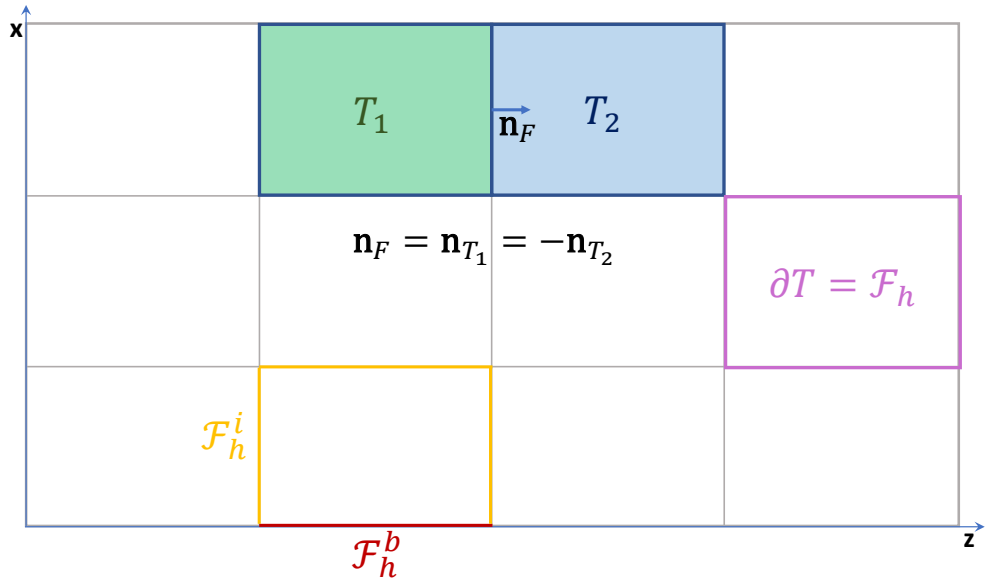


Abb. 5.1: Veranschaulichung der Definition. Die Abbildung stellt das Simulationsgebiet Ω dar. Das eingezeichnete Gitter entspricht dem Gitternetz \mathcal{T} . Zwei der Gitterelemente, T_1 und T_2 wurden in grün, bzw. blau eingefärbt. Der Normaleneinheitsvektor $\mathbf{n}_T = \mathbf{n}_{T_1}$ ist dabei der Einheitsvektor auf der Fläche T_1 und zeigt entgegengesetzt zum Einheitsvektor auf T_2 . Der Rand einer Zelle T entspricht einer Oberfläche \mathcal{F}_h , wie in lila eingezeichnet. \mathcal{F}_h besteht aus inneren Oberflächen (\mathcal{F}_h^i , gelb) und äußeren Randflächen (\mathcal{F}_h^b , rot).

Die Lösungen der Navier-Stokes-Gleichungen werden später für jede Zelle (also jedes Gitterelement) an jedem Quadraturpunkt angenähert. Als Quadraturpunkt bezeichnet man diejenigen Punkte, an denen eine solche Näherung vorgenommen wird. Dabei ist zu beachten, dass die Lösung lediglich innerhalb einer Zelle stetig ist. Entsprechend ist die Gesamtlösung der Geschwindigkeit eine stückweise stetige, polynomiale Funktion aus einem gebrochenen Polynomialraum V_h . Die Anzahl der Variablen entspricht dabei d , wo-

bei d die räumliche Dimension darstellt. Die Polynome haben einen maximalen Grad von k . Mathematisch notiert heißt das:

$$V_h := \mathcal{P}_d^k(\mathcal{T}_h) , \quad (5.31)$$

$$\mathcal{P}_d^k(\mathcal{T}_h) := \{v \in L^2(\Omega) \mid \forall T \in \mathcal{T}_h, v|_T \text{ ist Polynom, } \deg(v|_T) \leq k\} , \quad (5.32)$$

$$N_d^k = \dim(\mathcal{P}_d^k(T)) = \frac{(k+d)!}{k!d!} . \quad (5.33)$$

Da die polynomwertigen Funktionen auf den Rändern der Zelle ∂T nicht definiert sind, ist es notwendig, die Abstände und den Durchschnitt der Funktion bei zwei benachbarten Zellen zu kennen. Diese sind der Durchschnitts- und der Sprungoperator, die durch geschwungene bzw. eckige Klammern dargestellt und in den Umformungen verwendet werden. Die beiden Operatoren einer skalaren Funktion f_h auf einer Oberfläche zweier benachbarter Gitterelemente T_1 und T_2 sind definiert als:

$$[[f_h]] = \frac{1}{2} (f|_{T_1} + f|_{T_2}) , \quad (5.34)$$

$$\{\{f_h\}\} = f|_{T_1} - f|_{T_2} . \quad (5.35)$$

Vektorfunktionen werden komponentenweise behandelt.

Diese Operatoren müssen für die Randfläche \mathcal{F}_h^b definiert werden. In diesem Fall gilt:

$$[[f_h]] = \{\{f_h\}\} = f|_T . \quad (5.36)$$

Diese Bedingung ist immer erfüllt, sofern nichts Gegenteiliges erwähnt wird.

Mithilfe dieser Definitionen lassen sich die im Unterabschnitt 5.1.3 hergeleiteten Navier-Stokes-Gleichungen aus (5.21) diskretisieren. Dazu werden nun Hilfsterte definiert, die der diskreten Form der einzelnen Terme entsprechen. Anschließend wird eine schwache Formulierung der Gleichungen formuliert.

Zur Notation wird (\cdot, \cdot) als $L^2(\Omega)$ -Norm, also das innere Produkt von skalaren und vektorwertigen Funktionen, verwendet. Außerdem gilt die Einstein'sche Summenkonvention.

5.2.2 Schwache Formulierung der Gleichungen

Um die im Ort diskretisierte Form der Endgleichungen zu erhalten, werden in [1] die einzelnen Termteile diskretisiert und vereinfacht. Anschließend wird eine schwache Formulierung der Gleichungen aufgestellt, indem die Teile zusammengeführt werden.

Dabei gilt für die Geschwindigkeit $\mathbf{u}_h = u_{h,i} \mathbf{e}_i$ und für die Testfunktion $\boldsymbol{\varphi}_h = \varphi_{h,i} \mathbf{e}_i$, wobei $u_{h,i}, \varphi_{h,i} \in V_h$ und \mathbf{e}_i der Einheitsvektor in i -Richtung ist. Für den Stresstensor gilt die Schreibweise $\boldsymbol{\sigma}_h = \sigma_{h,ij} \mathbf{e}_i \mathbf{e}_j$. Der Druck und die skalare Testfunktion sind dabei $p_h, q_h \in V_h$.

Zur vereinfachten Schreibweise werden die einzelnen Termteile gesondert betrachtet und wie in [1] betitelt.

KONVEKTIVER TERM Unter Zuhilfenahme des Lax-Friedrichs-Flusses, der einen Strafterm bei der Integration berücksichtigt, ergibt sich für die konvektiven Terme für zwei Geschwindigkeiten \mathbf{u}_h und \mathbf{w}_h :

$$\begin{aligned} T(\mathbf{u}_h, \mathbf{w}_h, \boldsymbol{\varphi}_h) = & \\ & - \int_{\Omega} (\mathbf{u}_h \otimes \mathbf{w}_h) : \nabla \boldsymbol{\varphi}_h + \int_{\mathcal{F}_h} \left(\{\!\!\{ \mathbf{u}_h \otimes \mathbf{w}_h \}\!\!\} \cdot \mathbf{n}_{\mathcal{F}} + \frac{1}{2} \Lambda [\![\mathbf{u}_h]\!] \right) \cdot [\![\boldsymbol{\varphi}_h]\!] , \end{aligned} \quad (5.37)$$

mit $\Lambda = \max(\lambda|_{T_1}, \lambda|_{T_2})$ und λ als Eigenwert der Jacobi-Matrix $(\partial[(\mathbf{u} \otimes \mathbf{w}) \cdot \mathbf{n}_{\mathcal{F}}] / \partial \mathbf{u})_{\bar{\mathbf{u}}, \bar{\mathbf{w}}}$.

Abweichend von der vorherigen Regel für Durchschnitts- und Sprungoperator wie in Gleichung (5.36) definiert, gilt für das dyadische Produkt der Geschwindigkeiten und die Geschwindigkeiten selbst:

$$\begin{aligned} \{\!\!\{ \mathbf{u}_h \otimes \mathbf{w}_h \}\!\!\} &= \frac{1}{2} ((\mathbf{u}_h \otimes \mathbf{w}_h)|_T + (\mathbf{u}_h \otimes \mathbf{w}_h)) , & \mathcal{F} \subset \partial\Omega_D , \\ [\![\mathbf{u}_h]\!] &= (\mathbf{u}_h|_T - \mathbf{u}_D) & \mathcal{F} \subset \partial\Omega_D . \end{aligned} \quad (5.38)$$

Diese Regel gilt sowohl für die Geschwindigkeit \mathbf{u}_h wie auch für die einzelnen Komponenten $u_{h,i}$.

LAPLACETENSOR DES DRUCKS Bei der Diskretisierung des Laplacetensors des Drucks wird ein symmetrischer innerer Strafterm, in Form des sogenannten *Symmetric Interior Penalty* (SIP) Flusses [61], berücksichtigt:

$$\begin{aligned} a(p_h, q_h) = & \\ & - \int_{\Omega} \nabla p_h \cdot \nabla q_h + \int_{\mathcal{F}_h} (\{\!\!\{ \Delta p_h \}\!\!\} [\![q_h]\!] + \{\!\!\{ \Delta q_h \}\!\!\} [\![p_h]\!]) \cdot \mathbf{n}_{\mathcal{F}} - \int_{\mathcal{F}_h} \mu_p [\![p_h]\!] [\![q_h]\!] , \end{aligned} \quad (5.39)$$

wobei μ_p besagter Strafterm des Flusses ist. Minimalwerte für diesen Strafterm im Fall eines dreieckigen und tetraedrischen Gitternetzes sind in [62] vorgeschlagen. Für eine allgemeine Beschreibung, die auch z. B. hybride Gitter abdeckt, kann [63] konsultiert werden. Für viereckige Elemente werden die folgenden Parameter gesetzt:

$$\mu_p = \alpha_p c = \begin{cases} \alpha_p \max(c|_{T_1}, c|_{T_2}) & \mathbf{F} \in \mathcal{F}_h^i \\ \alpha_p c|_T & \mathbf{F} \in \mathcal{F}_h^b \end{cases} , \quad (5.40)$$

$$c_T = (k+1)^2 \frac{A \frac{\partial T \setminus \mathcal{F}_h^b}{2} + A (\partial T \cap \mathcal{F}_h^b)}{V(T)} . \quad (5.41)$$

Dabei ist $\alpha_p \geq 1$ ein benutzerdefinierter Koeffizient, der zur Stabilisierung des Algorithmus dient. A ist die Fläche der Elemente, V ihr Volumen.

Hier weicht der Durchschnittsoperator des Gradienten des Drucks sowie der Sprungoperator des Drucks von der Regel aus Gleichung (5.36) ab:

$$\begin{aligned} \{\!\!\{ \nabla p_h \}\!\!\} \cdot \mathbf{n}_{\mathcal{F}} &= \mathbf{n}_{\mathcal{F}} \cdot \frac{\mathbf{h}_h^n}{\Delta t} , & \mathcal{F} \subset \partial\Omega_D , \\ [\![p_h]\!] &= 0 , & \mathcal{F} \subset \partial\Omega_D . \end{aligned} \quad (5.42)$$

DIVERGENZ DES ZWISCHENTERMS Zur Abschätzung der Divergenz der Zwischengeschwindigkeit \mathbf{u}'_h wird folgende Schreibweise eingeführt:

$$B\left(\frac{\mathbf{u}'_h}{\Delta t}, q_h\right) = - \int_{\Omega} \frac{\mathbf{u}'_h}{\Delta t} \cdot \nabla q_h + \int_{\mathcal{F}_h} \left\{ \left\{ \frac{\mathbf{u}'_h}{\Delta t} \right\} \right\} \cdot \mathbf{n}_{\mathcal{F}} [\![q_h]\!] . \quad (5.43)$$

GRADIENT DES DRUCKS Analog kann auch der diskrete Gradient des Drucks behandelt werden. Dieser wird komponentenweise betrachtet und mit einem Index B_{gr} versehen:

$$B_{gr}(p_h, \boldsymbol{\varphi}) = B(p_h \mathbf{e}_i, \varphi_{h,i}) = - \int_{\Omega} p_h \mathbf{e}_i \cdot \nabla \varphi_{h,i} + \int_{\mathcal{F}_h} \{p_h\} \mathbf{e}_i \cdot \mathbf{n}_{\mathcal{F}} [\varphi_{h,i}] . \quad (5.44)$$

DISKRETE FORM DER VISKOSEN TERME Ebenso wird auch die diskrete Form der Divergenz des Spannungstensors $\boldsymbol{\sigma}$ gebildet. Diese entspricht dem viskosen Term der Beschleunigung. Da es sich beim Spannungstensor um eine tensorielle Größe handelt, wird auch hier $B(\cdot, \cdot)$ komponentenweise behandelt, um $B_{st}(\cdot, \cdot)$ zu erhalten. Zusätzlich wird auch hier ein Strafterm eingeführt:

$$B_{st} \left(\frac{\boldsymbol{\sigma}_h}{\rho}, \boldsymbol{\varphi}_h \right) = B \left(\frac{\sigma_{h,ij}}{\rho} \mathbf{e}_j, \varphi_{h,i} \right) - \int_{\mathcal{F}_h} \mu_v \mu_p [\mathbf{u}_{h,i}] [\varphi_{h,i}] , \quad (5.45)$$

$$\begin{aligned} &= - \int_{\Omega} \frac{\sigma_{h,ij}}{\rho} \mathbf{e}_j \cdot \nabla \varphi_{h,i} \\ &\quad + \int_{\mathcal{F}_h} \left\{ \left\{ \frac{\sigma_{h,ij}}{\rho} \right\} \right\} \mathbf{e}_j \cdot \mathbf{n}_{\mathcal{F}} [\varphi_{h,i}] \\ &\quad - \int_{\mathcal{F}_h} \mu_v \mu_p [\mathbf{u}_{h,i}] [\varphi_{h,i}] . \end{aligned} \quad (5.46)$$

Im Fall der Divergenz des Stresstensors $\boldsymbol{\sigma}$ darf der Strafterm nicht nur von der Art des Gitters abhängen, da der Stress eine Funktion der auf das System wirkenden Deformation ist. Die Divergenz des Stresstensors hängt also von der zweiten Ableitung der Geschwindigkeit ab. Entsprechend ist es sinnvoll, einen Strafparameter

$$\mu_v = \frac{\mu}{\rho} \quad (5.47)$$

einzufigen. Dabei ist μ die dynamische Viskosität eines äquivalenten Newtonschen Fluids ist. Zudem geht der zuvor eingeführte Strafparameter μ_p in den Strafterm ein.

DG-METHODE Setzt man die zuvor eingeführten Terme in die schwache Formulierung der zu lösenden Gleichungen aus (5.21) ein, erhält man:

$$(\mathbf{u}'_h, \boldsymbol{\varphi}_h) = \Delta t \left(-T(\mathbf{u}_h^n, \mathbf{u}_h^n, \boldsymbol{\varphi}_h) + \frac{1}{Re} B_{st} \left(\frac{\boldsymbol{\sigma}_h^n}{\rho}, \boldsymbol{\varphi}_h \right) + \left(\frac{\mathbf{g}_h^n}{\rho}, \boldsymbol{\varphi}_h \right) \right) \quad (5.48)$$

$$\alpha(\bar{p}_h^{n+1}, q_h) = B \left(\frac{\mathbf{u}'_h}{\Delta t}, q_h \right) , \quad (5.49)$$

$$(\mathbf{u}_h^*, \boldsymbol{\varphi}_h) = (\mathbf{u}'_h, \boldsymbol{\varphi}_h) - \Delta t B_{gr}(\bar{p}_h^{n+1}, \boldsymbol{\varphi}_h) , \quad (5.50)$$

$$(\mathbf{u}_h^{n+1}, \boldsymbol{\varphi}_h) = (\mathbf{u}_h^n, \boldsymbol{\varphi}_h) + (\mathbf{u}_h^*, \boldsymbol{\varphi}_h) . \quad (5.51)$$

Diese Gleichungen repräsentieren die diskontinuierliche Galerkin-Methode, die für die Approximation der Fluid-Gleichungen angewendet werden. Die Integrale über das Rechengebiet Ω sind mit einer Gauß-Quadratur approximiert. Diskrete Lösungen $\mathbf{u}_h, p_h \in V_h$ sind auf jedem Gitterelement bilineare oder biquadratische Polynome.

Teil III

UNTERSUCHUNG VON MD-SIMULATIONEN IM RAHMEN VON HYBRID-SCHEMATA

PARAMETEROPTIMIERUNG FÜR HYBRIDSIMULATIONEN VON WCA-TEILCHEN

In diesem Kapitel sollen die Parameter für Kolloide, die mit dem WCA-Potential modelliert werden, unter dem Aspekt der Hybridsimulation untersucht und optimiert werden. Da bei jedem einzelnen Zeitschritt der CFD-Simulationen für jeden Quadraturpunkt eine MD-Simulation nötig sein könnte, muss die Mikroskalensimulation optimiert werden, so dass die Systeme möglichst schnell gute und verlässliche Ergebnisse liefern. Aus diesem Grund wird in den nächsten Abschnitten untersucht, welches der beiden Thermostate für eine „on-the-fly“-Realisierung der Kopplung am besten geeignet ist.

Anschließend wird für das isokinetische Thermostat überprüft, welcher Zeitschritt akkurate Ergebnisse liefert, sodass die Simulationen entsprechend beschleunigt werden können, ohne die Resultate zu verändern. Zudem wird bestimmt, ab welcher Boxlänge in einer Dimension es zu Finite-Size-Effekten kommt, d. h., welche Boxdimensionen mindestens vonnöten sind, um verlässliche Ergebnisse zu liefern. Auf diese Weise kann später die Anzahl der Teilchen konstant gehalten werden, während die Dimension in der ungeicherten y -Richtung variiert wird.

Die in diesem Kapitel vorgestellten Ergebnisse wurden z. T. bereits in [2] präsentiert. Die diesem Paper entnommenen Bilder sind entsprechend gekennzeichnet. Die Beschreibungen der Bilder wurden für diesen Zweck ins Deutsche übersetzt und gegebenenfalls erweitert.

Es sei angemerkt, dass aufgrund der guten Statistik die Fehlerbalken in den gezeigten Bildern nicht oder kaum größer als die dargestellten Datenpunkte sind.

6.1 VERGLEICH VERSCHIEDENER THERMOSTATE

Bei den in diesem Abschnitt untersuchten Thermostaten handelt es sich um die in Kapitel 3 vorgestellten, das Lowe-Andersen-Thermostat (LAT) und das isokinetische Thermostat (ISO). In diesem Abschnitt werden die beiden Thermostate miteinander verglichen und auf ihre Verwendbarkeit für die Hybridsimulation hin überprüft.

Eine erste wichtige Eigenschaft ist, ob und wie schnell sich in den jeweiligen Thermostaten das lineare Scherprofil einstellt. Da im isokinetischen Thermostat der SLLOD-Algorithmus zu Hilfe genommen und das Scherprofil damit nicht nur über die Randbedingungen, sondern auch über die Bewegungsgleichungen angelegt wird, bildet sich das Scherprofil schneller aus. Im Lowe-Andersen-Thermostat wird es lediglich über die Randbedingungen angelegt und benötigt dadurch mehr MD-Schritte, bis es eingestellt ist. Die Badkollisionswahrscheinlichkeit des Lowe-Andersen-Thermostats wird, wenn nicht anders angegeben, auf $w = \Gamma \cdot \Delta t = 0.001$ gesetzt.

In Abbildung 6.1 werden die sich einstellenden Scherprofile für die beiden Thermostate gezeigt. Nach einer Relaxationsdauer von 200 MD-Zeiten werden die Trajektorien der einzelnen Teilchen aufgenommen und bis zu einer MD-Zeit von 1500 in den Abbildungen 6.1 (a) für das Lowe-Andersen-Thermostat und (b) das isokinetische Thermostat gemittelt.

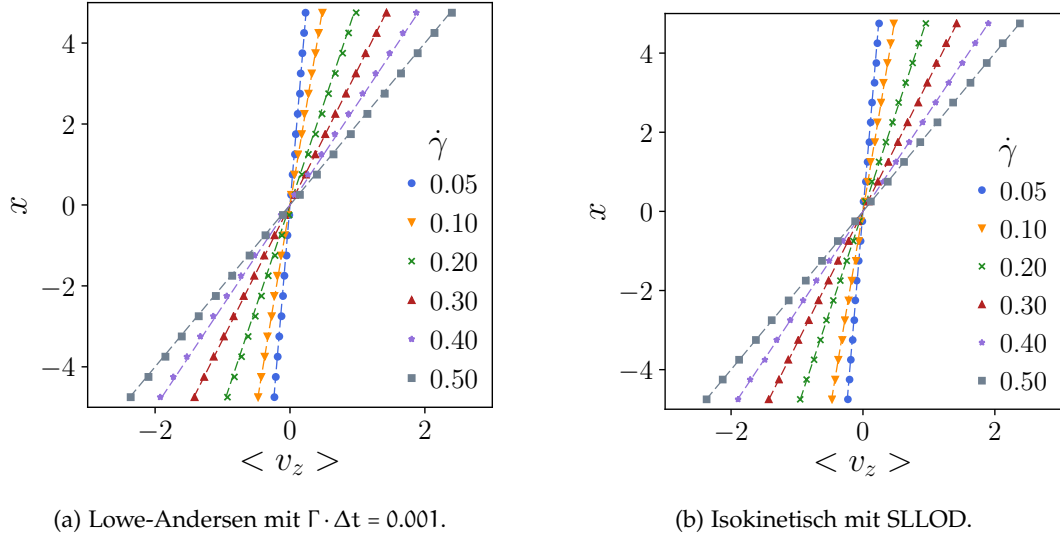


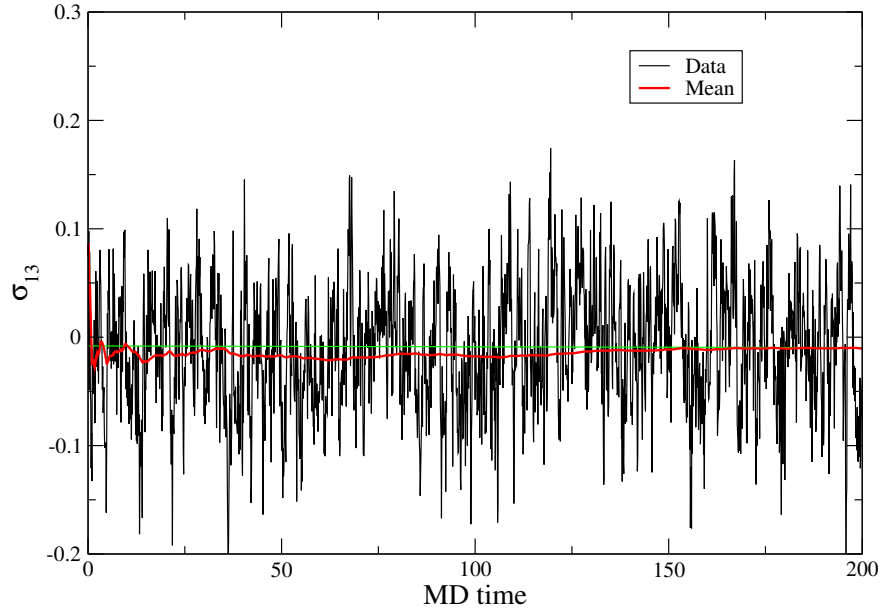
Abb. 6.1: Scherprofile eines Systems mit Dichte $\rho = 0.4$ (400 Teilchen in einem Volumen $V = 10 \times 10 \times 10 \sigma^3$) für die beiden Thermostate für verschiedene Scherraten. Der Integrationszeitschritt ist dabei 10^{-4} . Nach einer Relaxationszeit von 200 MD-Zeiten wird das Scherprofil für eine bestimmte MD-Zeit aufgenommen. In (a) ist dabei das Scherprofil für Simulationen mit dem Lowe-Andersen-Thermostat mit einer Badkollisionswahrscheinlichkeit von $\Gamma \cdot \Delta t = 0.001$ und einer MD-Zeit von 1500 dargestellt. (b) zeigt die Ergebnisse des isokinetischen Thermostats mit SLLOD-Algorithmus für eine MD-Zeit von 1500. Die hier gezeigten Daten sind auch in [2] veröffentlicht worden.

Die Relaxationszeit ist insofern essentiell, als dass die Startkonfigurationen zufällig erzeugt werden. Es braucht also nicht nur einige Schritte, bis sich das Scherprofil einstellen kann, sondern auch, dass die Teilchen relaxierte Positionen eingenommen haben und das System dadurch einen stabilen Zustand erreichen kann.

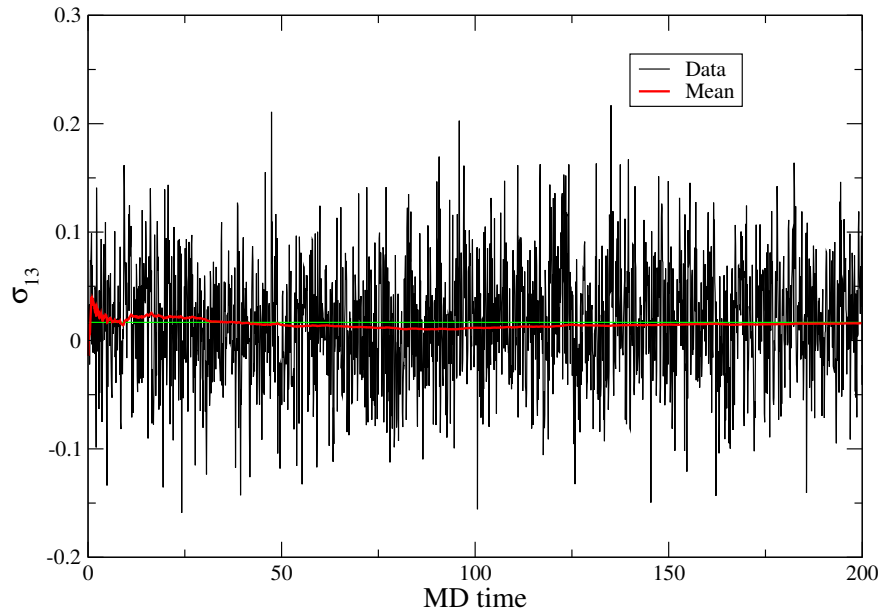
Die verschieden starken Scherraten, die an die untersuchten Systeme angelegt werden, führen bei beiden Thermostaten wie erwartet zu einem linearen Scherprofil proportional zur angelegten Scherrate. Dabei ist zu beachten, dass es bei niedrigen Scherraten ($\dot{\gamma} < 0.01$) wesentlich länger dauert, bis die Randbedingungen zu dem gewünschten linearen Scherprofil führen.

Das eingestellte lineare Scherprofil zeigt an, dass das System einen stabilen Zustand (engl. *Steady State*) angenommen hat. Neben diesem Profil, das von den Randbedingungen erzeugt (und beim isokinetischen Thermostat durch den SLLOD-Algorithmus an die Bewegungsgleichungen gekoppelt) wird, zeigt auch die Zeitentwicklung der einzelnen interessanten Größen, ob die für die Messung relevante Größe bereits relaxiert ist, also keine Einschwingvorgänge zu berücksichtigen sind. Diese Einschwingvorgänge können z. B. durch zufällig erzeugte Startkonfigurationen auftreten, da diese Art der Initialisierung Teilchen unter Umständen näher platziert, als laut Energielandschaft günstiger wäre.

In Abbildung 6.2 ist zur Untersuchung dieser Einschwingeffekte in schwarz die Zeitentwicklung der Nebendiagonalkomponente des Stresstensors σ_{13} mit angelegter Scherrate für die beiden verwendeten Thermostate exemplarisch anhand einer Scherrate von $\dot{\gamma} = 0.05$ für ein System mit Dichte $\rho = 0.4$ gezeigt. Neben den Daten sind auch die Durchschnittswerte dargestellt. Die rote Kurve des mitlaufenden Mittelwerts nähert sich nach unter 200 MD-Zeiten dem grünen Gesamtdurchschnitt (welcher auch spätere Werte bis 1500 MD-Zeiten berücksichtigt). Für die mit dem Lowe-Andersen-Thermostat temperier-

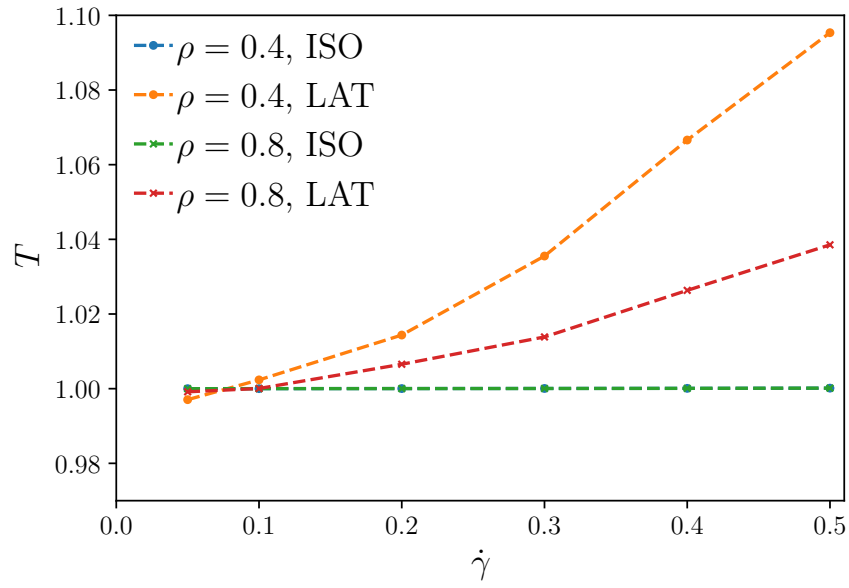


(a) Lowe-Andersen-Thermostat.

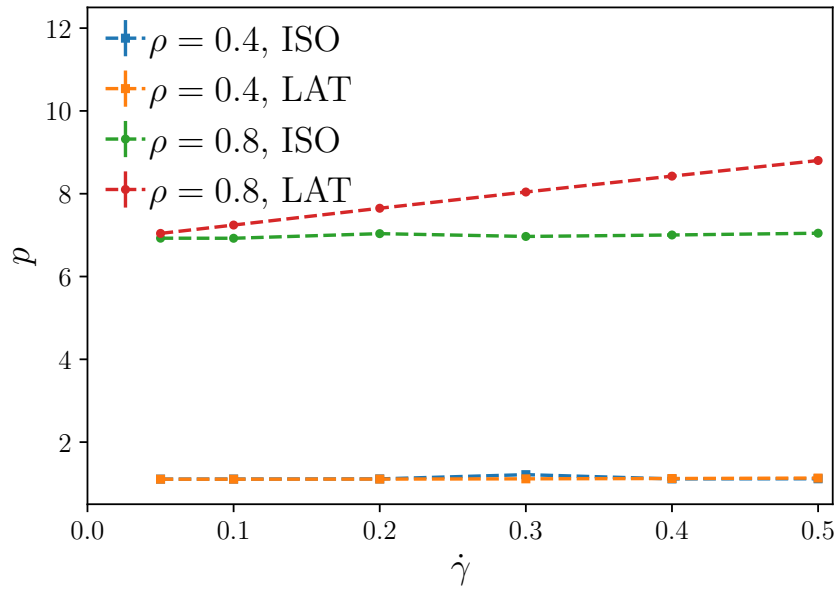


(b) Isokinetisches Thermostat.

Abb. 6.2: Einschwingvorgang der Größe σ_{13} zur Untersuchung der Zeit bis zum Auftreten des *Steady State* bei einer Scherrate von $\dot{\gamma} = 0.05$ für ein System mit Dichte $\rho = 0.4$ (a) für das Lowe-Andersen-Thermostat und (b) für das isokinetische Thermostat. Die momentanen Werte von σ_{13} sind in schwarz dargestellt, der Durchschnitt zu einer bestimmten MD-Zeit in rot und der Gesamtdurchschnitt in grün. Die hier gezeigten Daten in (b) sind auch in [2] veröffentlicht worden.



(a) Temperatur vs. Scherrate.



(b) Druck vs. Scherrate.

Abb. 6.3: Vergleich zwischen Lowe-Andersen- und isokinetischem Thermostat. (a) Temperatur und (b) Druck als Funktion der Scherrate nach einer Relaxationszeit von 200 MD-Zeiten. Das System besitzt ein Volumen von $V = 10 \times 10 \times 10 \sigma^3$, der Zeitschritt beträgt $\Delta t = 10^{-4}$. Im Lowe-Andersen-Thermostat ist die Wahrscheinlichkeit w auf $w = \Gamma \cdot \Delta t = 0.001$ gesetzt. Beim Lowe-Andersen-Thermostat kann die Temperatur in (a) für hohe Scherraten nicht gehalten werden, die Abweichung bei einer Scherrate von $\dot{\gamma} = 0.5$ beträgt etwa 1% bei einer Dichte von $\rho = 0.8$ und sogar 10% bei einer Dichte von $\rho = 0.4$. Die Abweichung im Druck in (b) ist nur bei der hohen Dichte ersichtlich. Die hier gezeigten Daten sind auch in [2] veröffentlicht worden.

ten Systeme in 6.2 (a) ist der Einschwingvorgang nach etwa 100 MD-Zeiten beendet. Die stärksten Schwankungen und eine erste Näherung für das isokinetische Thermostat in Abbildung 6.2 (b) tritt bei unter 50 MD-Zeiten auf. Um sicher zu gehen, wird eine großzügige Relaxationszeit von 200 MD-Zeiten für beide Thermostate angenommen. Dieser Umstand gilt auch für andere Scherraten. Entsprechend wird im Folgenden eine Messung stets nach der Relaxationszeit von 200 MD-Zeiten begonnen, sofern nicht anders beschrieben.

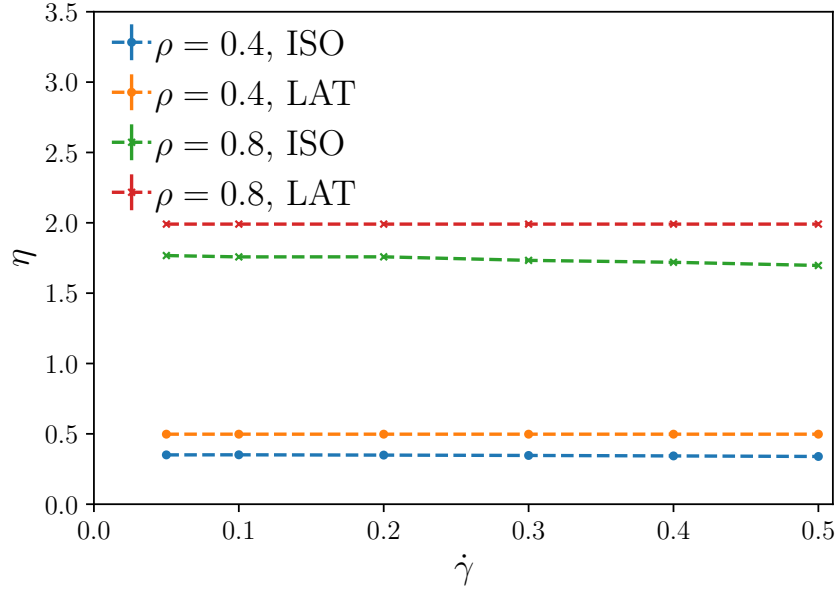
Im Folgenden wird untersucht, inwieweit das Lowe-Andersen-Thermostat in der Lage ist, das System adäquat auf der eingestellten Temperatur zu halten. Für das isokinetische Thermostat erwartet man konstruktionsbedingt keine Abweichungen in Temperatur und Druck.

In Abbildung 6.3 (a) ist beim Lowe-Andersen-Thermostat eine deutliche Temperaturerhöhung bei hohen Scherraten zu erkennen. Simuliert werden zwei Systeme mit je einer Größe von jeweils 10σ in jede Raumrichtung und einer Dichte von $\rho = 0.4$ und $\rho = 0.8$. Die Systeme werden einmal mit dem Lowe-Andersen und einmal mit dem isokinetischen Thermostat temperiert. Das isokinetische Thermostat reskaliert wie erwartet die Temperatur sehr gut, während das Lowe-Andersen-Thermostat eine Scherratenabhängigkeit verzeichnet. Die Abweichung für sehr hohe Scherraten von $\dot{\gamma} = 0.5$ beträgt dabei bei dem System mit Dichte $\rho = 0.4$ etwa 10% und führt zu einer Temperatur von $T \approx 1.1$. Dass die Temperatur bei niedriger Dichte weiter ansteigt, ist auf die geringere Nachbaranzahl innerhalb des Interaktionsradius des Thermostats zurückzuführen. Da weniger Nachbarn bei gleichbleibender Interaktionswahrscheinlichkeit vorhanden sind, reduziert sich auch die Temperaturänderungsrate.

Neben der korrekten Einstellung der Temperatur spielt auch der Druck für Hybridsimulationen eine große Rolle. In Abbildung 6.3 (b) ist der Druck gegen die Scherrate für die zuvor betrachteten Systeme aufgetragen. Für das weniger dichte System liegen die beiden Druckkurven der getesteten Thermostate nahezu aufeinander. Für das dichtere System weicht die rote Kurve, die mit dem Lowe-Anderson-Thermostat thermostiert wurde, für hohe Scherraten um etwa 10% von der grünen, mit dem isokinetischen Thermostat temperierten Kurve ab. Die Erhöhung des Druckwertes beim isokinetischen Thermostat beläuft sich auf wenige Prozent, es ist auch hier eine Abhängigkeit von der Scherrate zu verzeichnen. Diese Erhöhung ist aufgrund der Skala der p-Achse nicht zu erkennen. Da sich die Temperatur beim Lowe-Andersen-Thermostat mit steigender Scherrate erhöht, ist die Steigerung des Drucks vor allem darauf zurückzuführen.

Auch die Viskosität ist ein Parameter, der später in die Kopplung von CFD- und MD-Simulationen indirekt eingehen wird. Die Abhängigkeit der Viskosität von der Scherrate für die bereits behandelten Systeme ist in Abbildung 6.4 (a) zu sehen. Die Viskosität ist annähernd konstant, es handelt sich bei den simulierten harten Kugeln in guter Näherung um ein Newtonsches Fluid. Wie in Abschnitt 3.1 beschrieben, liegt die Viskosität der mit dem Lowe-Andersen-Thermostat gekühlten Systeme bei beiden Dichten über der Viskosität der mit dem isokinetischen Thermostat temperierten Systeme.

Die Abhängigkeit der Viskosität von der Badkollisionsfrequenz Γ des Lowe-Andersen-Thermostats ist in Abbildung 6.4 (b) für verschiedene Interaktionsradien r_C und Zeitschritte Δt zu sehen. Die Viskosität ist dabei in erster Linie nicht von der Kollisionswahrscheinlichkeit $\Gamma \cdot \Delta t$, sondern nur von der Frequenz Γ abhängig. Außerdem ist deutlich zu erkennen, dass die Viskosität für höhere Interaktionsradien r_C zunimmt. Dieser Zusammenhang geht ebenfalls aus Formel (3.6) hervor, die in Abbildung 6.5 mit eingezeichnet ist. Auf diese Weise ist es möglich, im Zusammenhang mit Hybridschemata Fluide mit



(a) Viskosität vs. Scherrate.

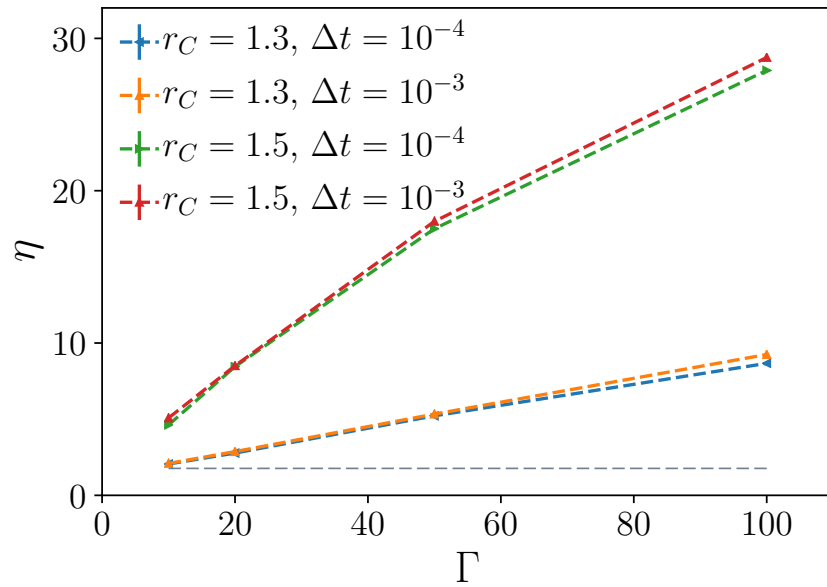
(b) Viskosität vs. Γ .

Abb. 6.4: Vergleich der Viskosität zwischen Lowe-Andersen- und isokinetischem Thermostat. (a) Viskosität als Funktion der Scherrate nach einer Relaxationszeit von 200 MD-Zeiten. Wie in [31] dargestellt, ist die Viskosität des Lowe-Andersen-Thermostats im Vergleich zum isokinetischen Thermostat leicht erhöht. Das System besitzt ein Volumen von $V = 10 \times 10 \times 10 \sigma^3$. Der Zeitschritt beträgt $\Delta t = 10^{-4}$. Im Lowe-Andersen-Thermostat ist die Wahrscheinlichkeit w auf $w = \Gamma \cdot \Delta t = 0.001$ gesetzt. In (b) wird die Viskosität als Funktion der Badkollisionsfrequenz Γ für verschiedene Thermostatinteraktionsradii bei zwei verschiedenen Zeitschritten $\Delta t_1 = 10^{-4}$ und $\Delta t_2 = 10^{-3}$ bei einem System mit Dichte $\rho = 0.8$ und einer Scherrate von $\dot{\gamma} = 0.1$ dargestellt. Der Zeitschritt selbst beeinträchtigt den Verlauf der Kurve nicht. Der Interaktionsradius r_C ist auf $r_C = 1.3$ gesetzt. Die hier gezeigten Daten sind auch in [2] veröffentlicht worden.

höherer Viskosität zu untersuchen, ohne auf ein ressourcenintensives hydrodynamisches Schema wie die Vielteilchendynamik (MPCD) angewiesen zu sein.

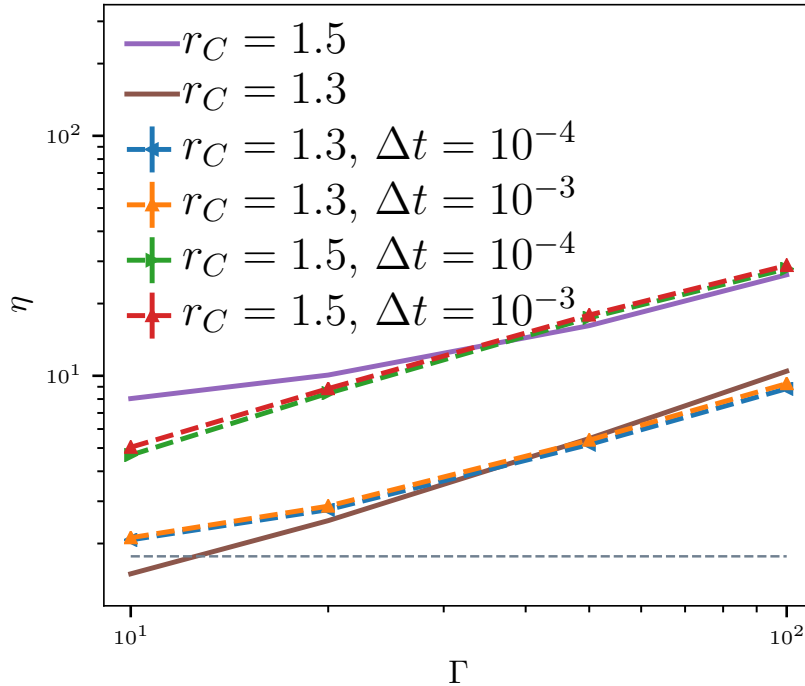
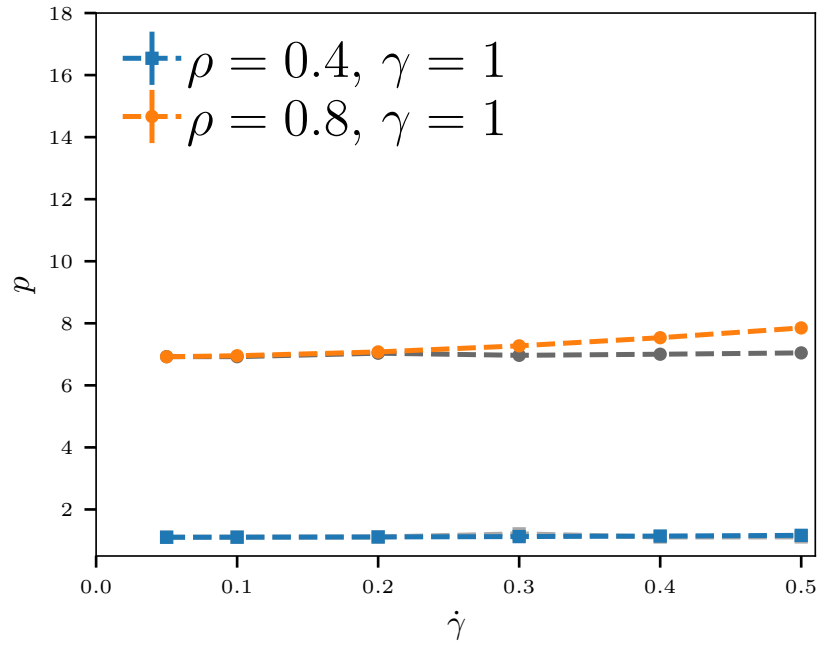


Abb. 6.5: Viskosität in Abhängigkeit der Parameter des Lowe-Andersen-Thermostats bei einer Teilchendichte von $\rho = 0.8$ und Scherrate $\dot{\gamma} = 0.1$. Hier ist auch der Verlauf der Kurven aus Gleichung 3.6 mit eingezeichnet. Dabei ist die Viskosität in Abhängigkeit der Badkollisionsfrequenz Γ gezeigt. Die Kurven sind für zwei unterschiedliche Interaktionsradii $r_C = 1.3$ bzw. $r_C = 1.5$ aufgenommen. Der tatsächliche Verlauf stimmt dabei nicht exakt mit dem theoretisch erwarteten überein. Für hohe Werte für die Badkollisionsfrequenz Γ kann der theoretische Verlauf besser reproduziert werden.

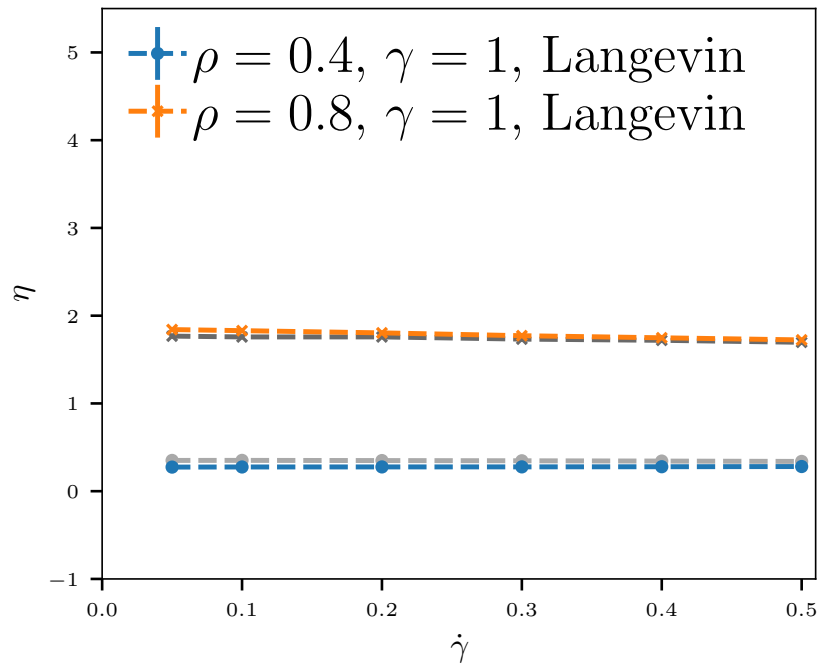
In Abbildung 6.5 wird die Abhängigkeit der Viskosität η von den beiden Thermostatparametern Interaktionsradius r_C und Badkollisionsfrequenz Γ untersucht und mit den Theoriekurven aus Gleichung (3.6) verglichen. Dabei ist ersichtlich, dass die aus den Simulationen erhaltenen Kurven nicht exakt mit der Theorie übereinstimmen. Dies kann vor allem an den Annahmen, die in der Berechnung des Zusatzterms gemacht werden, liegen. Für hohe Werte für die Badkollisionsfrequenz Γ stimmt der Verlauf besser überein.

Da das isokinetische Thermostat nicht nur das Scherprofil schneller einstellt, sondern auch eine günstigere Laufzeit besitzt, werden Hybridsimulationen mit dem isokinetischen Thermostat samt SLLOD-Methode im Folgenden bevorzugt durchgeführt. Auch wenn beide Algorithmen unphysikalisch sind, ist dies für die Hybridsimulationen irrelevant, da hierfür keine realistische Dynamik vonnöten ist, sondern die Eigenschaften des Fließgleichgewichts ermittelt werden sollen.

Mithilfe des Langevin-Thermostats soll nun gezeigt werden, dass das isokinetische Thermostat korrekte Ergebnisse des Fließgleichgewichts liefert und damit die zuvor in [2] getroffene Aussage korrekt ist. Da das isokinetische Thermostat künstlich die Geschwindigkeiten und damit die Temperatur reskaliert, können Einschwingvorgänge nicht unter-



(a) Druck vs. Scherrate.



(b) Viskosität vs. Scherrate.

Abb. 6.6: Vergleich zwischen Langevin- und isokinetischem Thermostat. (a) Druck und (b) Viskosität als Funktion der Scherrate nach einer Relaxationszeit von 200 MD-Zeiten. Die Systeme besitzen ein Volumen von $V = 10 \times 10 \times 10 \sigma^3$ mit einer Dichte von $\rho = 0.4$ ($N = 400$) und $\rho = 0.8$ ($N = 800$). Der Zeitschritt ist auf 10^{-3} gesetzt. Für das Langevin-Thermostat wird der Reibungskoeffizient auf $\gamma = 1$ gesetzt. Die Werte des isokinetischen Thermostats für die jeweiligen Dichten sind in grau eingezeichnet. Man sieht deutlich, dass für eine hohe Dichte von $\rho = 0.8$ die Werte der beiden Thermostate stärker abweichen. Für hohe Scherraten ist dieser Effekt beim Druck stärker ausgeprägt.

sucht werden. Für eine Anwendung bei Hybrid-Simulationen sind diese Dynamiken nicht relevant, lediglich der Mittelwert im Fließgleichgewicht ist von Bedeutung.

Wie in Abbildung 6.6 zu sehen ist, stimmen die Werte der beiden Thermostate für alle Scherraten überein. Hier werden zwei Systeme unterschiedlicher Dichte ($\rho = 0.4$ und $\rho = 0.8$) betrachtet und mit dem Langevin-Thermostat gekühlt. Dabei ist der Reibungskoeffizient γ auf 1 gesetzt. In Abbildung 6.6 a) ist der Druck dargestellt, bei dem für hohe Scherraten eine leichte Scherratenabhängigkeit zu erkennen ist. Dabei ist diese Abhängigkeit signifikanter für das dichtere System bei einer Dichte von $\rho = 0.8$.

Die Viskosität ist in Abbildung 6.6 b) aufgezeigt. Für das Langevin-Thermostat weicht der Wert minimal vom mit dem isokinetischen Thermostat temperierten System ab. Der Verlauf für das dünne System variiert lediglich um eine minimale Verschiebung, während der Verlauf bei hoher Dichte von der eines Newtonschen Fluids abweicht. Hier gezeigt sind vorläufige Ergebnisse.

Die Annahme, dass das isokinetische Thermostat die richtigen Ergebnisse liefert, kann also bestätigt werden. Die Nutzung dieses Thermostats ist legitim.

6.2 PARAMETERWAHL FÜR DAS ISOKINETISCHE THERMOSTAT

In diesem Abschnitt sollen die Simulationsparameter für das isokinetische Thermostat untersucht und die optimalen Parameter für eine Hybridsimulation bestimmt werden. Dabei wird für die Bestimmung des optimalen Zeitschritts ein System untersucht, dessen Dichte $\rho = 0.8$ beträgt. Die Abmessungen der Simulationsbox sind dabei $10 \times 7.5 \times 10 \sigma^3$ mit der entsprechenden Teilchenzahl $N = 600$.

Um mögliche Dichteabhängigkeiten ausschließen zu können, werden bei der anschließenden Untersuchung der Simulationsboxgröße Systeme verschiedener Dichte simuliert. Die Grundseite der Box in x - und z -Richtung bleibt mit $15 \times 15 \sigma^2$ konstant. Die untersuchten Dichten sind $\rho = 0.2, 0.4, 0.6$ und 0.8 . Bei diesen Dichten wird die Boxlänge in y -Richtung variiert.

AUTOKORRELATION Zur Minimierung des Rechenaufwands wird zunächst die Zeitschrittabhängigkeit von Druck und Viskosität bestimmt. Ein eine Größenordnung kleinerer Zeitschritt bedeutet, dass zehn Mal so viele Schritte notwendig sind, um eine feste MD-Zeit zu erreichen. Entsprechend kann bei der Wahl eines um eine Größenordnung größeren Zeitschrittes die Rechenzeit (im günstigsten Fall um 90 %) reduziert werden.

Dabei muss auch die Korrelationszeit für den am Ende gewählten Zeitschritt beachtet werden. Korrelation liegt vor, wenn zwei aufeinander folgende Messungen einander beeinflussen, die Konfigurationen sich also kaum voneinander unterscheiden. Dies beeinflusst auch den Fehler der gemessenen Größe, der tatsächliche Fehler der durch Korrelation behafteten Größe ist wesentlich höher, als der statistische Fehler der Messung. Nach [64] wird die Autokorrelationsfunktion einer Größe A wie folgt bestimmt:

$$\Phi_A(t) = \frac{[\langle A(0)A(t) \rangle - \langle A \rangle^2]}{[\langle A^2 \rangle - \langle A \rangle^2]}. \quad (6.1)$$

Dabei ist $A(t)$ die von der Messzeit t abhängige Messgröße. Die Autokorrelationsfunktion ist streng monoton fallend. Für die Relaxationszeit oder Autokorrelationszeit gilt nun

$$\tau_A \equiv \int_0^\infty \Phi_A(t) dt. \quad (6.2)$$

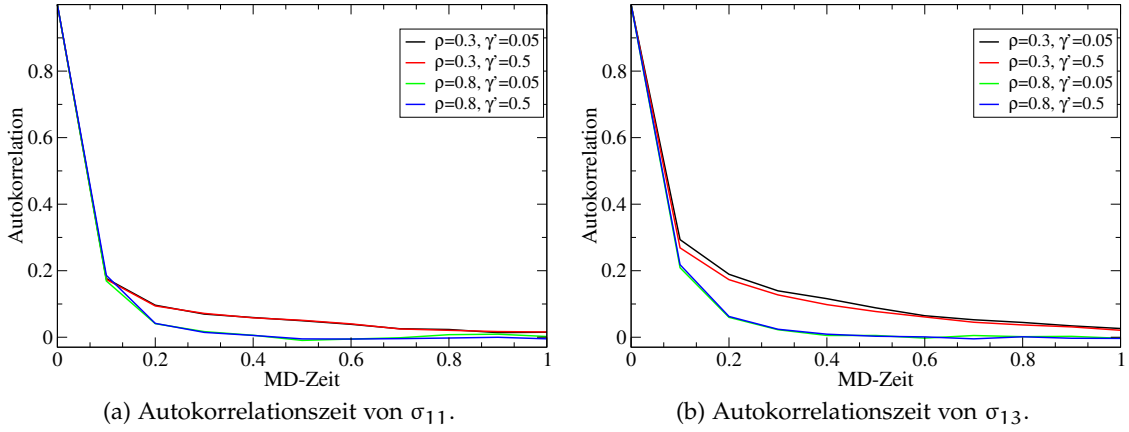


Abb. 6.7: Autokorrelationszeiten von σ_{11} und σ_{13} für einen Zeitschritt von $\Delta t = 10^{-3}$. Die Autokorrelationsfunktion der beiden Messgrößen ist in Abhängigkeit der MD-Zeit dargestellt. Es ist leicht ersichtlich, dass die Korrelationsfunktionen bei einer MD-Zeit von $\tau = 0.08$ auf einen Wert von $\frac{1}{e}$ abgeklungen sind.

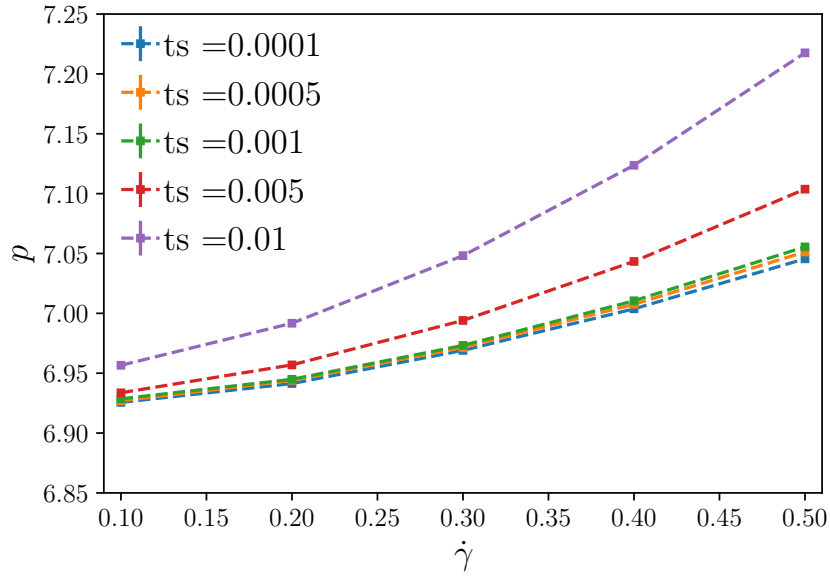
Alternativ kann τ_A auch als die Zeit aufgefasst werden, bei der die Autokorrelationsfunktion auf $\frac{1}{e}$ abgefallen ist. Diese Zeit gibt das Zeitintervall an, in dem die Messgröße korrelierte Ergebnisse angibt. Um den tatsächlichen Fehler einer korrelationsbehafteten Messgröße zu berechnen, muss folgende Korrektur nach [65] angewandt werden:

$$\langle (\Delta A)^2 \rangle = \frac{1}{N} [\langle A^2 \rangle - \langle A \rangle^2] \left(1 + 2 \frac{\tau_A}{\Delta t} \right). \quad (6.3)$$

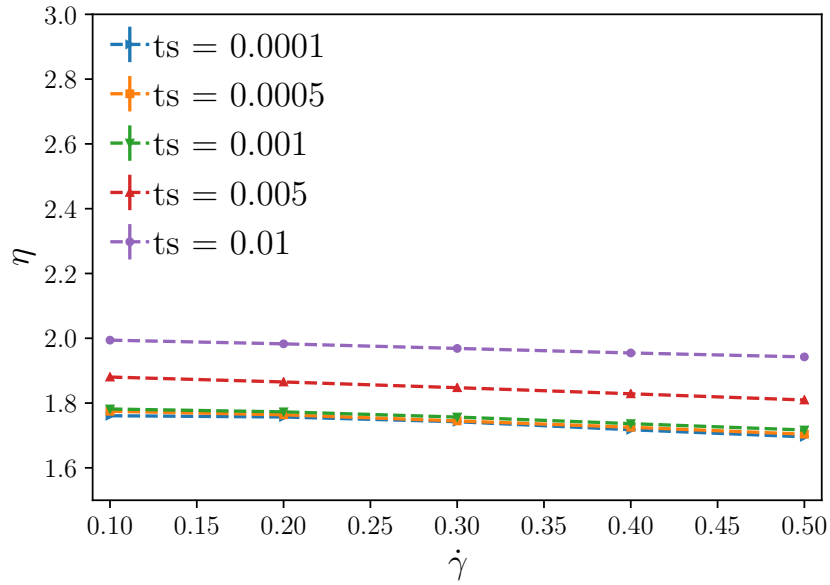
Wie in Abbildung 6.7 beispielhaft gezeigt ist, klingt die Autokorrelationsfunktion bei einer MD-Zeit von $\tau = 0.08$ sowohl für die σ_{11} - als auch die σ_{13} -Komponente des Stresstensors auf einen Wert von $\frac{1}{e}$ ab. Um also die Autokorrelationszeit der einzelnen Komponenten des Stresstensors bei den Messungen bzw. der späteren Analyse nicht berücksichtigen zu müssen und eine einfache Fehleranalyse mit Mittelwert und Fehler durch Bestimmung der Standardabweichung durchführen zu können, genügt es also, bei einem Zeitschritt von $\Delta t = 10^{-3}$ nur jede hundertste Messung zu betrachten. Entsprechend wird das Intervall, um den Stresstensor zu berechnen, auf $\Delta t_{\text{Mess}} = 0.1$ gesetzt, um die Korrelationszeit von $\tau_{\text{corr}} = 0.08$ zu berücksichtigen.

ZEITSCHRITTABHÄNGIGKEIT Der Druck p in Abhängigkeit der Scherrate für verschiedene Zeitschritte ist in Abbildung 6.8(a) gezeigt, die Viskosität η in (b). Beide Bilder zeigen deutlich, dass Zeitschritte größer als $\Delta t = 10^{-3}$ zu einer Abweichung sowohl bei dem Druck, als auch bei der Viskosität führen. Zeitschritte kleiner oder gleich 10^{-3} führen zu Werten, die innerhalb des Fehlers übereinstimmen. Dabei wird deutlich, dass die Abweichung von der Scherrate abhängt. Während bei einem Zeitschritt von $\Delta t = 0.005$ die relative Abweichung bei einer Scherrate von $\dot{\gamma} = 0.1$ für den Druck p bei etwa 0.2% liegt, steigt sie auf 0.8% für $\dot{\gamma} = 0.5$. Bei einem Zeitschritt von $\Delta t = 0.01$ beträgt die relative Abweichung bei der niedrigen Scherrate bereits 0.5% und steigt bei einer Scherrate von $\dot{\gamma} = 0.5$ auf 2,2%.

In Abbildung 6.9 werden Druck p und Viskosität η in Abhängigkeit des gewählten Zeitschrittes für eine feste Scherrate von $\dot{\gamma} = 0.5$ und $\rho = 0.8$ gezeigt. Bei dieser hohen Scherrate sind die Abweichungen am deutlichsten ausgeprägt. Die Abweichung für einen



(a) Druck vs. Scherrate für verschiedene Zeitschritte.



(b) Viskosität vs. Scherrate für verschiedene Zeitschritte.

Abb. 6.8: Vergleich von (a) Druck, (b) Viskosität als Funktion der Scherrate nach der Relaxation für verschiedene Zeitschritte Δt . Die Dichte ist dabei $\rho = 0.8$ in einer Box mit den Abmessungen $10 \times 7.5 \times 10 \sigma^3$. Die Werte bis zu einem Zeitschritt von $\Delta t \leq 0.001$ liegen dicht beieinander, ab einem Zeitschritt von $\Delta t = 0.005$ weichen die Werte für Druck p in (a) und Viskosität η in (b) deutlich ab. Die hier gezeigten Daten sind in [2] veröffentlicht worden.

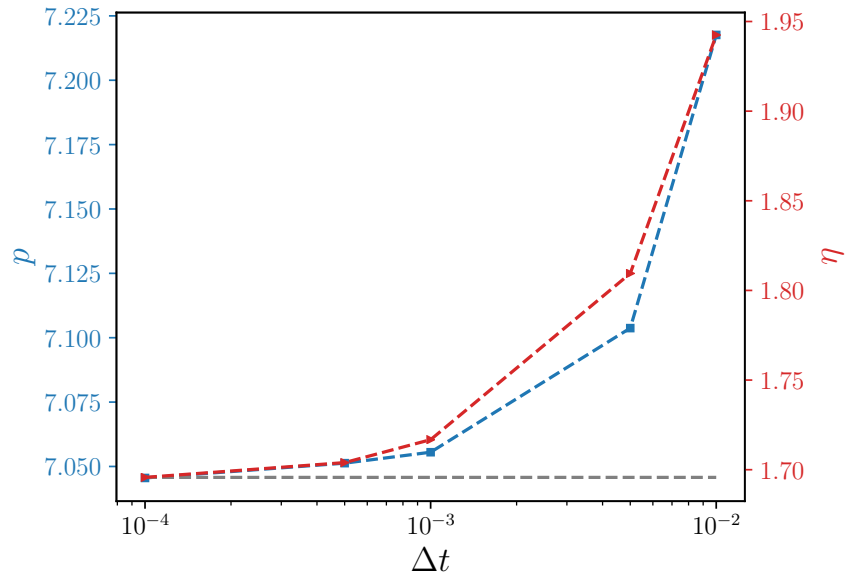


Abb. 6.9: Vergleich von Druck p und Viskosität η als Funktion des Zeitschritts Δt für eine Scherrate von $\dot{\gamma} = 0.5$. Als Hilfslinie ist die Grundlinie der Werte für einen Zeitschritt von $\Delta t = 10^{-4}$ in grau eingezeichnet. Ab einem Wert von $\Delta t = 10^{-3}$ ist die Abweichung vom Idealwert des niedrigen Zeitschrittes vernachlässigbar. Dies ist sowohl im Druck in blau (linke Skala), als auch in der Viskosität in rot (rechte Skala) ersichtlich. Auch hier beträgt die Dichte $\rho = 0.8$.

Zeitschritt von $\Delta t = 10^{-3}$ ist dabei für dieses *Worst-Case-Szenario* angemessen gering genug, um den Zeitschritt auf diesen Wert anheben zu können. Für die Viskosität ist der Verlauf in Abhängigkeit des Zeitschritts nahezu äquivalent.

Durch die Auftragung beider Achsen ist ein besserer Vergleich möglich. So erkennt man deutlich, dass der qualitative Verlauf beider Größen gut übereinstimmt. Auch hier ist eine signifikante Abweichung ab einem Zeitschritt von $\Delta t = 0.005$ zu erkennen. Entsprechend wird der Zeitschritt der MD-Simulation als Grundlage für Hybridschemata in den nachfolgenden Kapiteln auf $\Delta t = 10^{-3}$ gesetzt, falls nicht anders gekennzeichnet.

GRÖSSENUNTERSUCHUNG Im Folgenden wird im Zusammenhang mit Hybridschemata die Frage untersucht, ob viele parallele, kleine und voneinander unabhängige Systeme mit guter Statistik einem einzelnen großen System vorzuziehen sind. Bei korrekter Implementierung müssen bei jeder Kraftberechnung für jedes Teilchen mindestens eine konstante Anzahl von Wechselwirkungen berechnet werden (sofern das Paarpotential kurzreichweitig ist). Der Rechenaufwand pro Zeitschritt wächst also mindestens linear mit der Teilchenzahl N , was in der Praxis jedoch generell nicht erreicht wird. Daher ist es prinzipiell besser, viele kleine Systeme als ein großes zu simulieren und über diese zu mitteln. Ebenso dafür spricht die Relaxationszeit, die bei einem größeren System wesentlich länger ist.

In Tabelle 6.1 sind exemplarisch die Werte von MD-Simulationen zweier Systeme aufgezeigt, die sich in ihrer Systemdimension unterscheiden. Die Werte dienen einer groben Orientierung. Beide Simulationen sind bei einer Teilchendichte von $\rho = 0.5$ durchgeführt, eines der Systeme ist in y -Richtung jedoch 5 Mal größer als das andere. Die beiden Systeme werden mit dem isokinetischen Thermostat temperiert und mit einer Scherrate von

Teilchenzahl N	Volumen V	MD-Schritte / Sekunde	Dauer für 1 000 MD-Zeiten
500	$10 \times 10 \times 10 \sigma^3$	2 599.69	3 847 s
2 500	$10 \times 50 \times 10 \sigma^3$	472.615	21 161 s

Tabelle 6.1: Simulationsdetails zweier unterschiedlich großer Systeme bei gleicher Dichte.

$\dot{\gamma} = 0.5$ gesichert. Zur korrekten Einstellung des Scherprofils wird der SLLOD-Algorithmus verwendet. Das Zeitintervall eines MD-Schrittes ist auf $\Delta t = 0.001$ gesetzt.

Bei einem System mit $N_1 = 500$ Teilchen und einer y-Ausdehnung von 10σ der Simulationsbox können bei dieser Konfiguration fast 2 600 MD-Schritte pro Sekunde durchgeführt werden. Eine Simulation mit einer voreingestellten Zeit von 1 000 MD-Zeiten kann somit in einer Echtzeit von 3 847 Sekunden beendet werden. Das in der y-Komponente 5 Mal größere System, das bei gleicher Dichte entsprechend $N_2 = 2 500$ Teilchen enthält, benötigt für die gleiche Simulationszeit ganze 21 161 Sekunden. Die Leistung des Programms erreicht hier nur knapp 473 MD-Schritte pro Sekunde.

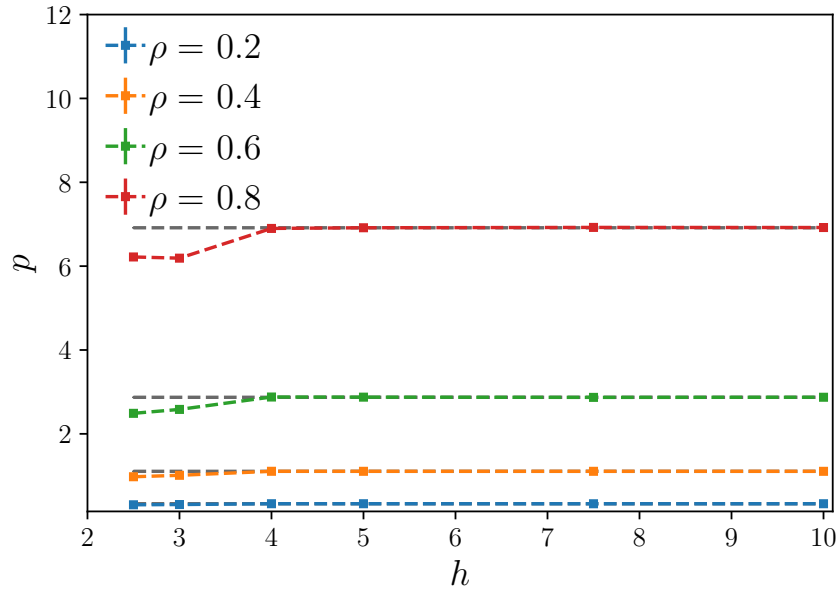
Geht man davon aus, dass 5 Mal mehr Teilchen zu einer ungefähr 5 Mal besseren Statistik für intrinsische Größen wie den Stresstensor σ führt, muss das kleinere System also 5 Mal solange simulieren, um ein Ergebnis mit ähnlicher Statistik in dieser Größe zu erzeugen.

Wenn das kleinere System für 5 000 MD-Zeiten Daten sammelt, würde es den Werten nach aber nur 19 235 Sekunden benötigen, während das größere für nur 1 000 Zeiten etwas mehr als 21 000 Sekunden braucht. Das große System benötigt also etwa 30 Minuten länger. Man erhält insgesamt eine Zeitersparnis von etwa 10 %, wenn man das kleinere System für eine ähnliche Statistik länger simuliert.

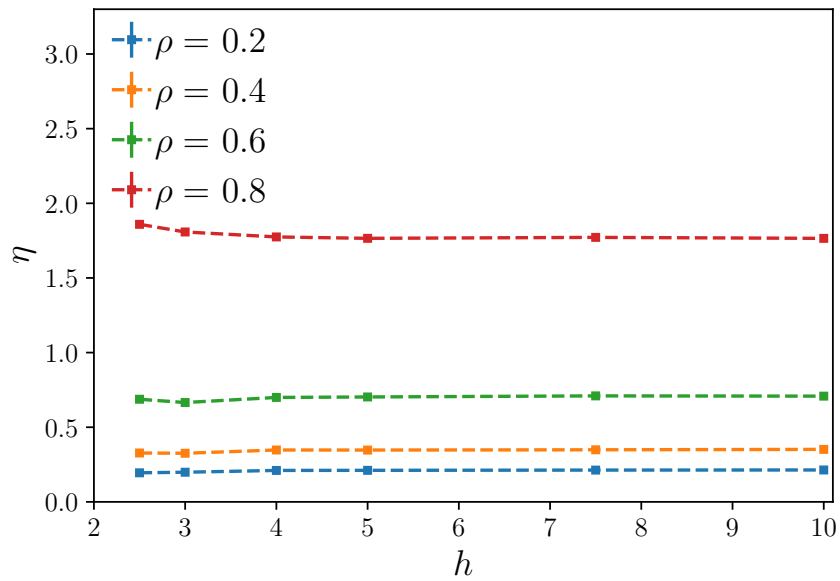
Auch hinsichtlich einer Parallelisierung sind kleine Simulationsboxen nützlicher. Die triviale „Parallelisierung“ in Form mehrerer, parallel laufender voneinander unabhängiger Simulationen, die zur Gewinnung von Statistik dienen und dazu nicht untereinander kommunizieren müssen, ist wesentlich weniger zeitaufwendig als ein großes Simulationsgebiet, das in einzelne, von einem Prozessor zu bearbeitende Scheiben geteilt wird und deren Randbereiche zwischen zwei Scheiben nur durch Kommunikation der beiden Prozesse bearbeitet werden können. Dabei ist zu beachten, dass ein System (auch und insbesondere ein kleines System) eine gewisse Mindestzahl an Teilchen haben muss, um eine physikalische Aussage zu liefern.

Auf dem HPC Cluster Mogon der Johannes Gutenberg-Universität Mainz werden genügend parallele Rechenkerne auf einem Knoten zur Verfügung gestellt, um eine solche triviale Parallelisierung auch im Rahmen von parallelen Simulationen mehrerer unterschiedlicher Parameter zu realisieren. Auf diese Weise können zu einem Parametersatz mehrere Simulationen parallel gestartet und dennoch verschiedene Parametersätze gleichzeitig auf einem Knoten abgearbeitet werden. Die Anzahl an parallelen Prozessen auf einem Knoten beträgt dabei maximal 64.

Um kleine Systeme berechnen zu können, muss eine Mindestgröße bestimmt werden, die nicht unterschritten werden darf. Sonst kommt es zu unerwünschten Nebeneffekten, die das Resultat verfälschen können. Diese Effekte werden allgemein als *Finite-Size-Effekte* bezeichnet. Um eine untere Grenze der Simulationsboxlänge angeben zu können, wird in Abbildung 6.10(a) und (b) die untere Grenze der Systemgröße unseres Modellsystems getestet, indem bei konstanter Dichte und gleichbleibender Grundfläche die Länge der Box in y-Richtung geändert wird. Finite-Size-Effekte kommen erst ab einer Boxlänge von



(a) Druck vs. Boxlänge.



(b) Viskosität vs. Boxlänge.

Abb. 6.10: Vergleich von (a) Druck und (b) Viskosität als Funktion der Boxlänge h in y -Richtung nach Relaxation mit Scherrate $\dot{\gamma} = 0.1$. Für verschiedene Dichten $\rho = 0.2, 0.4, 0.6$ und 0.8 wird die y -Dimension der Simulationsbox variiert, während die anderen beiden Dimensionen konstant bei $15 \times 15 \sigma^2$ gehalten werden. Die grau gepunktete Linie entspricht dabei dem Wert des ungesicherten Systems mit den Boxausmaßen $15 \times 10 \times 15 \sigma^3$. Die hier gezeigten Daten sind in [2] veröffentlicht worden.

unter 4σ zum Tragen. Für langreichweitigere Potentiale als das WCA-Potential wird dieser Wert höher sein. Außerdem können in der Nähe eines kritischen Punktes die Teilchen korreliert werden, wodurch die Größe der Box ebenfalls eine Rolle spielt.

Bei einer Dichte von $\rho = 0.8$ weicht der Druck p vom Wert ohne Scherung um etwa 10 % ab. Auch bei der Viskosität ist für eine Dichte von $\rho = 0.8$ eine Abweichung von etwa 10 % zu verzeichnen. Die scheinbar geringere Abweichung bei niedrigeren Dichten lässt sich durch die Wahl der Skala begründen. Auch hier ist die Abweichung etwa 10 %. Da die Finite-Size-Effekte ab einer Größe von $y = 4\sigma$ beginnen, werden die Simulationsboxen zur Sicherheit nicht kleiner als 5σ gewählt. Auf diese Weise sind die Simulationsboxen groß genug, um genügend Statistik zu liefern, ohne derartige Effekte hervorzurufen.

Neben einer unteren Grenze der Höhe ist auch eine untere Grenze der Grundfläche interessant. Bei dieser Untersuchung werden Breite und Tiefe in x - bzw. z -Richtung gleichzeitig variiert, da sonst nicht das gewünschte lineare Scherprofil entstehen kann und die Vergleichbarkeit nicht gewährleistet ist. Die entsprechenden Ergebnisse sind in Abbildung 6.11 dargestellt.

Aus Gründen der Übersichtlichkeit werden lediglich das dichteste und das dünnste System mit einer Dichte von $\rho = 0.4$ und $\rho = 0.8$ aufgetragen. Diese werden mit den zuvor bestimmten Kurven bei der Variation der Höhe in y -Richtung verglichen. Auch hier erkennt man vergleichbare Finite-Size-Effekte bei einer Boxlänge von $L_{x,z} < 4\sigma$. Der Effekt ist beim dichten System mit $\rho = 0.8$ bei der Variation der Grundfläche stärker zu sehen, während der Effekt beim dünneren System mit $\rho = 0.4$ insgesamt quantitativ geringer, relativ gesehen aber ähnlich ausfällt. Hier beträgt die Abweichung für den Druck für beide Dichten bei etwa 30 %.

Auch bei der Wahl der Grundfläche empfiehlt es sich also, bei einer Grundfläche von mindestens $5 \times 5 \sigma^2$ zu arbeiten. Um mehr Statistik zu sammeln, wird auch in Zukunft bei einer Grundfläche von mindestens $10 \times 10 \sigma^2$ simuliert, je nach Problemstellung kann sie auf $15 \times 15 \sigma^2$ erhöht werden.

6.3 FEHLERUNTERSUCHUNG DER VISKOSITÄT

Für die korrekte Bestimmung des Stresstensors bezüglich der Hybridsimulationen ist eine Fehlerabschätzung notwendig. Es genügt dabei, den Fehler der Mikrosimulation in der gleichen Größenordnung des Fehlers der Makrosimulation zu halten. Genauere Ergebnisse würden an dieser Stelle nur unnötigen Rechenaufwand bedeuten. Die Daten werden nachträglich geglättet (Vgl. Kapitel 8.2), sodass der Fehler durch den Fit der mit ihrem eigenen Fehler gewichteten Daten ermittelt wird. Zudem ist zu beachten, dass die Viskosität aus der 1,3-Komponente des Stresstensors σ berechnet wird, indem σ_{13} durch die Scherrate geteilt wird.

Für hohe Scherraten bedeutet dies, dass die Amplitude der Fluktuationen für die Viskosität abnehmen, während die Amplitude in σ_{13} für alle Scherraten etwa gleich hoch ist; lediglich der Wert, um den die Daten streuen, ist etwas höher. Für die Viskosität gilt also, dass eine Scherrate, die um eine Größenordnung größer ist, zu Fluktuationen führt, die eine Größenordnung niedriger sind, wie in Abbildung 6.12 deutlich gezeigt wird. Hier sind die zeitliche Entwicklung von σ_{13} in (a) und η in (b) für die Scherraten $\dot{\gamma} = 0.001$ und 0.1 aufgezeigt. Wie zu erwarten war, sinkt die Amplitude der Viskosität um einen Faktor von 100.

Da die Amplitude geringer wird, minimiert sich bei steigender Scherrate auch der Fehler in der Viskosität, wie in Abbildung 6.12 c) zu sehen. Für die gezeigten Datenpunkte

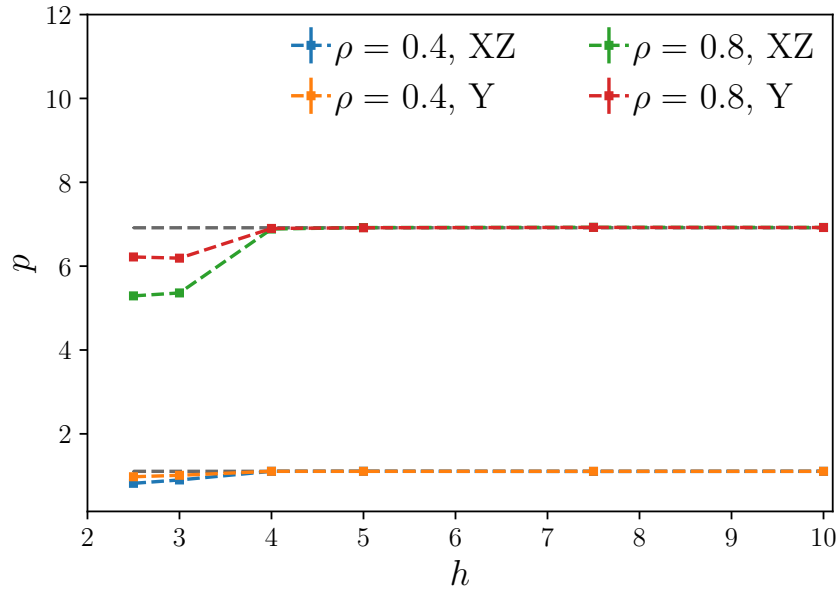
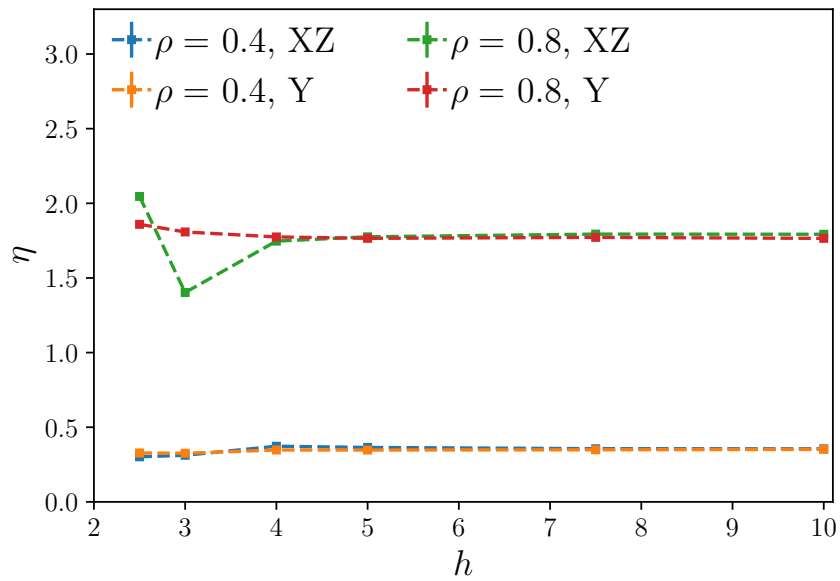
(a) Druck vs. Boxlänge in (x,z) -Richtung.(b) Viskosität vs. Boxlänge in (x,z) -Richtung.

Abb. 6.11: Vergleich von (a) Druck und (b) Viskosität als Funktion der Boxlänge h in x - und y -Richtung nach Relaxation mit Scherrate $\dot{\gamma} = 0.1$. Für verschiedene Dichten $\rho = 0.4, 0.8$ werden hier sowohl die x - als auch die z -Dimension der Simulationsbox variiert, während die Länge in y -Richtung konstant bei 15σ gehalten wird. In der Legende sind die jeweils variierten Längen vermerkt. Die hier gezeigten Daten sind teilweise in [2] veröffentlicht worden.

ist die gleiche Anzahl an MD-Schritten simuliert. Während der Fehler für die Viskosität bei niedrigeren Scherraten zunimmt, wie im eingesetzten Bild zu sehen, bleibt der Fehler für die Stresstensor Komponente σ_{13} konstant gering. Dabei ist ebenfalls schön zu sehen, dass der Wert der Viskosität lediglich leicht schwankt und sich das simulierte Fluid wie ein Newtonsches Fluid verhält.

Die beschriebene Diskrepanz in der Abhängigkeit des Fehlers von der Scherrate führt dazu, dass sich der Rechenaufwand um etwa zwei Größenordnungen erhöht, wenn die Scherrate um eine Größenordnung reduziert wird, um den gleichen statistischen Fehler zu erhalten. Durch die später eingeführte Senkung der Parameterordnung (vgl. Kapitel [8.2](#)) fällt dieser Fehler geringer aus, jedoch sollte diese Abhängigkeit nicht vernachlässigt werden.

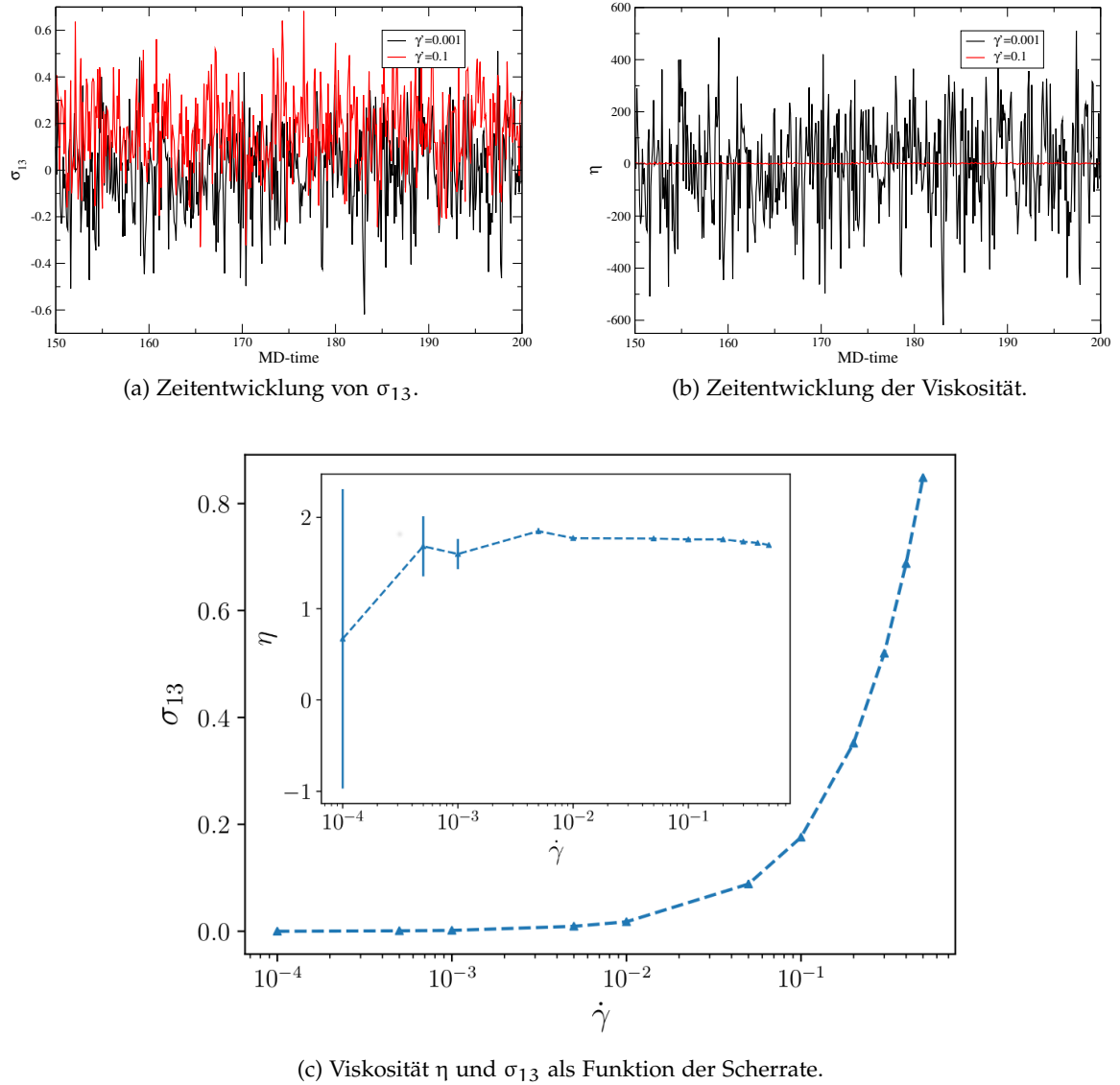


Abb. 6.12: Zeitentwicklung von (a) σ_{13} und (b) η . Bei allen Scherraten haben die Fluktuationen der zeitlichen Entwicklung in σ_{13} in etwa die gleiche Amplitude. Bei der Viskosität sinkt die Amplitude um zwei Größenordnungen, wenn die Scherrate um zwei Größenordnungen erhöht wird. (c) Viskosität η (eingesetztes Bild) und σ_{13} als Funktion der Scherrate. Die Dichte ist $\rho = 0.8$ in einer Simulationsbox mit den Ausmaßen $10 \times 7.5 \times 10 \sigma^3$. Die hier gezeigten Daten sind in [2] veröffentlicht worden.

PARAMETEROPTIMIERUNG DER SIMULATIONEN VON POLYMERKETTEN

In diesem Kapitel werden MD-Simulationen von Polymerketten betrachtet, die wie in Abschnitt 4.2 modelliert sind. Dabei werden Kettenlängen von $N_P = 2, 5, 10$ und 15 unter Scherung für drei verschiedene Monomerdichten $\rho = 0.4, 0.8$ und $\rho = 1.0$ betrachtet. Die Ergebnisse werden mit den in Kapitel 6 vorgestellten Ergebnissen des Monomersystems verglichen.

7.1 EINSCHWINGVORGANG

Da es sich um komplexere Systeme handelt, ist die Relaxationszeit des Systems etwas höher als beim einkomponentigen Kolloidsystem. Auch die Autokorrelationszeit ändert sich. Hier wird ein System mit einer Monomerdichte von $\rho = 0.8$ betrachtet. In einer Simulationsbox mit Volumen $V = 10 \times 10 \times 10 \sigma^3$ entspricht dies bei einer Kettenlänge von $N_P = 10$ genau 80 Polymerketten.

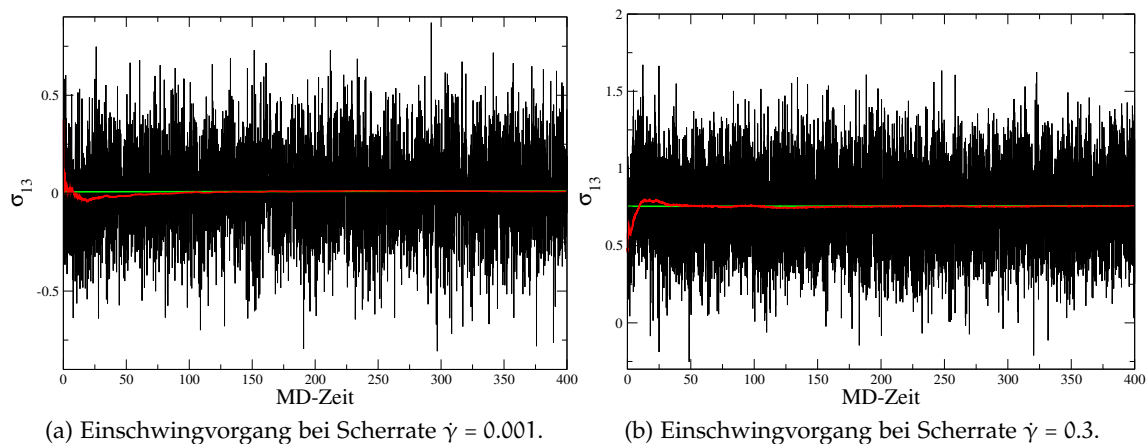


Abb. 7.1: Einschwingvorgang der Größe σ_{13} zur Untersuchung der Zeit bis zum stabilen Zustand bei einer Scherrate von (a) $\dot{\gamma} = 0.001$ und (b) $\dot{\gamma} = 0.3$ für Polymerketten mit einer Kettenlänge von $N_P = 10$ und Monomerdichte $\rho = 0.8$. Die momentanen Werte von σ_{13} sind in schwarz dargestellt, der Durchschnitt zu einer bestimmten MD-Zeit in rot und der Gesamtdurchschnitt in grün. Die stärksten Schwankungen in der roten Kurve sind für beide untersuchten Scherraten nach 250 MD-Zeiten vernachlässigbar.

Wie in Abbildung 7.1 gut zu sehen ist, benötigt die rote Kurve, die den Durchschnittswert von σ_{13} zu einer bestimmten MD-Zeit anzeigt, je nach Scherrate zwischen 200 MD-Zeiten (für $\dot{\gamma} = 0.001$ in (a)) und 150 MD-Zeiten (für $\dot{\gamma} = 0.3$ in (b)), um auf die grüne Kurve des Gesamtdurchschnitts zu konvergieren. Die Relaxationszeit beträgt also etwa 200 MD-Zeiten. Nach dieser Phase reduzieren sich die Schwankungen der Mittelwertskurve für beide untersuchten Scherraten $\dot{\gamma} = 0.001$ und $\dot{\gamma} = 0.3$ und die Kurve beginnt zu konvergieren. Um sicherzugehen, dass der Einschwingvorgang beendet ist, sollte etwa doppelt

solange abgewartet werden, der Beginn der Messungen sollte also nach $t_{\text{MD}} = 200$ bis 400 MD-Zeiten sein.

Um die teilchenbasierten Simulationen der Polymerketten für die Hybridsimulation zu optimieren, empfiehlt es sich also, den Einschwingvorgang als länger anzunehmen und zusätzlich den maximal erlaubten Fehler der MD-Simulationen zu Beginn der Hybridsimulationen zu erhöhen, um die Ergebnisse anschließend durch die im Kapitel 8 vorgestellten Methoden zu verfeinern.

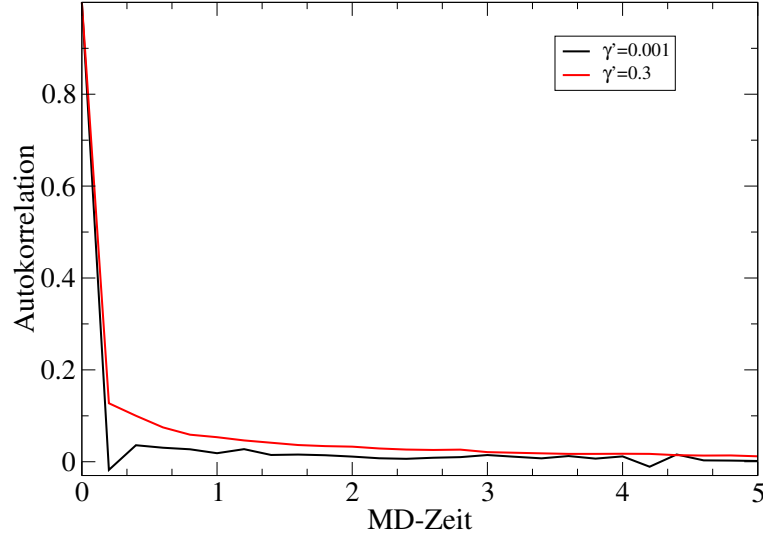


Abb. 7.2: Autokorrelationsfunktion der Polymerketten bei einer Monomerdichte von $\rho = 0.8$ und einer Kettenlänge von $N_P = 10$. Die Funktionen sind für eine Scherrate von $\dot{\gamma} = 0.001$ in schwarz und für $\dot{\gamma} = 0.3$ in rot aufgenommen. Der Zeitschritt ist auch hier $\Delta t = 10^{-3}$.

Auch bei der Aufnahme der Daten der Polymerketten muss die Autokorrelationszeit berücksichtigt werden. In Abbildung 7.2 sind exemplarisch die Autokorrelationen von σ_{13} für eine Scherrate von $\dot{\gamma} = 0.001$ und $\dot{\gamma} = 0.3$ gegen die MD-Zeit aufgetragen. Die Dichte des Systems liegt dabei wie oben erwähnt bei einer Monomerdichte von $\rho = 0.8$. Die Kettenlänge beträgt $N_P = 10$.

Ab einer MD-Zeit von $t_{\text{MD}} = 0.06$ ist die Autokorrelation für dieses System abgeklungen. Um sicher zu gehen, wird die Zykluszeit, also die Zeit, nach der Daten gesammelt wird, für Polymerketten auf $\Delta t_{\text{Mess}} = 0.1$ gesetzt. Da im Normalfall dünnere Systeme (als Beispielsystem wird auch hier ein System mit Monomerdichte $\rho = 0.8$ behandelt) mit Kettenlänge $N_P = 10$ betrachtet werden, ist diese Zeit für die Datennahme sinnvoll.

7.2 SCHERPROFILE

Im Gegensatz zu einem System aus reinen Monomeren, die sich dem Scherprofil problemlos und individuell anpassen können, versuchen sich bei Polymerketten die einzelnen Kettenmonomere der Geschwindigkeit der jeweiligen Schicht anzupassen, in der sie sich befinden. So kann sich der Anfang der Kette in der Mitte der Simulationsbox befinden, also eine Schergeschwindigkeit von $v_S = 0$ besitzen, während das Ende der Kette am oberen Rand der Simulationsbox die maximale Schergeschwindigkeit von $v_S = \dot{\gamma} \cdot L_z$ erfährt. Je nach Lage des Schwerpunkts der Polymerkette ist die Geschwindigkeit dieser Schicht ebenfalls zu berücksichtigen, sodass sich einzelne Monomere der Polymerkette entgegen

der eigentlich vorherrschenden Schergeschwindigkeit der Schicht bewegen können, da der Schwerpunkt der Kette mitbewegt wird. Je dichter die Systeme, desto wichtiger wird es, die Scherprofile zu kontrollieren.

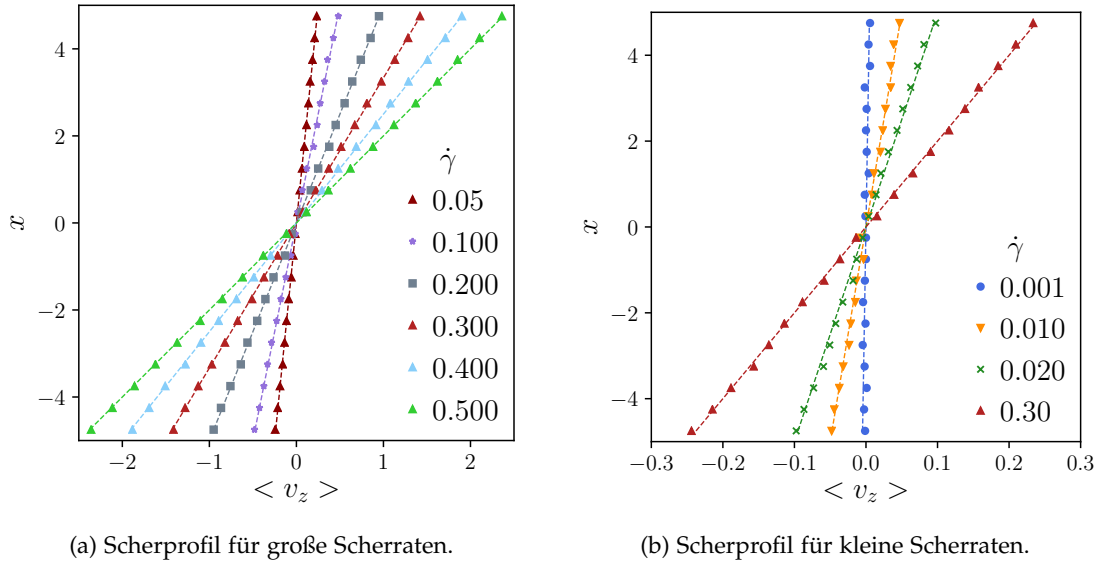


Abb. 7.3: Scherprofile eines Systems mit 80 Ketten, bestehend aus je 10 Teilchen in einer Box von $V = 10 \times 10 \times 10 \sigma^3$ ($\rho = 0.8$). Aufgrund der hohen Dichte benötigt dieses Systems länger als das Einteilchensystem, das im vorherigen Kapitel 6 betrachtet wurde, um ein lineares Scherprofil auch für kleine Scherraten auszubilden. Um die Daten aufzunehmen, wurde ein System unterschiedlichen Scherraten geschert. Nach einer Einschwingphase von etwa 200 MD-Zeiten beginnt die Datennahme. Die Daten, um das Scherprofil letztendlich zu bestimmen, werden anschließend über einen Zeitraum von 1000 MD-Zeiten aufgenommen.

Bei einer Monomerdichte von $\rho = 0.8$ ist eine Einschwingphase von 200 MD-Zeiten absolut ausreichend, um ein Scherprofil einzustellen, das nach 1000 MD-Zeiten gemessen wird. Wie in Abbildung 7.3 b) zu sehen, können mit diesen Zeiten auch kleine Scherraten eingestellt werden. Da einzelne Kettenteile durch die Schwerpunktsbewegung der Kette mitgezogen werden, kann es dazu kommen, dass die Statistik für diesen Fall etwas höher sein muss, um ein exaktes lineares Profil zu erhalten. Bei sehr hohen Kettenlängen und hohen Scherraten führt diese Schwerpunktsbewegung zu einer Orientierung entlang des Scherflusses [66, 67], die Polymerketten liegen innerhalb einer Schicht, wie in Abbildungen 7.8 dargestellt.

Die in Abbildung 7.3 dargestellten Scherprofile werden für ein System mit Boxlänge 10σ in allen Raumrichtungen aufgenommen. 80 Polymerketten mit einer Kettenlänge von $N_p = 10$ Teilchen werden unterschiedlich stark mit Scherraten von $\dot{\gamma} = 0.001$ bis $\dot{\gamma} = 0.5$ geschert.

Das System wird mit unterschiedlichen Scherraten geschert, die Datennahme beginnt ab einer Einschwingphase von 200 MD-Zeiten. Dabei ist ersichtlich, dass mithilfe des SLLOD-Algorithmus und dem isokinetischen Thermostat die hier dargestellten Zeiten auch für kleine Scherraten ausreichen, um ein lineares Profil einzustellen. Bei hohen Scherraten stellt sich das Scherprofil wesentlich schneller ein.

7.3 TEMPERATURVERLAUF DER POLYMERKETTEN

In diesem Abschnitt wird der Temperaturverlauf der Simulationen für Polymerketten betrachtet. Dies ist notwendig, um auszuschließen, dass die Kettenlänge der Polymere einen Einfluss auf die Temperatur des Systems hat.

Die hier vorgestellten Ergebnisse sind bei einer Monomerdichte von $\rho = 0.8$ aufgenommen. Die Simulationsbox hat eine Ausdehnung von $V = 10 \times 10 \times 10 \sigma^3$ mit 80 Polymerketten mit Kettenlänge $N_P = 10$.

Das System wird mit unterschiedlichen Scherraten geschert und dabei mithilfe des isokinetischen Thermostats gekühlt. Zur korrekten Einstellung der Scherprofile wird der SLLOD-Algorithmus zu Hilfe genommen.

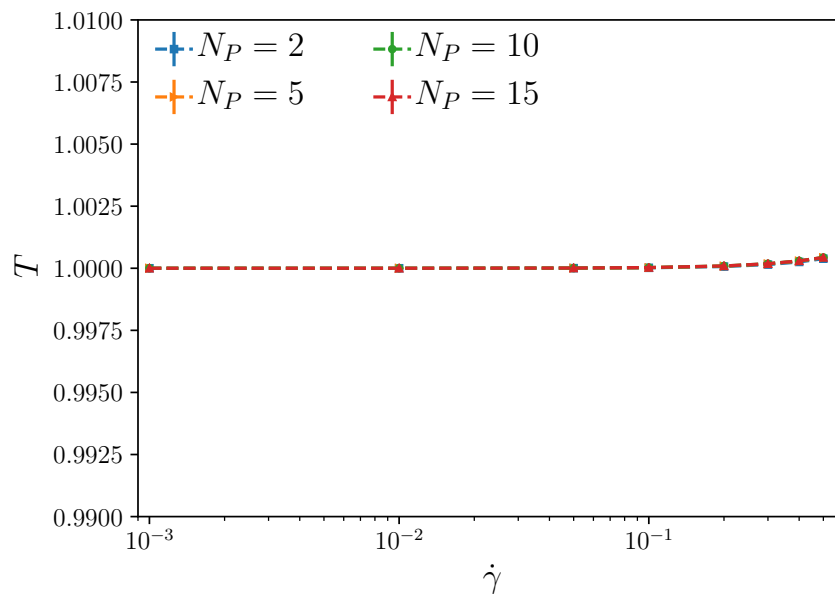


Abb. 7.4: Temperaturverlauf für Polymerketten unterschiedlicher Kettenlänge N_P in Abhängigkeit der Scherrate $\dot{\gamma}$. Wie bei den Monomersystemen ist auch hier lediglich eine minimale Temperaturerhöhung durch erhöhte Scherung zu verzeichnen. Der maximale Temperaturanstieg beträgt dabei etwa 0,01 %.

In Abbildung 7.4 ist der Temperaturverlauf für Kettenlängen $N_P = 2, 5, 10$ und 15 in Abhängigkeit der Scherrate dargestellt. Wie bei den Monomersystemen auch ist lediglich ein Temperaturanstieg von etwa 0,01 % für hohe Scherraten $\dot{\gamma} \geq 0.4$ zu verzeichnen. Zudem erkennt man, dass die Temperatur nicht durch die Kettenlänge beeinträchtigt wird, die einzelnen Kurven liegen gut übereinander. Das isokinetische Thermostat kann also auch für Polymerketten eingesetzt werden.

7.4 STRESSTENSOR UNTER SCHERUNG

Nach der Untersuchung der Einschwingvorgänge, wie Relaxationszeit und Autokorrelation, und der Einstellung der Scherprofile, sowie der Untersuchung einer möglichen Abhängigkeit der Temperatur von der Kettenlänge kann nun der Stresstensor für unterschiedliche Systeme untersucht werden. Dazu werden sowohl unterschiedliche Kettenlängen bei konstanter Dichte verglichen, als auch Systeme konstanter Kettenlänge bei unterschiedlichen Dichten.

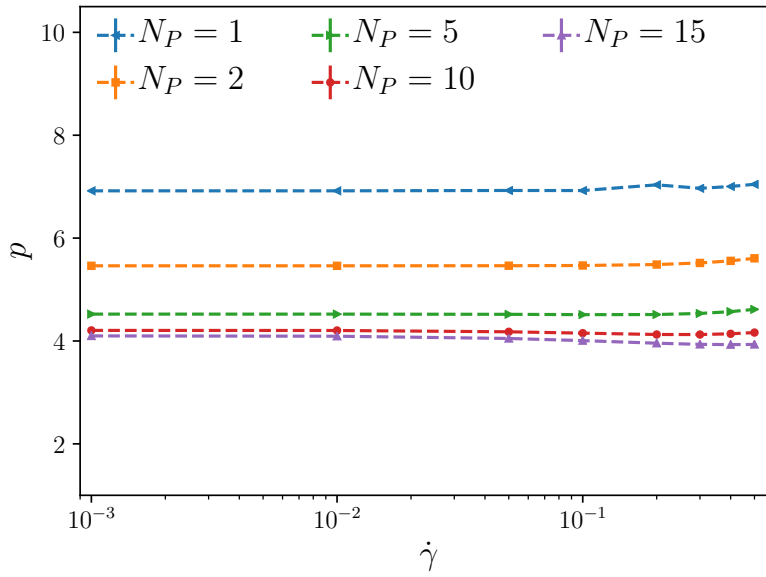
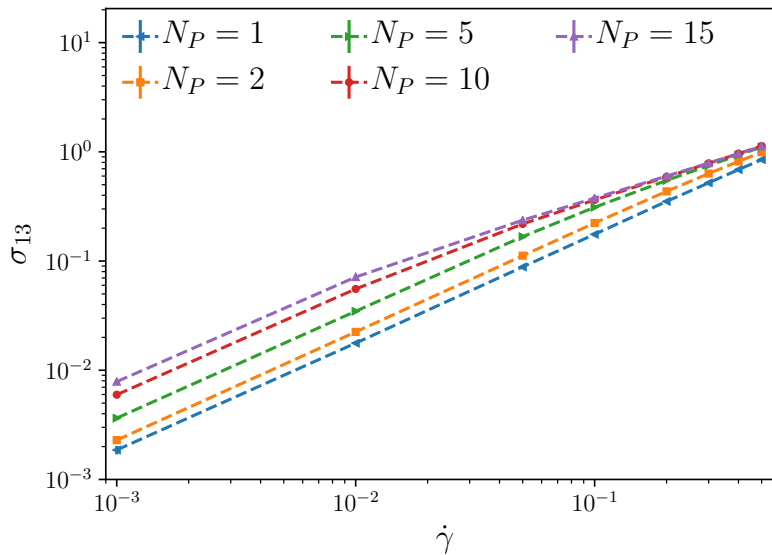
(a) p für verschiedene Kettenlängen.(b) σ_{13} für verschiedene Kettenlängen.

Abb. 7.5: (a) Druck p und (b) Stresstensor Komponente σ_{13} in Abhängigkeit der Scherrate $\dot{\gamma}$ für ein System konstanter Dichte ($\rho = 0.8$) für verschiedene Kettenlängen $N_P = 1, 2, 5, 10$ und 15 . Man sieht deutlich, dass der Druck mit zunehmender Kettenlänge sinkt. Grund hierfür ist, dass durch die Bindung der einzelnen Monomere mehr Freiraum gewonnen wird. Sowohl im Druck, als auch in der σ_{13} -Komponente des Stresstensors ist eine Änderung des Verhaltens verglichen mit den in blau eingezeichneten Daten des reinen Monomersystems ersichtlich. Eine Scherratenabhängigkeit ist nur für große Scherraten $\dot{\gamma}$ zu erkennen. Der Wert, ab dem dieser Wert eintritt, nimmt für zunehmende Kettenlänge ab. Für $N_P = 10$ ist der Wert $\dot{\gamma} > 0.001$, für $N_P = 2$ ist dieser Wert erst bei $\dot{\gamma} > 0.2$. Der Druck weist dabei lediglich eine leichte Tendenz auf, während die Kurven der Stresstensor Komponente σ_{13} auf die Kurve des Monomersystems konvergieren.

Die Systemparameter sind wie zuvor eine Simulationsbox mit $V = 10 \times 10 \times 10 \sigma^3$. Hier werden bei konstanter Dichte die Ergebnisse bei einer Monomerdichte von $\rho = 0.8$ präsentiert, was in diesem Fall 800 Monomeren entspricht. Die Anzahl an Polymeren variiert mit der Kettenlänge N_P von $\rho_P = 400$ bei einer Kettenlänge von $N_P = 2$ bis zu $\rho_P = 80$ bei einer Kettenlänge von $N_P = 10$.

Bei einer Kettenlänge von $N_P = 15$ wird eine größere Simulationsbox verwendet, um zu verhindern, dass Anfangs- und Endmonomer einer Polymerkette über die periodischen Ränder hinaus wechselwirken und somit das Polymer mit sich selbst interagiert. In diesem Fall ist das Volumen der Simulationsbox $V = 15 \times 15 \times 15 \sigma^3$ mit einer Anzahl von 2700 Monomeren bei einer Monomerdichte von $\rho = 0.8$.

Die Ergebnisse zu einer konstanten Monomerdichte von $\rho = 0.4$ sind im Anhang in Abschnitt A.2.1 zu finden.

In Abbildung 7.5 werden zunächst der Druck und die σ_{13} -Komponente des Stresstensors betrachtet. Dabei ist leicht zu erkennen, dass der Druck mit steigender Kettenlänge sinkt. Dieses Verhalten ist zu erwarten, da durch die Einschränkung der Bewegung der Monomere sowie ihre relative Nähe zueinander weniger Druck auf die Außenfläche der Simulationsbox aufgebaut werden kann.

Auf der doppel-logarithmischen Skala in Abbildung 7.5 b) sieht man deutlich die Abhängigkeit von σ_{13} von der Scherrate $\dot{\gamma}$, die dadurch in Abschnitt 8.2 gut durch eine Linearkombination aus Polynomen angenähert werden kann. Für Monomere ist die Kurve auf der doppel-logarithmischen Skala annähernd linear. Außerdem sind die Kurven der einzelnen Kettenlängen für σ_{13} bei niedrigen Scherraten annähernd parallel, wodurch sie sich nur durch einen Vorfaktor bei dieser Annäherung unterscheiden werden. Für hohe Scherraten nähern sich alle Kurven der in blau gehaltenen Linie des Einteilchensystems an.

Sowohl beim Druck p als auch bei σ_{13} sieht man einen anderen Grundwert. Für σ_{13} ist zudem eine Änderung des Verhaltens und damit eine Abhängigkeit von der Scherrate im Gegensatz zum Monomersystem zu verzeichnen. Hier konvergieren die Kurven zur Kurve des Monomersystems. Dies ist in der nachfolgenden Betrachtung der Viskosität sehr viel deutlicher zu erkennen.

In Abbildung 7.6 ist die Viskosität η für Polymerketten mit unterschiedlichen Parametern aufgetragen. Die Abhängigkeit der Viskosität von der Kettenlänge ist in Abbildung 7.6 a) dargestellt. Hier werden Systeme mit einem Volumen von $V = 10 \times 10 \times 10 \sigma^3$ simuliert. Die Kettenlängen der hier untersuchten Systeme betragen $N_P = 2, 5, 10$ und 15.

Um auszuschließen, dass einzelne Kettenglieder über die periodischen Ränder hinaus miteinander wechselwirken, muss für eine Kettenlänge von $N_P = 15$ die Boxlänge wieder in alle Dimensionen auf 15σ erweitert werden. Die Dichte der in a) betrachteten Systeme ist $\rho = 0.8$, was bei niedrigen Kettenlängen zu einer Monomerzahl von 800 Teilchen führt. Die Anzahl an Polymerketten ist dabei 400, 160 bzw. 80. Bei dem größeren System ergibt sich eine Monomerzahl von 2700 Teilchen und damit 180 Polymerketten.

Zum besseren Vergleich wird in Abbildung 7.6 a) die Viskosität in Abhängigkeit der Scherrate $\eta(\dot{\gamma})$ für Monomere ebenfalls eingezeichnet. Man erkennt deutlich, dass eine Verbindung von Monomeren zu Ketten aufgrund der höheren potentiellen Energie zu einer Erhöhung der Viskosität führt. Bereits Hantelsysteme, also ein System mit Ketten einer Kettenlänge von $N_P = 2$ (auch als Dimere bezeichnet), weisen eine erhöhte Viskosität auf. Dieser Effekt ist deutlicher, je mehr Monomere miteinander verbunden werden. Außerdem kommt es bereits für ein Hantelsystem bei hohen Scherraten zum sogenannten *Shear-Thinning*. Dieser Effekt ist der Grund für die Betrachtung dieser Polymerkettensysteme.

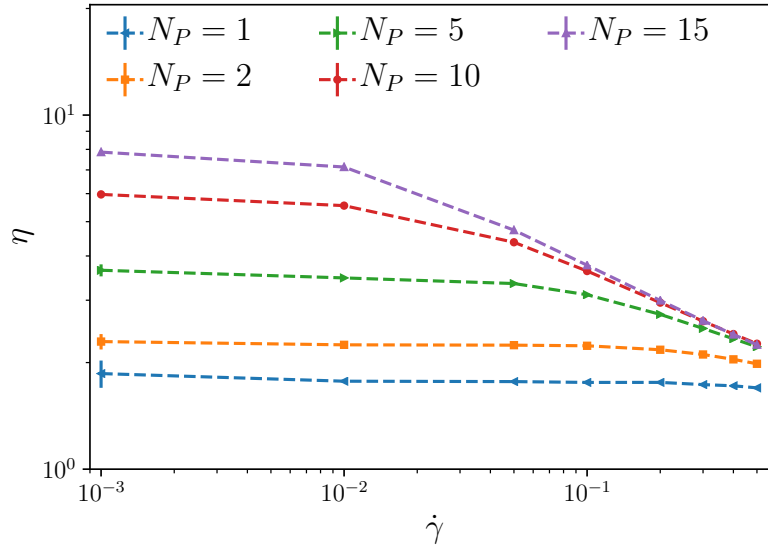
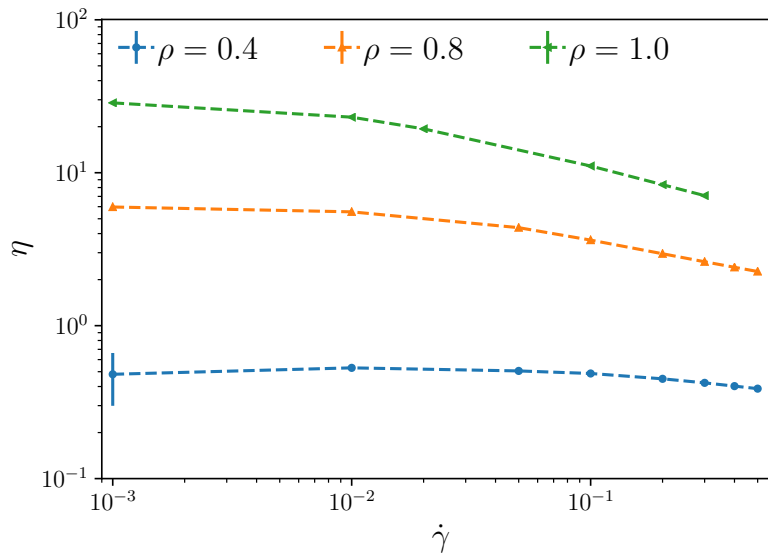
(a) η für verschiedene Kettenlängen.(b) η für verschiedene Dichten.

Abb. 7.6: (a) η in Abhängigkeit der Scherrate für ein System konstanter Dichte ($\rho = 0.8$) für verschiedene Kettenlängen $N_P = 1, 2, 5, 10$ und 15 und (b) η für eine konstante Kettenlänge von $N_P = 10$ und verschiedene Dichten $\rho = 0.4, 0.8$ und 1.0 . Die Viskosität weist bereits für Hanteln, mit einer Kettenlänge von $N_P = 2$, das erhoffte *Shear Thinning* auf, wie in (a) zu sehen. Der Effekt wird durch längere Ketten verstärkt. Auch die Dichte beeinflusst das *Shear Thinning*. Je dichter das System ist, desto stärker wird es ausgedünnt und desto weniger stark muss die angelegte Scherung sein. Bei einer Dichte von $\rho = 0.8$ beginnt der Effekt bei $\dot{\gamma} = 0.01$, bei $\rho = 0.4$ erst bei einer Scherrate von $\dot{\gamma} = 0.1$.

me im Rahmen der Hybridsimulationen. Dabei handelt es sich um einen Effekt, bei dem, wie in Abbildung 7.6 a) zu sehen, das System bei zunehmender Scherrate ausdünnt und weniger viskos wird. Der Effekt nimmt mit wachsender Kettenlänge N_P zu. Während dieser Effekt bei Hanteln erst ab einer Scherrate von $\dot{\gamma} = 0.2$ auftritt und nur minimal ausgeprägt scheint, ist er bei einer Kettenlänge von $N_P = 10$ schon ab einer Scherrate von $\dot{\gamma} = 0.01$ deutlich zu sehen. Auch der Abfall der Kurve selbst nimmt mit steigender Kettenlänge zu.

In Abbildung 7.6 b) wird die Viskosität für eine konstante Kettenlänge von $N_P = 10$ bei unterschiedlichen Dichten betrachtet. Die behandelten Dichten sind $\rho = 0.4, 0.8$ und $\rho = 1.0$. Da es sich um die Monomerdichte handelt und aufgrund des zusätzlichen FENE-Potentials die einzelnen Monomere dichter beieinander sind, sind solche Werte möglich, ohne dass die Systeme eine kristalline Struktur aufzuweisen. Das System mit der hohen Monomerdichte wurde verwendet, um die korrekte Funktionsweise des Programms durch einen Vergleich mit bereits publizierten Werten [68] zu überprüfen. Der Vergleich befindet sich im Anhang unter Abschnitt A.1. Ebenfalls im Anhang befindet sich der Vergleich unterschiedlicher Kettenlängen für ein System mit einer Monomerdichte von $\rho = 0.4$, vgl. hierzu Abbildung A.4.

Der erhoffte scherinduzierte Ausdünnungsprozess weist eine Dichtenabhängigkeit auf. So beginnt der Prozess für eine Monomerdichte von $\rho = 0.8$ bereits ab einer Scherrate von $\dot{\gamma} = 0.01$. Der Effekt liegt bei einer Monomerdichte von $\rho = 1.0$ sogar bei noch kleineren Werten für die Scherrate, bei denen keine Daten mehr aufgenommen wurden. Bei einer Dichte von $\rho = 0.4$ wird der Effekt erst ab einer Scherrate von $\dot{\gamma} = 0.1$ signifikant.

Ein weiterer interessanter Zusammenhang ist die Abhängigkeit der Viskosität η von der inversen Kettenlänge N_P , um eine Konvergenz bei hohen Scherraten $\dot{\gamma} \geq 0.1$ abschätzen zu können.

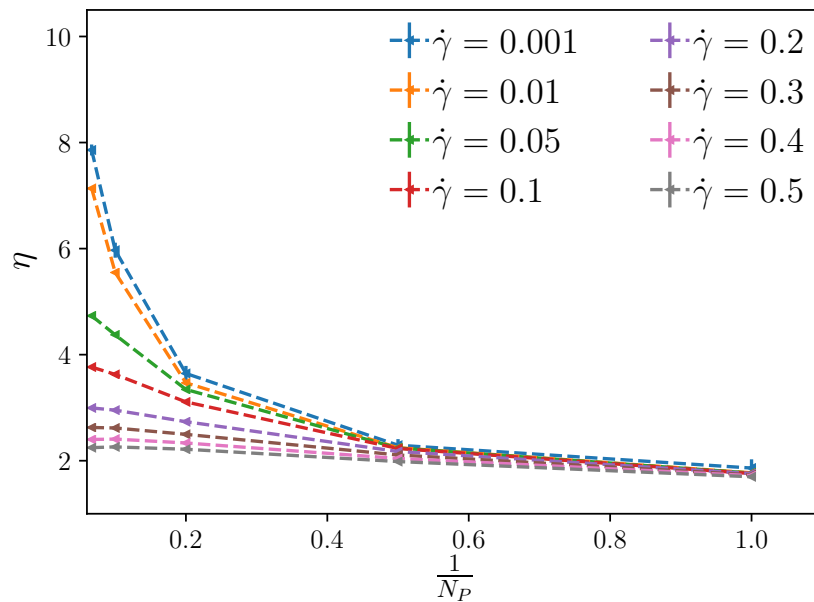


Abb. 7.7: Viskosität η der Polymerkettensysteme in Abhängigkeit der inversen Kettenlänge für verschiedene Scherraten bei einer Monomerdichte von $\rho = 0.8$. Die Werte konvergieren für hohe Scherraten zunehmend auf denselben Wert. Für niedrige Scherraten wächst der Wert exponentiell für wachsende Kettenlängen.

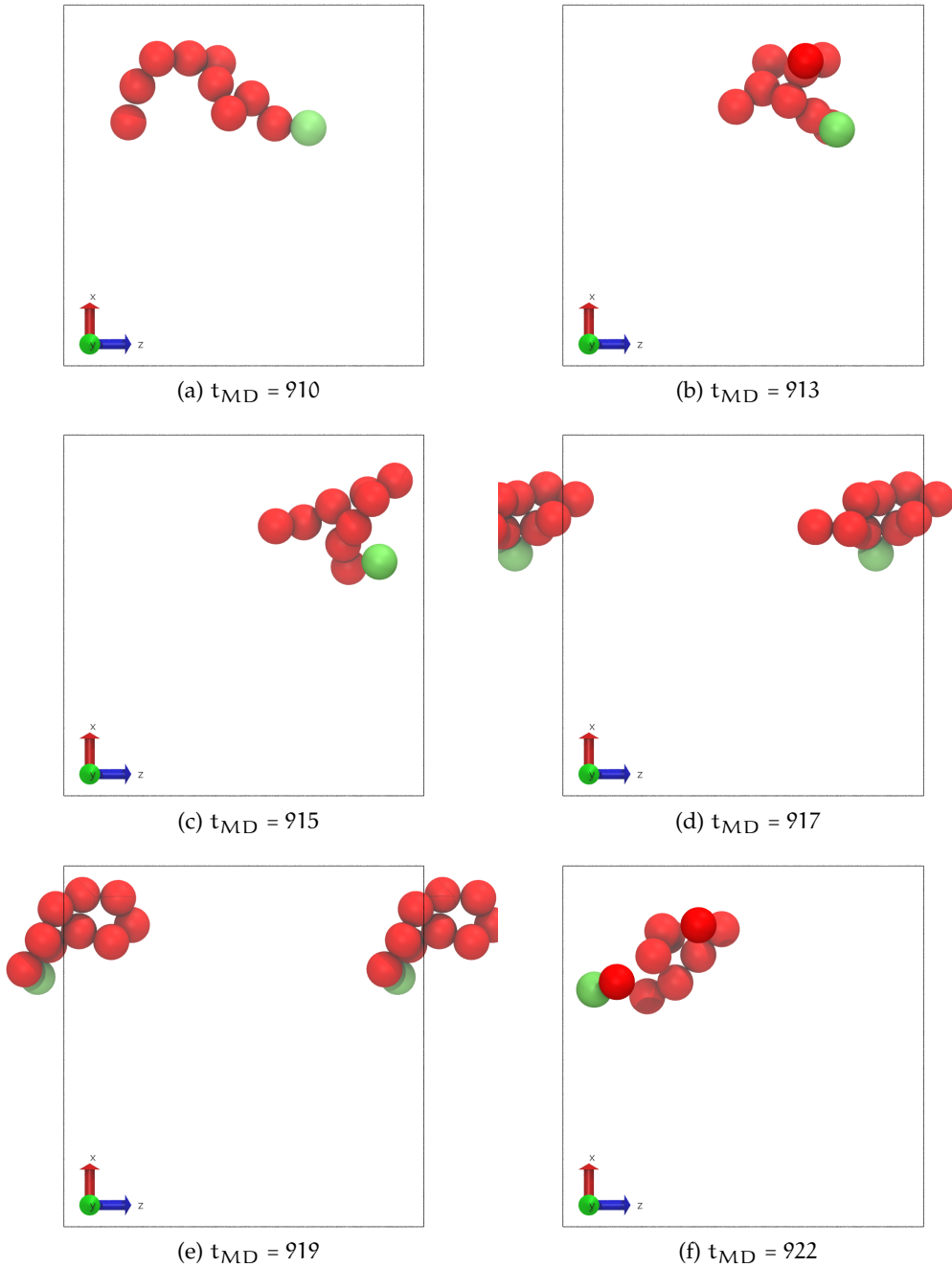


Abb. 7.8: Zeitentwicklung in Form von Momentaufnahmen einer einzelnen Polymerkette nach einer Simulationsdauer von 900 MD-Zeiten. Die simulierte Kette wird im Abstand von wenigen MD-Zeiten betrachtet, um das Verhalten im Fluss zu veranschaulichen. Dabei ist eine der beiden Endmonomere in grün dargestellt, um die Zeitentwicklung besser nachvollziehen zu können. In (d) und (e) wurden die periodischen Randbedingungen in z -Richtung angewendet, um die Kette im Ganzen betrachten zu können und den Übergang von einer Seite der Simulationsbox zur nächsten einfacher nachvollziehen zu können. Die Polymerkette hat sich in (a) der Strömung angepasst und in Scherrichtung ausgerichtet. Sie liegt zunächst als annähernd ausgestreckte Kette vor, die innerhalb einer Schicht liegt. Aufgrund einer Vortizität dreht sich die Kette innerhalb dieser Schicht immer wieder um ihren Schwerpunkt, wie in den nachfolgenden Bildern anhand einer solchen Drehung veranschaulicht wird. Dies wird als sogenanntes „Tumbling“ (deutsch: Taumeln oder Straucheln) bezeichnet. Die eingestellte Scherrate beträgt $\dot{\gamma} = 0.3$, die Monomerdichte ist $\rho = 0.8$. Die Polymerkette bewegt sich im Verlauf der Bilderserie in positiver z -Richtung. Die Bilder wurden mithilfe von vmd [4, 5] erzeugt.

Dieser Zusammenhang ist in Abbildung 7.7 dargestellt. Für hohe Scherraten ist ersichtlich, dass die Simulationen auf einen Wert konvergieren. Entsprechend scheint die Viskosität nicht weiter als bis zur Viskosität des Monomersystems abzusinken, dieser Wert stellt also vermutlich den Grenzwert für die Ausdünnung dar.

Durch Scherung kann es bei langen Ketten und starker Scherung dazu kommen, dass sich die Polymerketten nach einer Relaxationszeit entlang der Strömung ausrichten. Dieser Effekt ist in Abbildung 7.8 exemplarisch gezeigt. Die Serie der Momentaufnahmen entstammt einer Simulation für $\rho = 0.8$ und einer Scherrate von $\dot{\gamma} = 0.3$ für eine Polymerkettenschmelze mit Polymerketten der Länge $N_p = 10$. Dieser Effekt wurde auch für andere Ketten nachvollzogen.

In Abbildung 7.8 ist deutlich zu erkennen, dass sich die Kette zu einem Zeitpunkt τ zwar vollständig innerhalb einer Schicht befindet, die Schergeschwindigkeit v_s für alle Monomere der Kette also annähernd gleich ist, durch eine Vortizität der Scherung die Kette aber beginnt, sich zusammenzuknäueln und zu drehen. Sie liegt also abwechselnd gestreckt und in einem Knäuel vor. Dieser Effekt beruht auf einer Vortizität der Scherung, wie in [69] behandelt. Bei zukünftigen Simulationen komplexer Geometrien wie dem 4 – 1- oder dem H-Kanal wird ein neuer Algorithmus der MD-Simulationen für Polymerketten benötigt, der eine reine Scherung und Kombinationen aus reiner und simpler Scherung simulieren kann, wie in [69, 70] vorgestellt. Bei reiner Scherung kann die Viskosität der Polymerketten zunehmen und zu einem Shear Thickening führen [71].

Um den Effekt der scherinduzierten Ausdünnung als konkreten Kontrast zu den Newtonschen Monomeren, die mithilfe des WCA-Potentials simuliert werden, in den Hybrid-simulationen zu untersuchen, empfiehlt es sich insgesamt, lange Polymerketten bei relativ hohen Dichten zu betrachten. Da der Effekt vor allem bei relativ hohen Scherraten auftritt, empfiehlt es sich aufgrund der in Abschnitt 8.2 vorgestellten Methoden der Senkung der Parameterordnung und den Fitverfahren aus Abschnitt 8.2.2, schon zu Beginn der Simulation Daten des Stresstensors unter starker Scherung zur Verfügung zu stellen.

Teil IV

KOPPLUNG

Ziel dieser Arbeit ist, ein für Hybridsimulationen optimiertes, teilchenbasiertes Simulationsprogramm zu entwickeln. Um eine Hybridsimulation zu erreichen, werden die Makro- und die Mikro-Skala am Ende der Optimierung miteinander gekoppelt. Dies geschieht zunächst offline, indem ausführliche MD-Simulationen zu den zuvor bekannten, zu untersuchenden Parametern gestartet werden. Die erhaltenen Daten werden zunächst in Form einer Datei gespeichert und der Makro-Ebene, welche über eine CFD-Simulation realisiert wird, zur Verfügung gestellt. In derzeitig entwickelten Verfahren werden die Daten direkt an die CFD-Simulation übergeben.

Da die beiden Programme zunächst nicht direkt miteinander kommunizieren, handelt es sich in dieser ersten Phase um eine passive Kopplung. Auf diese Weise kann die generelle Funktionsweise der Programme getestet und optimiert werden. Außerdem können Algorithmen entwickelt werden, die den Simulationsaufwand der MD-Simulationen reduzieren und auf diese Weise die gesamte Hybrid-Simulation beschleunigen.

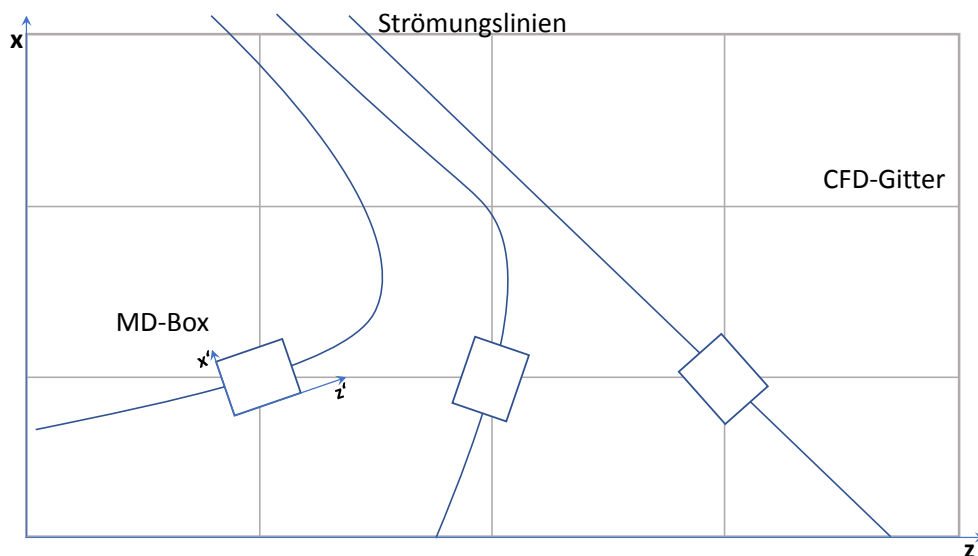


Abb. 8.1: Drehung der MD-Boxen entlang der Strömungslinien. Die Simulationsboxen der MD-Simulation werden dabei entlang der Strömungslinien derart gedreht, dass es zu einem eindimensionalen Scherfluss entsprechend den in Kapitel 2.2 vorgestellten Lees-Edwards-Randbedingungen kommt. Die Koordinatenachsen der Simulationen stimmen also nicht miteinander überein.

So werden zum Beispiel die MD-Simulationsboxen entlang der Strömungslinien gedreht, wie in Abbildung 8.1 gezeigt, sodass stets ein eindimensionaler Fluss entsteht. Dies wird auch in Abschnitt 8.1 näher beschrieben.

Nach den Optimierungen und der passiven Kopplung, um die generelle, korrekte Funktionsweise des Hybridschemas zu demonstrieren, wird schließlich ein Interface entwickelt,

welches beide Programme ansprechen und Informationen in Form von Startparametern der MD-Simulation bzw. Daten der sechs unabhängigen Stresstensorkomponenten σ_{ij} von einer Simulation zur anderen übergeben kann. Diese aktive Kopplung ist letztendlich das Ziel des Projekts. Das Interface selbst kann dabei die beiden Programme jeweils als Black-box ansprechen, sodass die beiden Elemente beliebig ausgetauscht und optimiert werden können, sofern die generelle Funktionsweise gleich bleibt, bzw. nur minimale Änderungen vorgenommen werden müssen.

Um die notwendigen MD-Simulationen so ressourcensparend wie möglich zu gestalten, werden verschiedene mathematische Verfahren verwendet, um die Daten zu optimieren. Dazu zählen ein Ansatz, der die Ordnung der Parameter senkt (engl. *Reduced-order approach*), sowie eine Datenverfeinerung (engl. *Data refinement strategy*), sodass am Ende eine Hybrid-Simulation mit reduzierter Komplexität entsteht, wie sie in [1] vorgestellt wurde. Dadurch reduziert sich die Anzahl der Simulationen auf im ersten Schritt vier Parametersätze zu Beginn, in jedem weiteren Schritt wird nach Möglichkeit nur ein weiterer Parametersatz begonnen.

In den nachfolgenden Abschnitten werden die erwähnten Methoden vorgestellt. Auch auf die passive und die aktive (online) Kopplung wird dabei eingegangen. Die diesen Kapiteln zugrundeliegenden Theorien und Algorithmen wurden von Dr. Nehzat Emany und Dr. Leonid Yelash entwickelt. Mithilfe einer von Dr. Leonid Yelash ausgearbeiteten MatLab-Routine [72] wurden die in [2] vorgestellten Routinen anhand von neuen Daten in drei Dimensionen nachvollzogen.

8.1 DREHUNG DER SIMULATIONSBOXEN

An jedem Quadraturpunkt des Gitters in der Makro-Simulation, also denjenigen Punkten, an denen die Navier-Stokes-Gleichungen bestimmt werden, werden wie in Abschnitt 5.2.1 gezeigt Informationen der Mikro-Ebene benötigt. Um die Komplexität der benötigten MD-Simulation so gering wie möglich zu halten, werden die Simulationsboxen wie in Abbildung 8.1 dargestellt entsprechend der Flussrichtung gedreht, sodass ein eindimensionaler Fluss bezüglich der in Abschnitt 2.2 eingeführten Lees-Edwards-Randbedingungen entsteht. Auf diese Weise reduziert sich bereits die Anzahl an notwendigen Mikro-Simulationen, da der Phasenraum nur von einem Parameter abhängt, der Scherrate $\dot{\gamma}$.

Da die mit der Simulation ermittelten Daten nun für einen eindimensionalen Fluss bestimmt sind, müssen die Daten anschließend rücktransformiert werden. Der Austausch der Informationen ist in Abbildung 8.2 dargestellt. Dabei wird der Verzerrungsgeschwindigkeitstensor E (in dem die Scherrate enthalten ist) über die Drehmatrix Θ modifiziert und mit der charakteristischen Länge l_b^* gewichtet. Diese charakteristische Länge ist im Standardfall die Länge der Simulationsbox.

Der modifizierte Verzerrungsgeschwindigkeitstensor E^* wird an die MD-Simulation in Form der Scherrate $\dot{\gamma}$ übergeben. Die Simulation wird für eine bestimmte Dichte ρ durchgeführt. Da in dieser Phase des Projekts inkompressible Flüssigkeiten betrachtet werden, ist die Dichte einer Gesamtsimulation konstant.

Nach einer Einschwingphase der Mikro-Simulation wird der Stresstensor σ^* über eine gewisse Zeit hinweg bestimmt und anschließend rücktransformiert, um an die Makro-Skala zurückgegeben zu werden. Die Zeit, über die der Stresstensor gemittelt bzw. aufgenommen wird, hängt dabei von der zugrundeliegenden Problemstellung, bzw. Phase des Projekts ab. In der Offline-Phase wird sie möglichst lange angesetzt, um eine Datenbasis

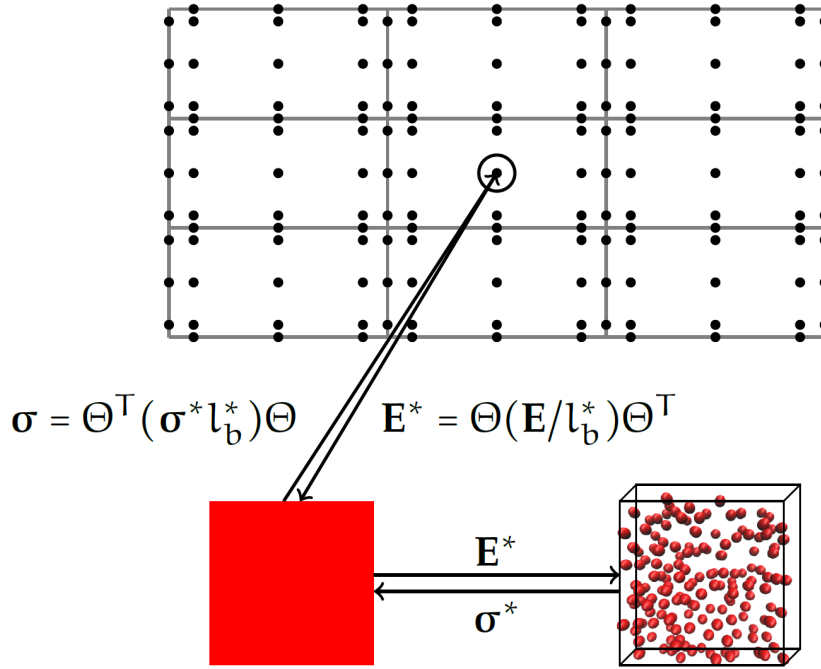


Abb. 8.2: Datentransfer im vorgestellten Hybridsimulationsschema. Hier ist das gesamte Simulationsgebiet Ω der Makro-Simulation dargestellt. Mithilfe eines Gitters wird das Gebiet in kleinere Zellen aufgeteilt. Die schwarzen Punkte an und zwischen den einzelnen Gitterelementen entsprechen den Quadraturpunkten, an denen die Navier-Stokes-Gleichungen gelöst werden und die einzelnen MD-Simulationsboxen platziert werden. Θ ist die Rotationstransformation, die zu einem 1D Scherfeld führt. E stellt dabei den Verzerrungsgeschwindigkeitstensor dar und enthält die Information der Scherrate.

zu erzeugen, auf die später zurückgegriffen werden kann. Bei der aktiven Kopplung wird ein Wert für den Fehler als eine Art „Güte“ angegeben, den das Programm erreichen soll.

Der lokale Verzerrungsgeschwindigkeitstensor E enthält die Information der Stärke der Scherung. Es handelt sich dabei um einen symmetrischen, reellen Tensor der Form:

$$E = \begin{pmatrix} \frac{\partial u_x}{\partial x} & \frac{1}{2} \left(\frac{\partial u_x}{\partial y} + \frac{\partial u_y}{\partial x} \right) & \frac{1}{2} \left(\frac{\partial u_x}{\partial z} + \frac{\partial u_z}{\partial x} \right) \\ \frac{1}{2} \left(\frac{\partial u_y}{\partial x} + \frac{\partial u_x}{\partial y} \right) & \frac{\partial u_y}{\partial y} & \frac{1}{2} \left(\frac{\partial u_y}{\partial z} + \frac{\partial u_z}{\partial y} \right) \\ \frac{1}{2} \left(\frac{\partial u_z}{\partial x} + \frac{\partial u_x}{\partial z} \right) & \frac{1}{2} \left(\frac{\partial u_z}{\partial y} + \frac{\partial u_y}{\partial z} \right) & \frac{\partial u_z}{\partial z} \end{pmatrix}. \quad (8.1)$$

Für die rotierten Simulationsboxen wird E derart gedreht, dass alle Elemente außer den Nebendiagonalelementen mit Index 1,3 verschwinden:

$$E^* = \Theta E \Theta^T = \begin{pmatrix} 0 & 0 & E_{13}^* \\ 0 & 0 & 0 \\ E_{31}^* & 0 & 0 \end{pmatrix}. \quad (8.2)$$

Die übrigbleibenden, nicht Null werdenden Nebendiagonalelemente E_{13}^* und E_{31}^* sind aufgrund der Symmetrie des Tensors gleich und bilden die lokale Scherrate für die reine Scherung:

$$\frac{\dot{\gamma}}{2} = E_{13}^* = E_{31}^*. \quad (8.3)$$

Die Drehmatrix Θ wird in Abschnitt 8.3 näher besprochen. Dabei sei angemerkt, dass es sich bei dieser Darstellung um eine reine Scherung handelt. Der in der MD-Simulation angewandte Verzerrungsrate tensor besitzt nur in einer Komponente einen Eintrag ungleich 0. Dieser lässt sich über eine Linearkombination aus einem symmetrischen und einem antisymmetrischen Tensor ausdrücken:

$$\nabla \mathbf{u} = \mathbf{E}' = \underbrace{\begin{pmatrix} 0 & 0 & \frac{\dot{\gamma}}{2} \\ 0 & 0 & 0 \\ \frac{\dot{\gamma}}{2} & 0 & 0 \end{pmatrix}}_{\text{reine Scherung}} + \underbrace{\begin{pmatrix} 0 & 0 & -\frac{\dot{\gamma}}{2} \\ 0 & 0 & 0 \\ \frac{\dot{\gamma}}{2} & 0 & 0 \end{pmatrix}}_{\text{reine Rotation}}. \quad (8.4)$$

Dies ist die Darstellung der einfachen Scherung (engl.: *simple shear*), die als Kombination einer reinen Scherung (engl.: *pure shear* oder *extensional shear*) und einer reinen Rotation (Starrkörperrotation oder *rigid body rotation*, siehe [69]) aufgefasst werden kann. Zur Zeit wird in der MD-Simulation die einfache Scherung realisiert. Diese wird bei Umrechnungen zur Zeit als reine Scherung behandelt.

Mit dieser Scherrate wird eine eindimensionale, teilchenbasierte MD-Simulation mit Lees-Edwards-Randbedingungen in transformierten Koordinaten (x', y') gestartet. Der in diesen Koordinaten erhaltene viskose Spannungstensor τ^* bzw. der Cauchy-Stresstensor σ wird anschließend in die ursprünglichen Koordinaten rücktransformiert:

$$\tau = \Theta^T \tau^* \Theta. \quad (8.5)$$

Dabei sei angemerkt, dass im MD-Programm die Komponenten des Stresstensors $\sigma_{\alpha\beta}$ bestimmt werden, der über (2.35) mit dem Spannungstensor τ zusammenhängt. Außerdem wurde in der Arbeit in den vorherigen Kapiteln σ statt σ^* verwendet, wobei auch hier zur Transformation folgende Gleichung angewendet wird:

$$\sigma = \Theta^T \sigma^* \Theta, \quad (8.6)$$

da zum einen die Unterscheidung noch nicht eingeführt wurde, zum anderen da diese Schreibweise der Übersichtlichkeit dient.

8.2 SENKUNG DER PARAMETERORDNUNG

Die Herangehensweise der nachfolgenden Abschnitte folgt [1]. Dabei wird der Algorithmus zur Senkung der Parameterordnung beschrieben, mit dessen Hilfe die Anzahl an teilchenbasierten Simulationen weiter reduziert wird.

Dazu wird die unbekannte nicht-lineare Funktion des Cauchy-Stresstensors $\sigma^* = f(\mathbf{E}^*)$ optimiert. Der Stresstensor hängt also vom Verzerrungsgeschwindigkeitstensor \mathbf{E} ab. Da die für die hier zugrundeliegende Problemstellung wichtige Scherrate $\dot{\gamma}$ wie in Gleichung (8.3) beschrieben in \mathbf{E} enthalten ist, gilt $f(\mathbf{E}^*) = f(\dot{\gamma})$.

Dabei wird σ^* wie folgt angenähert:

$$\sigma_i^* \approx \tilde{f}_i(\dot{\gamma}) = \sum_{j=0}^k x_{ij} B_j. \quad (8.7)$$

Der Index i entspricht dabei dem i -ten unabhängigen Eintrag des Stresstensors. Aufgrund der Symmetrie des Stresstensors müssen die äquivalenten Komponenten nicht betrachtet

werden. In zwei Dimensionen ist $i = \{1, 2, 3\}$, in drei reduziert sich die Anzahl der Komponenten von 9 auf 6, sodass $i = \{1, \dots, 6\}$. k entspricht dem Grad des Polynoms, mit dem die Funktion genähert wird. In \mathbf{B} ist die Basis enthalten, mit der gefittet wird. So lässt sich das Problem mit Polynomen fitten, die entsprechende Basis aus Monomen hat die Form:

$$\mathbf{B}(\dot{\gamma}) = \{1, \dot{\gamma}, \dot{\gamma}^2, \dot{\gamma}^3, \dots\} . \quad (8.8)$$

Für die später durchgeführte Chebyshev¹-Näherung werden Chebyshev-Polynome erster Art verwendet. In diesem Fall hat \mathbf{B} die Form:

$$\begin{aligned} \mathbf{B}(\dot{\gamma}) &= \{T^0(\dot{\gamma}), T^1(\dot{\gamma}), T^2(\dot{\gamma}), T^3(\dot{\gamma}), \dots\} , \\ &= \{1, \dot{\gamma}, 2\dot{\gamma}^2 - 1, 4\dot{\gamma}^3 - 3\dot{\gamma}, \dots\} . \end{aligned} \quad (8.9)$$

Für die Offline-Trainingsphase wird ein Problem der kleinsten Quadrate mit der sogenannten Tikhonov-Regularisierung für jede Komponente i gelöst. Diese Regularisierung berücksichtigt bei iterativen Verfahren einen „Strafterm“, der den Fehler der Iteration ausgleichen soll. Die zu fittenden Parameter x_{ij} werden dabei in einem Vektor \mathbf{x} zusammengefasst:

$$\arg \min_{\mathbf{x}} (\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 + \alpha_1^2 \|\mathbf{x}\|_2^2 + \alpha_2^2 \|\mathbf{D}\mathbf{x}\|_2^2) . \quad (8.10)$$

Dabei ist \mathbf{b} der Vektor, der die n Datenpunkte des Stresstensors σ aus der MD-Simulation enthält, wobei $n \geq k + 1$ gelten muss, um die gefitteten Polynome sinnvoll bestimmen zu können, da die Koeffizienten der Polynome sonst unterbestimmt sind. α_1 und α_2 sind Regularisierungsparameter. Die Matrix \mathbf{A} ist die sogenannte Vandermonde-Matrix der Basis der Chebyshev-Polynome. Für die einzelnen Elemente der Vandermonde-Matrix in Monombasis, bzw. Chebyshev-Basis gilt:

$$V_{jk}(\dot{\gamma}) = (\dot{\gamma}_k)^j \quad \text{mit } j = \{0, \dots, P_{\max} - 1\} \quad \text{für Monombasis,} \quad (8.11)$$

$$A_{jk}(\dot{\gamma}) = T^j(\dot{\gamma}_k) \quad \text{mit } j = \{0, \dots, P_{\max} - 1\} \quad \text{für Chebyshev-Basis,} \quad (8.12)$$

wobei der Vektor $\dot{\gamma}$ die Werte der Scherrate enthält, für die Daten genommen wurden. Die Vandermonde-Matrix ist also eine $j \times k = N_{\dot{\gamma}} \times P_{\max}$ -Matrix.

Der Dämpfungsterm $\mathbf{D}\mathbf{x}$ entspricht dabei dem oben erwähnten Strafterm der Tikhonov-Regularisierung und dient der Abschwächung der Oszillation in der Differentialgleichung

$$\frac{\partial \mathbf{A}}{\partial \dot{\gamma}} \mathbf{x} = \mathbf{A} \mathbf{D} \mathbf{x} . \quad (8.13)$$

Das Äquivalenzproblem

$$(\mathbf{A}^T \mathbf{A} + \alpha_1^2 \mathbf{I} + \alpha_2^2 \mathbf{D}^T \mathbf{D}) \mathbf{x} = \mathbf{A}^T \mathbf{b} \quad (8.14)$$

wird gelöst, indem eine LU-Faktorisierung der LAPACK-Bibliothek [73] verwendet wird. Verschiedene Fallbeispiele hierfür sind in [1] zu finden.

8.2.1 Datenverfeinerung

Das Ziel der Senkung der Parameterordnung ist es, weniger kostspielige bzw. rechenzeit-aufwendige MD-Simulationen zu starten. Um also diese Anzahl an Simulationen zu minimieren, wird das obige Optimierungsproblem mit einer kleinen Anzahl an Stichproben –

¹ Je nach Quelle kann auch die Schreibweise Tschebyschow o.ä. verwendet werden.

also zuvor berechneten Datensätzen – durchgeführt. Danach wird mithilfe des „Greedy“-Algorithmus (Suche nach dem schlimmsten Fall) [74] diejenige Scherrate herausgesucht, an deren Stelle der Fit des Polynoms den größten Fehler aufweist. Beim nächsten Durchlauf der Fitmethode wird mindestens ein zusätzlicher Datensatz am vorgeschlagenen Phasenraumpunkt vorhanden sein und den Fit an der unsichersten Stelle korrigieren und verbessern.

Um die neu zu simulierende Scherrate zu erhalten, wird das Residuum von $A\mathbf{x} - \mathbf{b}$ in Abhängigkeit von $\dot{\gamma}$ betrachtet. Die vorgeschlagene neue Scherrate ist

$$\dot{\gamma}_{\text{neu}} = \frac{1}{2} (\dot{\gamma}_M + \dot{\gamma}_N) , \quad (8.15)$$

wobei $\dot{\gamma}_M = \arg \max_{\dot{\gamma}} (A\mathbf{x} - \mathbf{b})$ und $\dot{\gamma}_N$ der rechte oder linke Nachbar von $\dot{\gamma}_M$, das zum größeren Residuum gehört.

8.2.2 Erweiterung der Näherung auf ρ

Da neben der Scherrate auch die Teilchendichte des Systems den Stresstensor beeinflusst, ist es sinnvoll, die zuvor angewendete Näherung auf die Dichte zu erweitern. Zudem könnte auf diese Weise die Makro-Simulation auf kompressible Flüssigkeiten ausgedehnt werden. Entsprechend wird Gleichung (8.7) um unterschiedliche Dichten erweitert, man betrachtet daher die multivarianten Polynome des Grades (k_1, k_2) :

$$\sigma_i^* \approx \tilde{F}_i(\dot{\gamma}, \rho) = \sum_{j_1=0}^{k_1} \sum_{j_2=0}^{k_2} x_{ij_1j_2} \dot{\gamma}^{j_1} \rho^{j_2} , \quad (8.16)$$

wobei auch hier der Index i für die unabhängigen Komponenten des Stresstensors steht und je nach Dimension $i = \{1, 2, 3\}$ in zwei Dimensionen, bzw. $i = \{1, \dots, 6\}$ in drei Dimensionen ist. Dabei existieren wieder n unterschiedliche Datensätze für jede Scherrate $\dot{\gamma}$ und m unterschiedliche Datensätze für die Dichte ρ .

Die zugehörige Matrix A hat die Größe:

$$\left(\sum_{l=1}^m n_l \right) \times [(k_1 + 1) \cdot (k_2 + 1)] . \quad (8.17)$$

Um die angemessenen Polynomgrade k_1 bzw. k_2 zu bestimmen, wird für jede Dichte zunächst die zugehörige Näherung aus Abschnitt 8.2 bzw. Unterabschnitt 8.2.1 vorgenommen. Auf diese Weise bestimmt sich das Maximum für k_1 aus dem Maximum der bestimmten Polynomgrade für alle Dichten.

Anschließend wird bei $k_2 = 0$ begonnen, also ein Polynom vom Grad 0 an die Werte gefittet, um die obige Näherung (8.16) auf dieselbe Art und Weise zu bestimmen, wie zuvor (8.7). k_2 wird nach und nach erhöht, bis das Residuum gegen 0 konvergiert.

8.2.3 Optimierungsalgorithmus

In diesem Unterabschnitt werden die Schritte der zuvor erklärten Herangehensweise noch einmal zusammengefasst. Dabei ist der Algorithmus für Gleichung (8.7) angepasst, er kann aber auch auf Gleichung (8.16) angewendet werden. In letzterem Fall ist k_1 ein konstanter Wert, wie in 8.2.2 beschrieben. k_2 entspricht dem k des nachfolgenden Algorithmus, während m dem n entspricht. Zudem werden natürlich eine erweiterte Matrix A, b und ein erweiterter Standardfehler der Daten stderr verwendet.

Ein weiterer Parameter ist k_{\max} als obere Grenze des Polynomgrades. δ ist ein Schwellenwert, ab dem das Polynom konvergiert. Ein weiterer Schwellenwert ist ϵ , der bestimmt, ab wann das Residuum klein genug ist, damit keine weiteren verfeinernden Simulationen gestartet werden müssen.

- * **Schritt 1:** Setze die Startwerte der einzelnen Parameter, $k = \deg_{\min} = 0$, $n_{\min} = \deg_{\min} + 1$. $n = \max(n_{\min}, N)$, wobei N die Anzahl an vorhandenen Stichproben des der Verfeinerung zugrunde liegenden Parameters ist. Dabei ist zu beachten, dass mindestens n_{\min} Datensätze zur Verfügung stehen müssen, um den Optimierungsalgorithmus anwenden zu können. Der maximale Wert des Grades beträgt dabei $\deg_{\max} = \min(n, k_{\max})$.
- * **Schritt 2:** Entsprechend der Größe von k wird die zugehörige Matrix erstellt, um damit das Problem der kleinsten Quadrate aus Gleichung (8.10) zu lösen. Nun werden in einem Zwischenspeicher die Werte abgelegt, um sie später mit den Ergebnissen höheren Polynomgrades vergleichen zu können. Dabei wird $x_{\text{old}} = x$, $\text{res}_{\text{old}} = Ax - b$ und $\text{res}_{L^2, \text{old}} = \|x_{\text{old}}\|_2^2$ gesetzt.

- * **Schritt 3:** Nun wird die Differenz

$$\text{diff} = \frac{\|Ax - b\|_2^2 - \text{res}_{L^2, \text{old}}}{\text{res}_{L^2, \text{old}}}$$

bestimmt. Gilt $\text{diff} < \delta$, was ein Maß für die Konvergenz von k ist, werden die Werte auf die alten Werte zurückgesetzt, also $x = x_{\text{old}}$, $\text{res} = \text{res}_{\text{old}}$, $\text{res}_{L^2} = \text{res}_{L^2, \text{old}}$ und $k = k - 1$.

- * **Schritt 4:** In diesem Schritt wird überprüft, ob die Lösung optimiert ist. Dies trifft zu, wenn das Residuum $\text{res} \leq 3 \cdot \text{stderr}$ für jeden Datenpunkt ist.
- * **Schritt 5:** Konvergiert die Lösung nicht und ist auch nicht optimiert, wird $k = k + 1$ gesetzt. Anschließend wird die Matrix A angepasst und $x_{\text{old}} = x$, $\text{res}_{\text{old}} = \text{res}$ und $\text{res}_{L^2, \text{old}} = \text{res}_{L^2}$ gesetzt. Anschließend wird das Problem der kleinsten Quadrate aus Gleichung (8.10) gelöst.
- * **Schritt 6:** Ist $k < \deg_{\max}$ und die Lösung ist weder konvergiert noch optimiert, wird ab **Schritt 3** die Prozedur wiederholt. Andernfalls enthält der Vektor x die gesuchten Koeffizienten und k ist der Polynomgrad der Näherung.
- * **Schritt 7:** In diesem Schritt wird $\text{res}_{\text{RMS}} = \sqrt{\text{res}/n}$ berechnet. Ist $\text{res}_{\text{RMS}} > \epsilon$, wird mithilfe des Greedy-Algorithmus (Suche nach dem schlimmsten Fall) der neue Punkt im Phasenraum bestimmt, der den Fit der Polynome weiter verfeinern wird. Dieser neue Wert für $\hat{\gamma}_N$ ist Gleichung (8.15) zu entnehmen. Mit diesem neuen Parametersatz werden die benötigten teilchenbasierten Simulationen durchgeführt. Dadurch erhöht sich die Anzahl an Stichproben um $N = N + 1$. Der Algorithmus beginnt nun wieder bei **Schritt 1**. Anderenfalls wird der Optimierungsprozess beendet.

Dieser Algorithmus wurde in [1] ausführlich erörtert. Dort sind zudem einige Testfälle vorgestellt.

8.3 EIGENWERT-DEKOMPOSITION

Die Idee der nachfolgenden Methode ist es, den Stresstensor σ^{MD} als eine Matrix-Funktion des Verzerrungsgeschwindigkeitstensors \mathbf{E} herzuleiten. Der Stresstensor σ^{MD} , den man aus den MD-Simulationen erhält, wird durch eine Eigenwertdekomposition rotiert und damit in eine Diagonalform transformiert. Auf diese Weise reduziert sich die Anzahl der zu optimierenden Gleichungen auf 2, $f(\lambda)$ und $\sigma_{22}(\dot{\gamma})$. Wie bereits angedeutet werden die Daten einer einfachen Scherung (*simple shear*) als eine reine Scherung behandelt, ohne den antisymmetrischen Term, der aufgrund der Vortizität der einfachen Scherung hinzukommt (siehe [69]), dabei zu berücksichtigen. In Zukunft wäre es denkbar, komplexere Strömungsprobleme mithilfe dieser Methode zu untersuchen, indem die Eigenwert-Dekomposition auf weitere Dimensionen erweitert wird.

Die nachfolgenden Unterabschnitte sind [2] entnommen. Die einzelnen Methoden wurden in Form eines MatLab-Programms von Dr. Leonid Yelash zur Verfügung gestellt. Mithilfe eigener Daten von Simulationen in 3 Dimensionen wurden die aus zweidimensionalen Simulationen erworbenen Ergebnisse aus [2] nachvollzogen.

Jeder Verzerrungsgeschwindigkeitstensor \mathbf{E} kann als symmetrische Matrix aufgefasst werden. Dabei ist er eine Komposition aus dem Gradienten der Flussgeschwindigkeit $\mathbf{u} = (u, v)^T$ bzw. $\mathbf{u} = (u, v, w)^T$:

$$\mathbf{E} = \frac{1}{2} (\nabla \mathbf{u} + \nabla \mathbf{u}^T) . \quad (8.18)$$

Die Komponentenschreibweise ist in Gleichung (8.1) für drei Dimensionen gezeigt.

Anders als in [2] wird in dieser Arbeit der Fokus bei der Herleitung der nachfolgenden Methode auf den dreidimensionalen Fall gelegt, da dieser in dieser Arbeit vorzugsweise behandelt wird. Der zweidimensionale Fall folgt analog und kann in [2] nachvollzogen werden. Später werden die Ergebnisse der Veröffentlichung in Kapitel 9 präsentiert, bei der eine zweidimensionale Couette-Strömung und ein dreidimensionaler Poiseuille-Fluss simuliert und mit dieser Methode optimiert werden.

In Abschnitt 8.1 wurde bereits besprochen, dass der Verzerrungsgeschwindigkeitstensor derart gedreht werden kann, dass es zu einer reinen Scherung in einem planaren Fluss kommt. Der dafür benötigte Winkel hat nach [75] den Wert

$$\theta = \frac{1}{2} \arctan \left(-\frac{E_{11}}{E_{13}} \right) , \quad (8.19)$$

wobei E_{ij} die entsprechenden Einträge aus dem Verzerrungsgeschwindigkeitstensor sind. Setzt man diesen Winkel θ in eine dreidimensionale Rotationsmatrix Θ ein und wendet diese Rotation auf \mathbf{E} an, ergibt sich daraus ein reiner Scherdeformationstensor, wie in Gleichung (8.2) gezeigt, und man erhält:

$$\mathbf{E}^* = \Theta \mathbf{E} \Theta^T = \begin{pmatrix} 0 & 0 & \frac{\dot{\gamma}}{2} \\ 0 & 0 & 0 \\ \frac{\dot{\gamma}}{2} & 0 & 0 \end{pmatrix} \quad (8.20)$$

mit der Scherrate $\dot{\gamma} = 2(E_{13} \cos(2\theta) - E_{11} \sin(2\theta))$.

Über eine sogenannte Eigenwert-Dekomposition des reinen Scherdeformationstensors \mathbf{E}^* erhält man:

$$\mathbf{E}^* = \mathbf{P} \mathbf{\Lambda} \mathbf{P}^T = \mathbf{P} \begin{pmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{pmatrix} \mathbf{P}^T , \quad (8.21)$$

wobei $\lambda_i \in \mathbb{R}$ die Eigenwerte des Tensors \mathbf{E}^* und $\mathbf{P} = (p_1, p_2, p_3)$ die Matrix der zugehörigen Eigenvektoren sind. Durch Lösen des Eigenwertproblems ergeben sich für die Eigenwerte $\lambda_{1,3} = \mp \frac{\dot{\gamma}}{2}$ die positive und negative Scherrate, für $\lambda_2 = 0$, sowie die Eigenvektormatrix

$$\mathbf{P} = \begin{pmatrix} \frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} \\ 0 & -1 & 0 \\ -\frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} \end{pmatrix}. \quad (8.22)$$

Wie zuvor auch wird nach einer Fitfunktion $f(\dot{\gamma})$ gesucht, mit der man die teilchenbasierten Simulationsdaten annähern kann. Dabei sind die Daten nun $\boldsymbol{\sigma}^{\text{MD}} \equiv \{\sigma_{\alpha\beta}(\dot{\gamma})\}_{N_{\dot{\gamma}}}^{N_{\text{sets}}}$, wobei $N_{\dot{\gamma}}$ die Anzahl an Simulationen mit verschiedenen Scherraten $\dot{\gamma}_i$ und N_{sets} die Anzahl an statistisch unabhängigen Datensätzen sind. An dieser Stelle ist wichtig, dass die Daten für $\boldsymbol{\sigma}$ unkorreliert sind und nur im stationären, stabilen Zustand (engl. *Steady State*) aufgenommen werden. Für die gesuchte Matrix-Funktion soll nun

$$f(\dot{\gamma}) \approx \boldsymbol{\sigma}^{\text{MD}} \quad (8.23)$$

gelten. Da \mathbf{E}^* als Einträge lediglich die Scherrate als Information enthält, ist diese Aussage gleichbedeutend mit

$$f(\dot{\gamma}) = f(\mathbf{E}^*) = f\left(\mathbf{P} \begin{pmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{pmatrix} \mathbf{P}^T\right) \quad (8.24)$$

$$= \mathbf{P} \begin{pmatrix} f\left(-\frac{\dot{\gamma}}{2}\right) & 0 & 0 \\ 0 & f(0) & 0 \\ 0 & 0 & f\left(\frac{\dot{\gamma}}{2}\right) \end{pmatrix} \mathbf{P}^T \quad (8.25)$$

$$\approx \boldsymbol{\sigma}^{\text{MD}}. \quad (8.26)$$

Für eine Näherung der kleinsten Quadrate für $f(\dot{\gamma})$ müssen die Datensätze der teilchenbasierten Simulation in die Basis der Eigenvektoren von \mathbf{E}^* transformiert werden. An dieser Stelle wird angenommen, dass die Basisvektoren von \mathbf{E} und $\boldsymbol{\sigma}$ gleich sind, was die Konsequenz der Tatsache ist, dass nur reine Scherung betrachtet und die Vortizität der einfachen Scherung vernachlässigt wird. Diese Annahme ist nicht allgemein gültig, gilt aber für Newtonsche Fluide (und einige Nichtnewtonsche Fluide) [69]. In Zukunft soll das Hybrid-Verfahren auf allgemeine Strömungsprofile erweitert werden.

Durch Anwenden der Basistransformation erhält man eine Diagonalmatrix der nachfolgenden Form:

$$\tilde{\boldsymbol{\sigma}} \equiv \mathbf{P}^T \boldsymbol{\sigma}^{\text{MD}} \mathbf{P} \quad (8.27)$$

$$\approx \begin{pmatrix} f\left(-\frac{\dot{\gamma}}{2}\right) & 0 & 0 \\ 0 & f(0) & 0 \\ 0 & 0 & f\left(\frac{\dot{\gamma}}{2}\right) \end{pmatrix}. \quad (8.28)$$

Entsprechend der Transformation wird angenommen, dass die Nebendiagonalkomponenten 0 ergeben. Dieser Umstand ist nur für den Fall zutreffend, dass eine reine Sche-

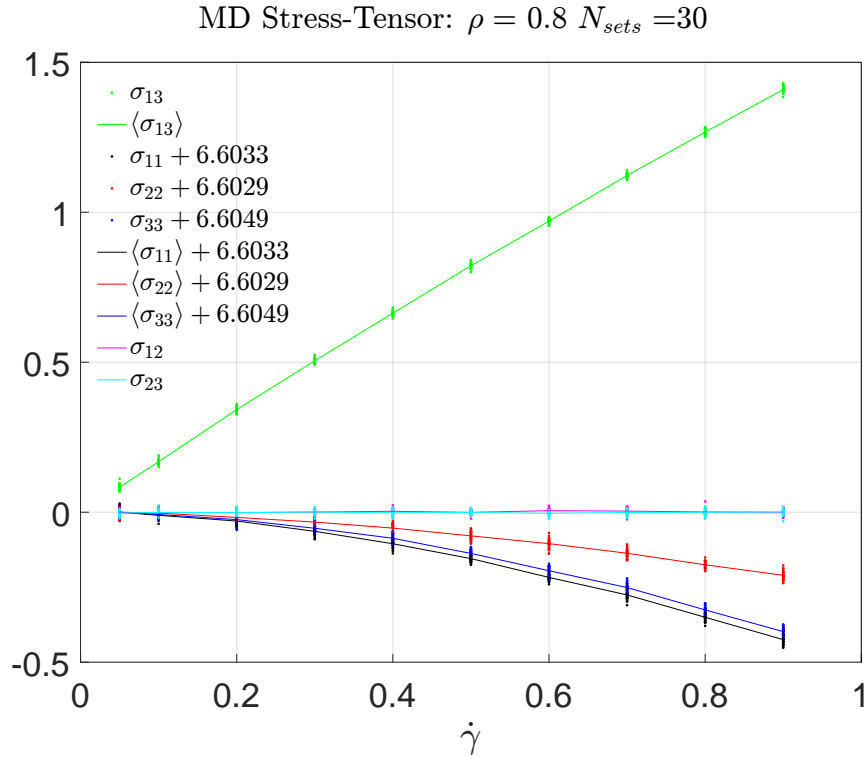


Abb. 8.3: σ^{MD} , aufgetragen gegen die Scherrate $\dot{\gamma}$ eines Systems mit einer Dichte von $\rho = 0.8$. Während die Diagonalkomponenten des Stresstensors nur eine geringe Abhängigkeit von der Scherrate aufweisen, steigt σ_{13} fast linear, was zu erwarten ist. Die anderen beiden Nebendiagonalkomponenten sind annähernd 0. Sowohl bei den Hauptdiagonalkomponenten σ_{ii} als auch bei der σ_{13} -Komponente ist neben den Datenpunkten auch der jeweilige durchschnittliche Weg als durchgezogene Linie dargestellt. Die Hauptdiagonalelemente wurden um ihren jeweiligen ungescherten Wert verschoben.

rung vorliegt, da für eine einfache Scherung die Eigenvektoren minimal verschieden sind. Durch die Annahme einer Diagonalmatrix reduziert sich die notwendige Optimierung auf

$$f(\lambda_1) = f\left(-\frac{\dot{\gamma}}{2}\right) \approx \tilde{\sigma}_{11} , \quad (8.29)$$

$$f(\lambda_3) = f\left(\frac{\dot{\gamma}}{2}\right) \approx \tilde{\sigma}_{33} , \quad (8.30)$$

wodurch lediglich eine Gesamtfunktion f optimiert werden muss. Da für den Eigenwert $\lambda_2 = 0$ die Datenwerte von $\tilde{\sigma}_{22}$ auf einen einzigen Punkt projiziert werden, muss $\tilde{\sigma}_{22}$ gesondert betrachtet und optimiert werden.

Die Daten für den Stresstensor σ^{MD} werden für verschiedene Dichten unter Verwendung des isokinetischen Thermostats, sowie dem SLLOD-Algorithmus erstellt. Die harten Kugeln werden wie zuvor mit einem WCA-Potential modelliert, das in Kapitel 4 besprochen wird.

Exemplarisch wird in Abbildung 8.3 die Abhängigkeit der einzelnen Komponenten des Stresstensors von der Scherrate dargestellt. Die Werte für die eingestellte Scherrate variieren dabei von $\dot{\gamma} = 0.001$ bis $\dot{\gamma} = 0.9$. Die Anzahl an Datensätzen mit unterschiedlicher Scherrate liegt dabei bei $N_{\dot{\gamma}} = 10$. Jeder Parametersatz wurde $N_{sets} = 30$ mal unabhängig voneinander simuliert.

Die σ_{13} -Komponente des Stresstensors zeigt eine schwache Abhängigkeit von der Scherrate. Auch die Diagonalkomponenten σ_{ii} weisen eine leichte Abhängigkeit von der Scherrate auf. Zur besseren Übersichtlichkeit werden diese Daten um den Wert bei einer Scherrate von $\dot{\gamma} = 0$ verschoben, sodass sie bei 0 beginnen. Dabei ist besonders zu beachten, dass σ_{22} weniger von $\dot{\gamma}$ abhängt als die übrigen Diagonalkomponenten.

Wie zu erwarten ist, stellen die übrigen Komponenten des Stresstensors unabhängig von der Scherrate ein Rauschen nahe 0 dar. Zudem ist das Vorzeichen der Diagonalkomponenten dem Vorzeichen der σ_{13} -Komponente entgegengerichtet.

Diese Daten, die direkte Ergebnisse aus der Simulation darstellen, werden nun in die Eigenvektorbasis des reinen Verzerrungsgeschwindigkeitstensors transformiert. Zudem werden sie nach λ abgebildet, also den Eigenwert, der entweder die positive oder die negative, halbe Scherrate darstellt.

Die einzelnen Komponenten des transformierten Tensors sind in Abbildung 8.4 dargestellt. Dabei werden die Komponenten in Abhängigkeit des Eigenwerts λ gezeigt.

In Abbildung 8.4 a) werden lediglich die Diagonalkomponenten des transformierten Tensors gezeigt, die ein monotonen Verhalten aufweisen. Auffällig ist hier, dass die beiden „Äste“ der Funktion nicht symmetrisch sind. Insgesamt ist die Krümmung der Kurve bei $\lambda_1 \leq 0$ stärker, beide „Äste“ lassen sich gut durch Polynome beschreiben.

Da der Wert bei $\lambda = 0$ nicht extrapoliert, sondern interpoliert werden kann, ist die Fehlerquelle hierfür wesentlich geringer. Die Annahmen, die man für eine Extrapolation treffen muss, sind nicht notwendig. Außerdem ist ersichtlich, dass $f(\lambda_2)$ gut zwischen den beiden Ästen liegt und $f(\lambda_{1,3})$ interpoliert.

In Abbildung 8.4 b) sind die Nebendiagonalkomponenten nach der Transformation zu sehen. Alle, auch die zuvor von der Scherrate abhängige σ_{13} -Komponente, fallen wie erwartet auf annähernd Null. Die Amplitude der Punkte liegt dabei bei etwa $\Delta f = 0.06$, was $\approx 1\%$ des Wertes der Diagonalkomponenten des transformierten Tensors $\tilde{\sigma}$ entspricht. Diese relativ große Streuung ist durch statistisches Rauschen und Ignorieren der Vortizität der einfachen Scherung zu begründen. In den nachfolgenden Untersuchungen wird angenommen, dass die Nebendiagonalelemente verschwinden.

8.4 RAUSCHREDUZIERUNG

Da die Daten aus der teilchenbasierten MD-Simulation statistischen Schwankungen unterliegen, ist es sinnvoll, eine Hauptkomponentenanalyse, in [2] als *Proper orthogonal Decomposition* (POD) bezeichnet, durchzuführen. Auf diese Weise können physikalisch relevante Informationen aus den MD-Daten entnommen werden. Zudem führt dies zu einer Reduzierung des Rauschens.

Diese Methode der POD basiert auf einer Singulärwertzerlegung der Matrix, engl. *Singular Value Decomposition* (SVD). Diese Zerlegung erlaubt es, die Daten zu gewichten. Neben der Gewichtung erhält man zusätzlich einen Einblick darauf, wie stark die einzelnen Zeilen und Spalten der ursprünglichen Matrix miteinander korreliert sind.

Für die SVD wird der Stresstensor als Multiplikation dreier Matrizen dargestellt:

$$\tilde{\sigma} = \mathbf{V} \mathbf{\Sigma} \mathbf{Z}^T, \quad (8.31)$$

wobei \mathbf{V} eine unitäre $m \times m$ Matrix aus Links-Singulärvektoren ist. Bei \mathbf{Z}^T handelt es sich um die Adjungierte einer unitären $n \times n$ -Matrix \mathbf{V} aus Rechts-Singulärvektoren. Die Matrix $\mathbf{\Sigma}$ ist eine $m \times n$ -Matrix mit Rang $r \leq \min(m, n)$, auf deren Hauptdiagonalen die

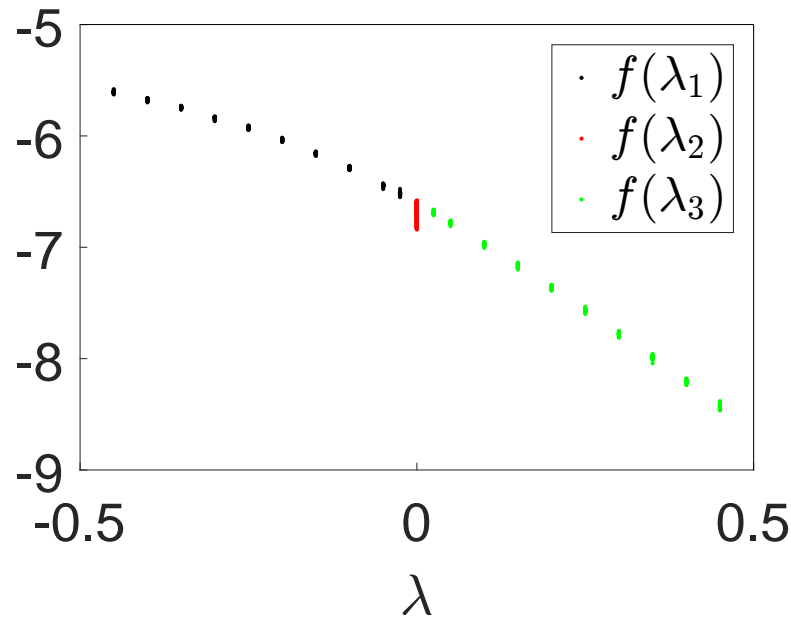
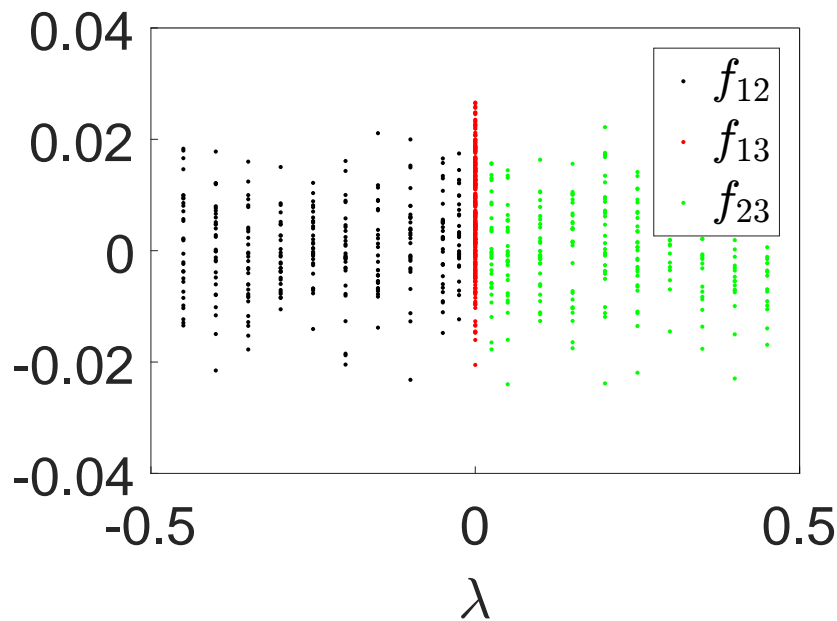
(a) Diagonalkomponenten vs. λ .(b) Nebendiagonalkomponenten vs. λ .

Abb. 8.4: Basistransformierte Daten, aufgeteilt in (a) Diagonalkomponenten $\tilde{\sigma}_{ii}^{\text{MD}}$ und (b) Nebendiagonalkomponenten $\tilde{\sigma}_{ij}^{\text{MD}}$ eines Systems mit einer Dichte von $\rho = 0.8$, aufgetragen gegen λ . Die Funktion $f(\lambda)$ wird an diese Daten angenähert und optimiert. (a) zeigt dabei die Diagonalkomponenten, die monoton fallend sind. Durch die Basistransformation sind nur noch die beiden Komponenten $\tilde{\sigma}_{11}$ und $\tilde{\sigma}_{33}$ stark von der Scherrate abhängig, die mittlere Hauptdiagonale hat den Eigenwert $\lambda_2 = 0$ für alle γ und ist entsprechend keine Funktion $f(x)$. Die Werte liegen dennoch wie erwartet zwischen den beiden Ästen. Die Nebendiagonalkomponenten in (b) sind nach der Transformation wie zu erwarten nahe Null.

positiven Singulärwerte der Matrix $\tilde{\sigma}$ liegen. Die übrigen Einträge sind Null. Dabei gilt $s_1 \geq s_2 \geq s_3 \geq \dots \geq s_r > 0$:

$$\Sigma = \begin{pmatrix} s_1 & & 0 \\ & \ddots & \\ 0 & & s_r \\ & & & 0 \end{pmatrix}. \quad (8.32)$$

Der Rang m der ersten Matrix entspricht dabei der Anzahl an γ_i Werten N_γ , bei denen simuliert wird. Der Rang der dritten Matrix ist n und entspricht der Anzahl an unabhängigen Simulationen, die pro Parametersatz durchgeführt werden, also dem zuvor eingeführten Parameter N_{sets} . Die Singulärwerte s_i sind dabei $s_i^2 = \lambda_i^{(C)}$, also die Quadrate der Eigenwerte von Matrix $C = \tilde{\sigma}^T \tilde{\sigma}$.

Mithilfe der POD-Methode lässt sich der Rang der ursprünglichen Matrix $\tilde{\sigma}$ reduzieren, indem man eine Niedrigrangapproximation zur Matrix $\tilde{\sigma}_k$ durchführt. Hierbei werden die ersten k der r Singulärwerte s_i in Σ gehalten:

$$\begin{aligned} \tilde{\sigma} &\approx \tilde{\sigma}_k = \mathbf{V} \Sigma_k \mathbf{Z}^T \\ &= \sum_{i=1}^k s_i \mathbf{V}_i \otimes \mathbf{Z}_i^T, \end{aligned} \quad (8.33)$$

wobei k der verringerte Rang der Matrix ist. Entsprechend handelt es sich bei Σ_k um die Singulärwert-Matrix mit reduziertem Rang.

Bei der SVD-Analyse des Stresstensors ergibt sich bei allen Komponenten, dass der erste Singulärwert s_1 um zwei Größenordnungen höher ist, als die übrigen s_i . Dies ist in Abbildung 8.5 (a) für den σ_{11} -Wert dargestellt. In diesem Fall wurden $N_{\text{sets}} = 30$ unabhängige Simulationen mit den jeweiligen Parametern durchgeführt.

Die übrigen Werte von s_i für $i > 1$ haben annähernd denselben Wert oder zumindest dieselbe Größenordnung und entsprechen statistischem Rauschen der MD-Daten. Bei sehr großen Werten von $i > 25$ sinkt die Größenordnung der Singulärwerte erneut um eine Größenordnung. Daraus lässt sich schließen, dass die erste Hauptkomponente mit der größten Varianz entlang der Richtung des zugehörigen Vektors \mathbf{V}_1 die für uns relevanten Daten des Stresstensors annähert:

$$\tilde{\sigma} \approx \tilde{\sigma}_k = s_1 \mathbf{V}_1 \otimes \mathbf{Z}_1^T. \quad (8.34)$$

Daraus ergibt sich ein relativer Projektionsfehler der Methode:

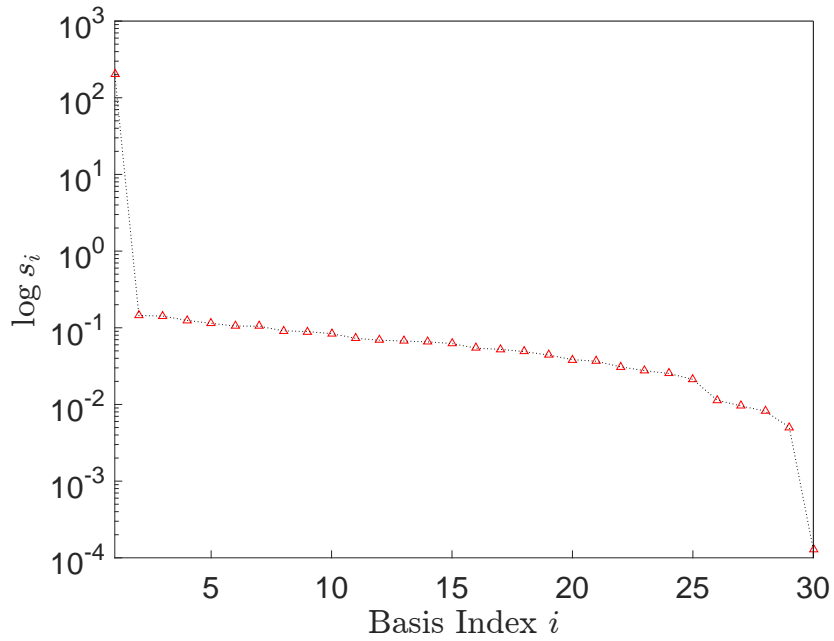
$$\varepsilon_{(1)}^2 = \frac{\sum_{i=2}^r s_i^2}{\sum_{i=1}^r s_i^2}. \quad (8.35)$$

Dieser Fehler ist in Abbildung 8.5 (b) in Abhängigkeit der Größe des Basisraums k dargestellt.

Bei einer Rang-1-Näherung für $\tilde{\sigma}_{(1)}$ ergibt sich dementsprechend ein relativer Wert von $\varepsilon_{(1)} \lesssim 0.9\%$.

8.5 CHEBYSHEV-APPROXIMATION MIT TIKHONOV-REGULARISIERUNG

Die im vorherigen Abschnitt erhaltenen, rauschreduzierten Daten der Hauptachsen des Stresstensors $\tilde{\sigma}_{(1)}$ werden nun mit Polynomen angenähert. Dafür wird die Methode der



(a) Singulärwerte vs. Index.

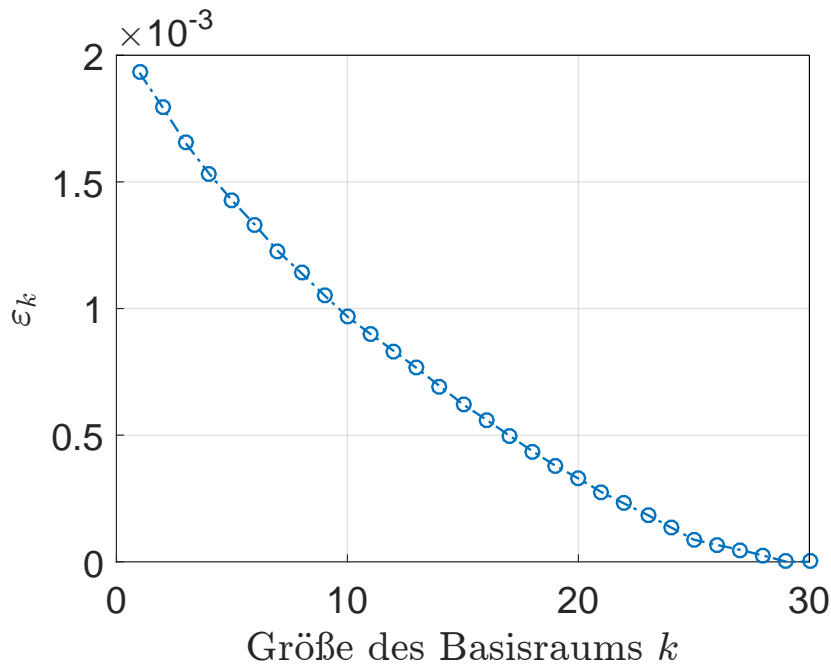
(b) Rel. Fehler vs. k .

Abb. 8.5: Singulärwerte und relativer Fehler der Singulärwertdekomposition der Niedrigrangreduzierung. In (a) sind die Singulärwerte s_i logarithmisch gegen ihren Index i aufgetragen. Man erkennt deutlich, dass der erste Singulärwert mit einem Unterschied von zwei Größenordnungen der größte Singulärwert und damit wichtigste ist. Die übrigen Werte entsprechen dem statistischen Rauschen der MD-Daten. In (b) ist der relative Fehler der Projektion bei der Niedrigrangapproximation in Abhängigkeit des reduzierten Rangs k zu sehen. Der Maximalwert von k ist dabei 30, da mit $N_{\text{sets}} = 30$ unabhängigen Datensätzen gearbeitet wurde.

kleinsten Quadrate unter Verwendung der orthogonalen Chebyshev-Polynome verwendet, wie in Abschnitt 8.2 angedeutet:

$$\tilde{\sigma}_{(1)} \approx f(\lambda) = \sum_{i=0}^m x_i T_i(\lambda), \quad \lambda \in \mathbb{R}. \quad (8.36)$$

Dabei ist m der Grad des annähernden Polynoms, x_i sind die Koeffizienten der Chebyshev-Polynome T_i . Die Wahl dieser Polynome führt zu einer geringeren Oszillation verglichen mit der Verwendung von reinen Monomen X_i . Insgesamt sollen die Simulationsdaten mithilfe einer einfachen Funktion $f(\gamma)$ angenähert werden, um diese einfache Funktion für die Lösung der makroskopischen Gleichungen verwenden zu können.

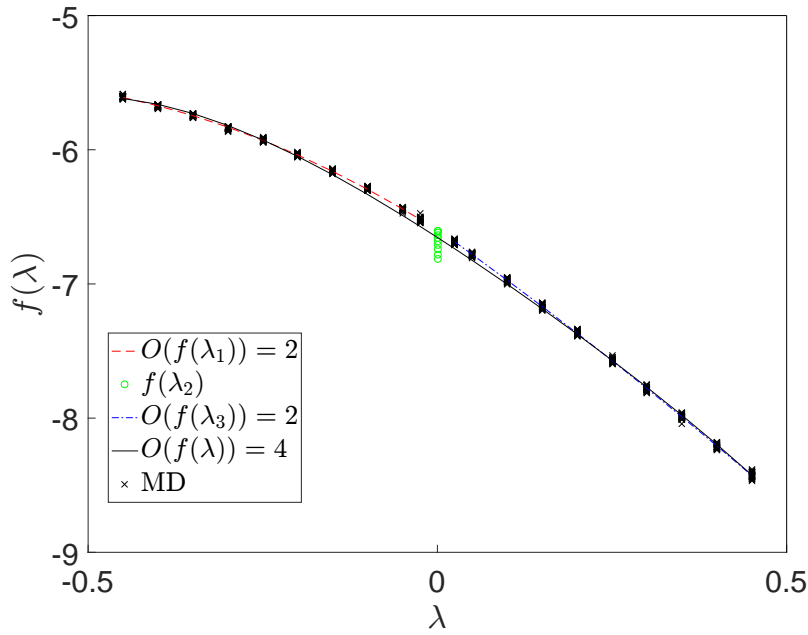
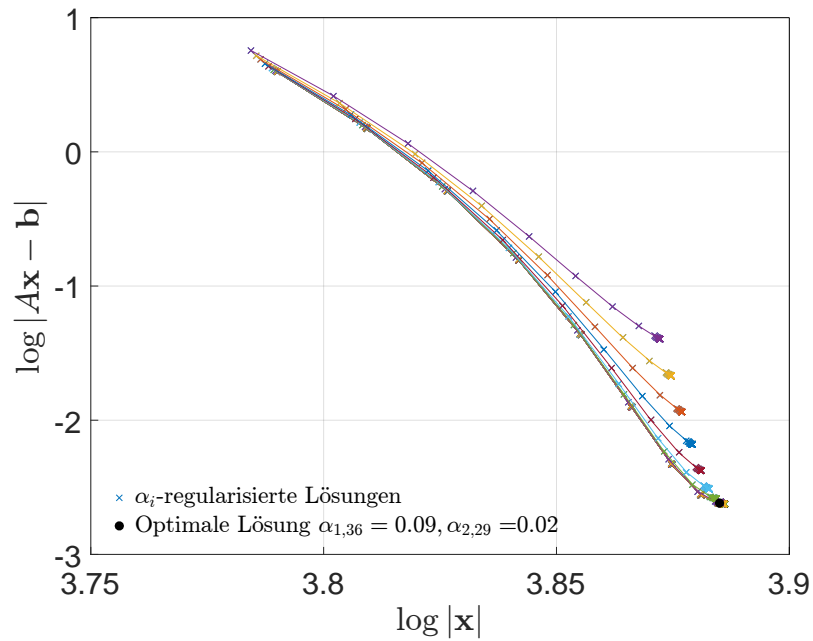


Abb. 8.6: Tikhonov-Näherung mit Chebyshev-Polynomen. Die Daten können mit einem Polynomgrad 4 angemessen gut beschrieben werden.

In der Offline-Trainingsphase wird ein Problem der kleinsten Quadrate gelöst, indem die Tikhonov-Regularisierung für jede Hauptkomponente des Stresstensors angewendet wird. Diese Regularisierung verbessert die Konditionszahl der Matrix A . Es wird ein Vektor x gesucht, der wie in Abschnitt 8.2 besprochen das erweiterte Residuenfunktional aus Gleichung (8.10) minimiert. Die im Abschnitt 8.2 besprochene Faktorisierung wird auf die reduzierten Daten angewendet.

Aus dieser Faktorisierung ergibt sich die Chebyshev-Näherung mit der angewendeten Tikhonov-Regularisierung, die einen Strafterm in Form eines Dämpfungsterms berücksichtigt, wie in Abbildung 8.6 für den in den vorherigen Kapiteln vorgestellten Datensatz dargestellt wird. Da die beiden Äste für die unterschiedlichen Eigenwerte ein unterschiedliches Verhalten aufweisen, kann für beide Eigenwerte $\lambda_{1,3}$ einzeln eine Näherung angenommen werden, sowie für die Gesamtfunktion $f(\lambda)$.

Sowohl $f(\lambda_1)$ mit $\lambda_1 < 0$ als auch $f(\lambda_3)$ mit $\lambda_3 > 0$ werden gesondert betrachtet. Beide Funktionen können mit einem Polynom zweiter Ordnung genähert werden, wie in Abbildung 8.6 zu sehen ist. Wenn beide Funktionen zusammen betrachtet werden, ergibt



(a) Die L-Kurve des Tikhonov Fits mit Chebyshev-Polynomen.

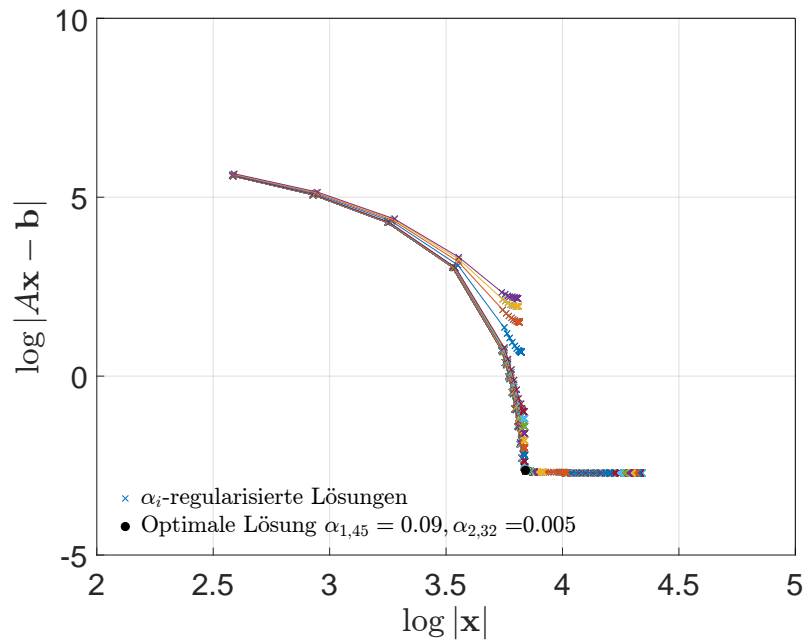
(b) Die L-Kurve des Tikhonov Fits mit Monomen für Polynomgrad $P_{\max} = 8$.

Abb. 8.7: L-Kurve des Tikhonov-Fits der Gesamtlösung $f(\lambda)$ (a) mit Chebyshev-Polynomen und (b) mit Monomen. Die Kurven sind im Hilbertraum \mathcal{H}_1 berechnet. Die L-Kurve der Chebyshev-Polynome weist nicht die typische Form auf, wie sie in (b) für Monome deutlich zu erkennen ist. In (b) ist neben der anderen Basis auch der maximal erlaubte Polynomgrad auf $P_{\max} = 8$ gesetzt. Werden die beiden Äste der Funktion einzeln gefittet, kann für jeden Fit eine eigene L-Kurve zur Analyse angefertigt werden.

sich eine wesentlich höhere Ordnung, um beide Äste optimal beschreiben zu können. In diesem Fall ergibt sich für $f(\lambda)$ ein Polynomgrad von $P = 4$.

Für optimale Effizienz ist es bevorzugt, wenn ein Polynom mit möglichst niedrigem Polynomgrad als Näherung verwendet wird. Dadurch wird die numerische Effizienz der automatisierten Algorithmen enorm verbessert. Außerdem sorgt es für eine höhere Stabilität der Algorithmen zur Datenverarbeitung.

In Abbildung 8.7 sind die L-Kurven des Problems für die Lösung des Gesamtbereichs $f(\lambda)$ dargestellt. Bei L-Kurven handelt es sich um doppelt-logarithmische Plots, die die Norm der Residuen-Norm $\|A\mathbf{x}_\alpha - \mathbf{b}\|$ zur Norm der entsprechenden regularisierten Lösung $\|\mathbf{x}_\alpha\|$ auftragen. Je nach Quelle können die Achsen vertauscht sein. Mithilfe der L-Kurven lässt sich die Größenordnung der optimalen Regularisierungsparameter $\alpha_{1,2}$ bestimmen. Sie gibt außerdem Rückschlüsse auf die Regularisierungsmethode des zugrundeliegenden Problems, siehe [76]. Mithilfe dieser Methode kann also bestimmt werden, wie stark man die zuvor eingeführten Strafterme berücksichtigen muss, um die Lösung zu optimieren.

Da zwei Parameter $\alpha_{1,2}$ betrachtet werden, ergibt sich eine Schar von Lösungen, die in unterschiedlichen Farben gekennzeichnet sind. Der optimale Wert der Parameter $\alpha_{1,2}$ wird durch einen schwarzen Punkt gekennzeichnet. Im Normalfall kennzeichnet die Ecke der L-Form diese optimalen Parameter, da an dieser Stelle kleine Änderungen in \mathbf{x}_α , den anzunähernden Parametern der Lösung, große Änderungen im Residuum hervorrufen. Dabei bildet der Eckpunkt des L den größtmöglichen Wert für $\|\mathbf{x}_\alpha\|$ bei kleinstmöglichem Wert für die Residuen-Norm $\|A\mathbf{x}_\alpha - \mathbf{b}\|$.

Die Form der L-Kurve hängt dabei zum einen von der Wahl des zu bestimmenden Polynomgrads der Lösung ab, wie im Anhang in Kapitel C für verschiedene Polynomgrade gezeigt. Außerdem ist die Lösung je nach Wahl der zugrundeliegenden Basis besser oder schlechter. Der untere Teil der L-Kurve, also der waagerechte Teil der Kurve, ist in den für diesen Fit gewählten Parameter in Abbildung 8.7 a) nur durch starkes Zoomen ersichtlich. Auch der horizontale Teil des L flacht schnell ab. Dies ist ein Indiz dafür, dass die Parameter für das zugrundeliegende Problem gut gewählt sind, die Daten werden gut durch die Lösung beschrieben.

Um die typische L-Form zu erhalten, kann auf künstliche Art die Lösung verschlechtert werden, indem zum Beispiel der Polynomgrad erhöht und die Basis von Chebyshev-Polynomen (vgl. Gleichung (8.9)) zu Monomen (vgl. Gleichung (8.8)) gewechselt wird. Die L-Kurve, die man mit diesen Änderungen erhält, ist in Abbildung 8.7 b) zu sehen. Anhand dieser Kurve lässt sich die Lösung des Problems angemessen optimieren.

Chebyshev-Polynome führen, da sie orthogonale Polynome darstellen, zu einer stabileren Lösung. Auch mit erhöhtem Polynomgrad erreicht man mit Chebyshev-Polynomen einen sehr kleinen, waagerechten Abschluss der Kurve, wie im Anhang in Kapitel C näher erläutert.

8.6 RÜCKTRANSFORMATION

Nachdem die Daten aus den teilchenbasierten Simulationen durch eine Transformation in eine Form gebracht sind, in der sie mithilfe einer einzigen Polynomnäherung approximiert werden können, muss diese Näherung nun wieder zurückgedreht werden, da der von der Makrosimulation geforderte Input von der Dehnungsrate \mathbf{E} und nicht \mathbf{E}^* abhängen soll. Die Rotation findet also in den Basisraum der Makrosimulation hinein statt.

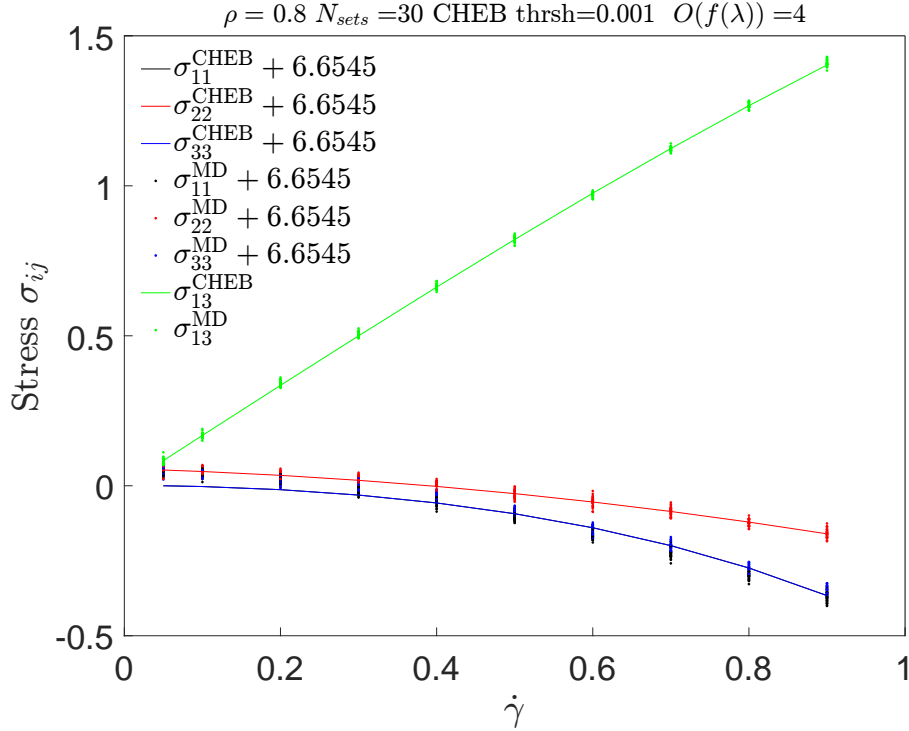


Abb. 8.8: Chebyshev-Näherung $f(\lambda)$ nach Rücktransformation in die Basis der Mikrosimulation $\sigma(\dot{\gamma})$. Erneut sind die Hauptdiagonalelemente von σ_{ii} um einen konstanten Wert verschoben, um eine bessere Übersichtlichkeit zu garantieren. Die Symbole entsprechen dabei den ausgelesenen Datenpunkten, die Linien entsprechen der Näherung nach der Transformation, die die Daten sehr gut beschreiben.

Dementsprechend muss der mithilfe der Chebyshev-Näherung ermittelte Wert für die Hauptdiagonalen des Stresstensors in der Eigenbasis $f(\lambda)$ rücktransformiert werden in $\tau(\mathbf{E})$ aus Gleichung (8.5). Hierfür werden die beiden Transformationsgleichungen aus Gleichungen (8.2) und (8.28) erneut umgekehrt angewandt:

$$\sigma(\mathbf{E}) = \Theta^T \mathbf{P} \begin{pmatrix} f\left(-\frac{\dot{\gamma}}{2}\right) & 0 & 0 \\ 0 & f(0) & 0 \\ 0 & 0 & f\left(\frac{\dot{\gamma}}{2}\right) \end{pmatrix} \mathbf{P}^T \Theta. \quad (8.37)$$

Bei dieser Rücktransformation muss darauf geachtet werden, dass die korrekte Rotationsmatrix Θ verwendet wird, da sie spezifisch für eine bestimmte Anfangskonfiguration steht.

In Abbildung 8.8 ist die Rücktransformation in die Mikrosimulationsbasis zu sehen, als die Rücktransformation von Gleichung (8.28). Man erkennt, dass die Näherung die Daten sehr gut beschreibt. Vor allem der annähernd lineare Zusammenhang zwischen σ_{13} und der Scherrate $\dot{\gamma}$ ist gut zu sehen.

Wie bei den Ergebnissen im zweidimensionalen Fall, die in [2] präsentiert wurden, sind auch im dreidimensionalen Fall nach der Rücktransformation $\sigma_{11}^{\text{CHEB}}$ und $\sigma_{33}^{\text{CHEB}}$ identisch. Dies ist darauf zurückzuführen, dass nach der Transformation die Nebendiagonalkomponenten auf 0 gesetzt sind und die einfache Scherung als reine Scherung behandelt wird.

Durch eine Verfeinerung des Algorithmus und eine Erweiterung der Fitmethode auf zusätzliche Tensoren wie in [69] aufgezeigt wurde, ist anzunehmen, dass sich die Daten besser annähern und beschreiben lassen. Auf diese Weise lassen sich auch allgemeine gescherte Systeme mit dieser Methode beschreiben.

In diesem Kapitel sollen die bisherigen Ergebnisse der Kopplung der beiden Simulationsebenen vorgestellt werden. Dabei werden die Offline-Ergebnisse vorgestellt, die auch in [2] präsentiert wurden. Da diese einen einfachen Couette- bzw. Poiseuille-Fluss lösen und zu diesen Ergebnissen exakte numerische Lösungen vorliegen, kann damit das Grundkonzept der hybriden Simulation überprüft und die Funktionsweise gezeigt werden. Die hier vorgestellten Ergebnisse wurden mithilfe der von Dr. Leonid Yelash entwickelten Routinen aus dem vorherigen Kapitel und dem von Dr. Nehzat Emamy entwickelten CFD-Programm erstellt. Die zugrundeliegenden MD-Daten stammen für den dreidimensionalen Poiseuille-Fluss aus der in dieser Arbeit vorgestellten MD-Simulation. Für den zweidimensionalen Fall wurde ein von Dr. Leonid Yelash entwickelter Code verwendet, der eine zweidimensionale Simulation als Grundlage hat.

9.1 COUETTE-STRÖMUNGEN MIT MONOMEREN

In diesem Abschnitt wird die Strömung in einem Kanal mithilfe der CFD-Simulation simuliert. Die in dieser Arbeit präsentierten Ergebnisse aus den MD-Simulationen werden mit einer Offline-Kopplung eingespeist und mit den in Kapitel 8 vorgestellten Methoden weiterverarbeitet. Die Couette-Strömung wird dabei mit zweidimensionalen MD-Simulationen, der Poiseuille-Fluss mit dreidimensionalen Daten berechnet. Die Ergebnisse wurden bereits in [2] veröffentlicht.

Für die Couette-Strömung betrachtet man das Simulationsgebiet $\Omega = [-1, 1] \times [0, 1]$. An den Wänden wird eine Haftbedingung hinzugefügt. Wie in Abbildung 9.1 dargestellt, wird bei $y = 0$ die Geschwindigkeit Null gehalten. Bei $y = 1$ wird die Wand des Kanals mit einer konstanten Geschwindigkeit U in x -Richtung bewegt, wodurch sich ein lineares Scherprofil einstellt.

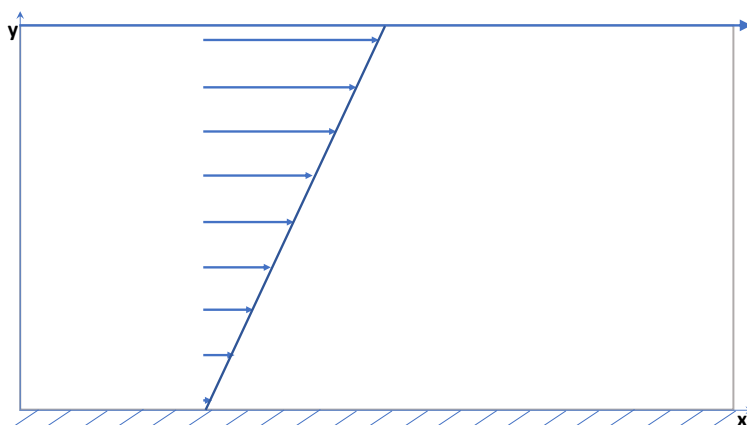


Abb. 9.1: Darstellung der Couette-Strömung. Bei dieser Art der Strömung wird eine Komponente des Kanals mit einer konstanten Geschwindigkeit U bewegt. Dadurch entsteht ein lineares Scherprofil im Inneren des Kanals, wie im Bild angedeutet.

Auf das Gebiet werden 3×3 Zellen gelegt, wie in Abschnitt 5.2.1 vorgestellt. Der eingestellte Polynomgrad wird auf $k = 1$ gesetzt, die Reynoldszahl $Re = 1$. Die Simulationen finden bei einer Dichte von $\rho = 0.8$ statt. Die Navier-Stokes-Gleichungen vereinfachen sich zu:

$$\frac{d^2 u}{dy^2} = 0. \quad (9.1)$$

Analytisch wird dies über

$$u(y) = U \frac{y}{h} \quad (9.2)$$

gelöst, wobei h die Höhe des Kanals in y -Richtung ist. y entspricht der y -Koordinate des Kanals und U ist die Geschwindigkeit der oberen Wand. Die Adams-Bashforth-Methode [54] wird wie in Gleichung (5.25) zur Zeitintegration angewendet.

Da der Fluss zweidimensional ist, werden hier Daten aus zweidimensionalen Simulationen eines von Dr. Leonid Yelash angefertigten Programms verwendet. An dieser Stelle wird die x -Dimension in y -Richtung geschert.

In Abbildung 9.2 sind sowohl die analytische Lösung als auch die Simulationsergebnisse eingezeichnet. Die durchgezogene Linie entspricht der Theoriekurve, während die Punkte ermittelte Daten darstellen.

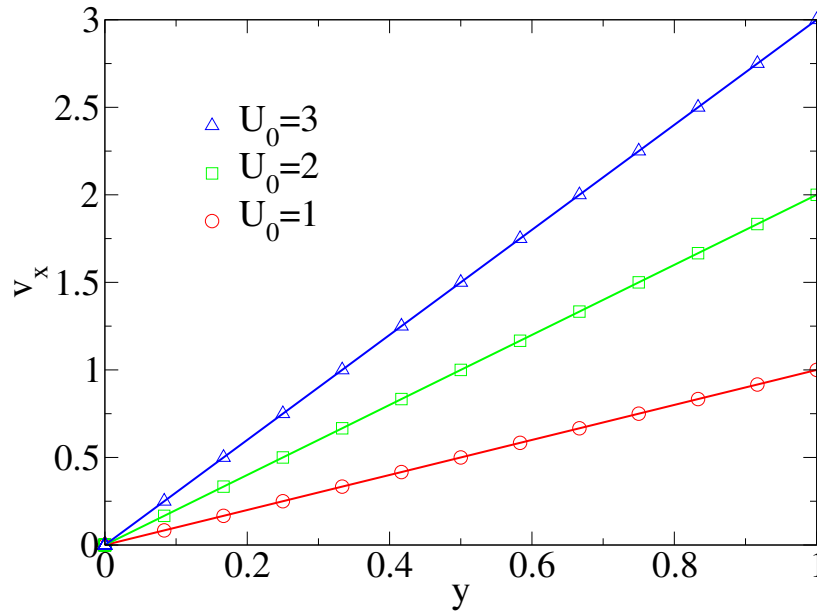


Abb. 9.2: Geschwindigkeitsprofil des Couette-Flusses in zwei Dimensionen mit der Offline-Methode. Die angelegte Geschwindigkeit der Kanalwand wird variiert von $U_0 = 1, 2$ zu $U_0 = 3$. Es ist ersichtlich, dass die Datenpunkte, die den Symbolen entsprechen, die analytische Lösung in Form der durchgezogenen Linien exakt beschreiben. Die hier gezeigten Daten sind in [2] veröffentlicht worden.

Wie gut zu erkennen ist, stimmen die Simulationsergebnisse des gekoppelten Hybrid-schemas mit der analytischen Lösung überein. Es ergibt sich eine dynamische Viskosität von $\eta = \frac{\sigma_{12}}{\dot{\gamma}}|_{\dot{\gamma}=1} = 1.96481$. An dieser Stelle wird, da es sich um zweidimensionale Daten und einen Stresstensor σ^{2D} der Größe 2×2 handelt, die σ_{12}^{2D} -Komponente des ermittelten Stresstensors zur Bestimmung der Viskosität verwendet.

9.2 POISEUILLE-FLUSS MIT MONOMEREN

Beim Poiseuille-Fluss wird keine Scherung, sondern ein Druckprofil angelegt. Das Simulationsgebiet ist in diesem Fall $\Omega = [0, 1] \times [-1, 1] \times [-1, 1]$ und wird im Vergleich zum zweidimensionalen Fall im vorherigen Abschnitt in z -Richtung erweitert. Auch hier wird eine Haftbedingung an die Wände bei $x = 0$ und $x = 1$ gestellt. An diesen Wänden ist nach Druckprofil die Geschwindigkeit $u = 0$. Der angelegte Druckgradient in z -Richtung hat die Form:

$$f_z = -2\mu \frac{P}{Re} \quad (9.3)$$

mit der Reynoldszahl $Re = 1$. P entspricht dem angelegten Druck. Im dreidimensionalen Fall wird das angelegte Gitter auf $3 \times 3 \times 3$ erweitert. Die Strömung ist in y - und z -Richtung symmetrisch. Ein Schaubild des Poiseuille-Flusses ist in Abbildung 9.3 dargestellt. Die Lösung des Scherprofils sind biquadratische Polynome. Entsprechend wird der eingestellte Polynomgrad auf $k = 2$ gesetzt. Die Adams-Bashforth-Methode [54] wird wie in Gleichung (5.25) als Zeitintegrator angewendet.

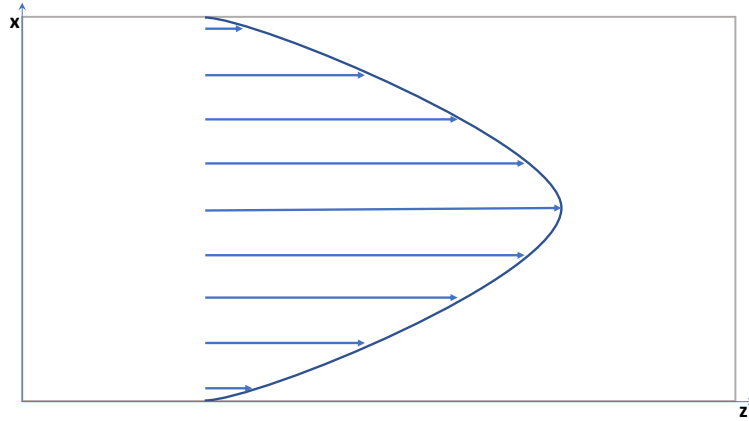


Abb. 9.3: Darstellung des Poiseuille-Flusses. Bei dieser Art der Strömung werden die Komponenten des Kanals konstant gehalten. Ein Druckprofil wird angelegt. Dadurch entsteht ein quadratisches Geschwindigkeitsprofil im Inneren des Kanals, wie im Bild angedeutet.

In Abbildung 9.4 sind die Ergebnisse gezeigt. Die durchgezogene Linie entspricht wieder der analytischen Lösung, die in diesem Fall

$$u_z = x(1 - x)P \quad (9.4)$$

lautet. x ist in dieser Gleichung die x -Koordinate innerhalb des Kanals. Der Druck P wurde zwischen $P = -3$ bis $P = 6$ variiert.

Mithilfe der Scherspannung lässt sich die dynamische Viskosität zu $\eta = \frac{\sigma_{13}}{\dot{\gamma}}|_{\dot{\gamma}=0.5} = 1.61776$ bestimmen. Wieder stimmen die Daten der Hybridsimulation sehr gut mit der analytischen Lösung überein.

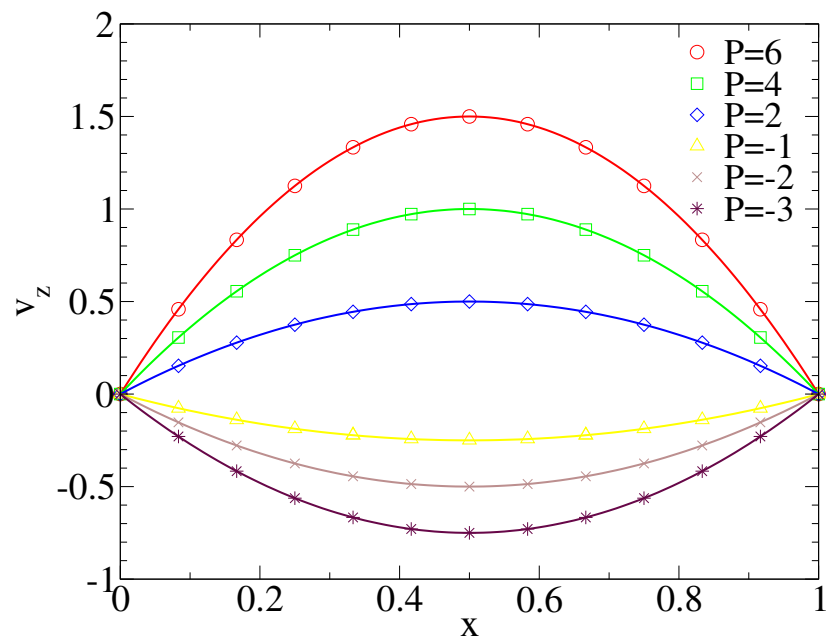


Abb. 9.4: Geschwindigkeitsprofil des Poiseuille-Flusses mit der Offline-Methode. Für unterschiedlich eingestellten Druck P stellt sich das Geschwindigkeitsprofil ein und reproduziert die Werte der analytischen Lösung. Die hier gezeigten Daten sind in [2] veröffentlicht worden.

Teil V

ZUSAMMENFASSUNG UND FAZIT

ZUSAMMENFASSUNG UND FAZIT

Simulationen zu Eigenschaften realer Materialien reduzieren ein komplexes Problem entsprechend der Fragestellung auf eine möglichst minimale Anzahl an Freiheitsgraden. Dabei gehen auf der einen Seite aufgrund der Vergrößerung des Systems Informationen auf der Mikroebene verloren, auf der anderen Seite ist ein sehr detailliertes Modell wenig effizient und bietet nur auf einer Mikroebene Ergebnisse. Nur kleine Längen- wie auch Zeitskalen, die von der Simulationsart abhängen, können von dieser Mikroebene in endlicher Rechenzeit simuliert werden.

Hybride Simulationsschemata haben zum Ziel, unterschiedliche Skalen miteinander zu verbinden. Auf diese Weise können Informationen der Mikroebene in der Makroebene verwendet und somit akkurate Zeit- und Längenskalen simuliert werden. Wichtig bei diesem Hybridschema, der Heterogenen Multiskalen-Methode (HMM), ist dabei, dass die beiden miteinander zu verbindenden Größenskalen voneinander unabhängig sind, also Vorgänge auf der Makroebene die Mikroebene nicht weiter beeinflussen und umgekehrt. Die Mikroebene bildet dabei den Flaschenhals der Hybridsimulationen, da für jeden Zeitschritt der CFD-Simulation eine Mikrosimulation gestartet werden muss.

Ziel dieser Arbeit war, die Simulationsparameter der Mikroebene in Form eines funktionierenden Molekulardynamik-Programms zu optimieren und auf diese Weise die benötigten Informationen der Mikroebene der Makroebene zur Verfügung zu stellen. Dazu wurde das von Dr. Alexander Winkler aufgesetzte MD-Programm, das Basis seiner Dissertation [36] war, weiterentwickelt und auf die Bedürfnisse einer Hybridsimulation angepasst. In dieser Arbeit konnten erfolgreich Systeme aufgesetzt werden, deren untersuchte Ergebnisse in Verbindung mit der Makroebene theoretische Werte bestätigen konnten.

In diesem Kapitel werden die bisherigen Ergebnisse zusammengefasst. Dabei wird zunächst auf die Ergebnisse des Hartkugel-Systems eingegangen, mit dem monodisperse, kugelförmige Kolloide simuliert werden. Anschließend wird auf die Ergebnisse der Polymersysteme eingegangen, bei denen einzelne Monomere mithilfe von finiten Federn verbunden werden. Zum Abschluss wird ein Fazit gegeben, das mit einem Ausblick abgerundet wird.

Das Monomersystem besteht aus harten, kugelförmigen Kolloiden, welche über ein WCA-Potential simuliert werden. Zur Beschleunigung der MD-Simulation wurden verschiedene Parameter untersucht und optimiert. Dabei ist zu vermerken, dass WCA-Teilchen ein Newtonsches Fluid nachstellen, ihre Viskosität ist also weitgehend unabhängig von der Scherrate. Dieses Verhalten konnte in dieser Arbeit reproduziert werden.

Zur Optimierung unter dem Aspekt der Hybridsimulation wurden in Abschnitt 6.1 zwei unterschiedliche Thermostate miteinander verglichen. Da die Viskosität eines mit dem Lowe-Andersen-Thermostat temperierten Systems sehr anfällig für die beiden Thermostatvariablen ist, dem Interaktionsradius r_C und der Kollisionsfrequenz Γ , wie in den Abbildungen 6.3 und 6.5 gezeigt, eignet es sich hervorragend, um Flüssigkeiten zu simulieren, deren Viskosität eingestellt werden kann. Entsprechend ist das Lowe-Andersen-Thermostat eine ressourcensparende Alternative für hydrodynamische Wechselwirkun-

gen, wie z. B. durch Vielteilchendynamik (engl.: *Multi-particle Collision Dynamics* (MPCD)) realisierbar.

Da bei Hybridsimulationen der Steady State interessant ist und nicht der Prozess, wie das System in diesen Zustand gelangt, welcher z. B. in [77] untersucht wurde, ist es ausreichend, ein (unphysikalisches) ressourcensparendes und effizientes Thermostat zu verwenden, das isokinetische Thermostat. Aufgrund der Zeitersparnis wird dieses Thermostat für die geplanten Online-Hybridsimulationen gewählt.

Neben der Wahl des Thermostats ist unter anderem der Zeitschritt der Simulation ein wichtiger Parameter zur Beschleunigung. Je größer der Zeitschritt, desto mehr Schritte können in einer vorgegebenen Simulationszeit erfolgen. Der Abstand der Datennahmen sollte dabei minimal größer als die Autokorrelationszeit der Daten sein. Wie in Abschnitt 6.2 gezeigt, genügt es, einen Zeitschritt von $\Delta t = 10^{-3}$ zu wählen.

Da die Simulationszeit nicht ganz linear mit der Teilchenzahl N skaliert, wie in Kapitel 6.2 und Tabelle 6.1 gezeigt, ist es sinnvoller, eine triviale Parallelisierung in Form von mehreren kleinen Simulationen durchzuführen, als eine große Simulationsbox zu parallelisieren. Hierfür ist es notwendig, eine minimale Boxgröße zu finden, die nicht unterschritten werden darf. Diese minimale Länge, ab der Finite-Size-Effekte auftreten, beträgt 4σ . Die Simulationsbox wird in zukünftigen Hybridsimulationen auf eine Grundfläche von $10 \times 10\sigma^2$ in x - und z -Richtung festgelegt. Bei einer Teilchenzahl von $N = 600$ wird die y -Länge variiert, um der gewählten Dichte zu genügen. Die quadratische Grundfläche ist dabei notwendig, da sich die Ergebnisse bei unterschiedlichen x - und z -Längen der Box nicht mehr sinnvoll vergleichen lassen. Zwar ist das Scherprofil nur von der Scherrate $\dot{\gamma}$ abhängig, die Schergeschwindigkeit ändert sich jedoch abhängig von der Boxdimension, wodurch die Ergebnisse nicht gut vergleichbar sind.

Die einzelnen Komponenten des für die Hybridsimulation relevante Stresstensors sind ebenfalls von der angelegten Scherrate $\dot{\gamma}$ abhängig. Dabei ist die σ_{13} -Komponente bei Newtonschen Fluiden linear zur Scherrate $\dot{\gamma}$. Entsprechend sinkt der relative Fehler der Viskosität η , wenn bei hohen Scherraten simuliert wird. Dabei kann die Simulationszeit zur Reduzierung des Fehlers um eine Größenordnung um $\mathcal{O}(10^2)$ reduziert werden.

Durch die Senkung der Parameterordnung, wie in Abschnitt 8.2 beschrieben, genügen zu Beginn vier Simulationen bei unterschiedlichen Scherraten. Zur besseren Statistik und der Abschätzung der Güte der Fits der Datensätze wird bevorzugt bei den ressourcengünstigen, hohen Scherraten simuliert. Danach muss zur anschließenden Verfeinerung des Fits an den entscheidenden Stellen des Phasenraums eine erneute MD-Simulation gestartet werden.

Bei dem dieser Arbeit zugrundeliegenden Projekt wird eine Hybridsimulation entwickelt, die zwei unterschiedliche Skalen miteinander verknüpft. Dabei sollen die beiden Programmebenen weitestgehend unabhängig voneinander arbeiten. Lediglich die Randbedingungen für die MD-Simulation der Mikro-Ebene werden über ein Interface von der Makro-Ebene übertragen. Außerdem werden die Ergebnisse der MD-Simulation über dieses Interface zurück an die CFD-Simulation geleitet. Durch diese Kommunikation zwischen den beiden Programmen können die einzelnen Teile autark voneinander implementiert und getestet werden. Das Nadelöhr der Simulationen bildet dabei die MD-Simulation. Im Laufe des Projektes wurden neben der Parameteroptimierung weitere Methoden von Dr. Nehzat Emamy und Dr. Leonid Yelash entwickelt, um den Aufwand der Simulationen so gering wie möglich zu halten.

Um die MD-Simulationen möglichst einfach zu halten, werden die Simulationsboxen an den Positionen des Makrogebietes entlang der Flusslinien rotiert, wodurch ein eindimensionaler Scherfluss der x -Komponente in z -Richtung entsteht. Diese Methode wird in Abschnitt 8.1 vorgestellt. Zusätzlich wird durch das Anpassen der Daten des Stressensors $\sigma_{\alpha\beta}$ in Abhängigkeit der Scherrate $\dot{\gamma}$ (Senkung der Parameterordnung), wie in Kapitel 8 beschrieben, der Aufwand der notwendigen Simulationen auf der Mikroebene reduziert, da nicht in jedem Zeitschritt für einen noch nicht behandelten Parametersatz eine (beziehungsweise mehrere) MD-Simulationen gestartet werden müssen.

Bevor die Daten gefittet werden, wird eine Eigenwertdekomposition durchgeführt, um an die Daten nicht 6 Gleichungen, sondern nur zwei Funktionen an $f(\lambda)$ und $\sigma_{22}(\dot{\gamma})$ zu fitten, wobei λ die Eigenwerte des Verzerrungsgeschwindigkeitstensors \mathbf{E} bilden und $\lambda_i = \{-\dot{\gamma}/2, 0, \dot{\gamma}/2\}$ sind. Dies ist zur Zeit nur für eine reine Scherung möglich, die Eigenwertdekomposition wird auf verschiedene Scherarten erweitert, um allgemeine Scherungen beschreiben zu können.

Dabei sei angemerkt, dass diese Eigenwertdekomposition eigentlich einen reinen Scherfluss beschreibt. Bei WCA-Teilchen, die durch Rotation keine Deformation oder Anordnung im Scherfluss erfahren, ist diese Beschreibung adäquat, auch wenn die Differenz zwischen σ_{11} und σ_{33} nicht in guter Näherung beschrieben werden kann.

In Kapitel 9 wurde anhand der in [2] präsentierten Ergebnisse gezeigt, dass die Hybrid-simulation die analytische Lösung einfacher Strömungsprobleme reproduzieren kann. Die Kopplung von Mikro- und Makro-Skala führt zu korrekten Ergebnissen und ist entsprechend eine hervorragende Methode, auch komplexere physikalische Strömungsprobleme zu untersuchen.

Bei Simulationen einer vorgegebenen Geometrie einer Flüssigkeit sind zur Vergleichbarkeit unterschiedlicher Mechanismen und Systeme verschiedene Fluide interessant, die den Simulationen zugrunde liegen. Dazu zählen zum einen Fluide mit unterschiedlichen Materialeigenschaften, wie der Viskosität. Diese kann in der Simulation zum Beispiel über Vielteilchendynamik (MPCD) realisiert sein. Aus Effizienzgründen kann dies in der Hybridsimulation über das Lowe-Andersen-Thermostat eingestellt werden. Zum anderen sind Nichtnewtonsche Fluide von Interesse, d. h. Fluide, deren Viskosität nicht annähernd unabhängig von der Scherrate ist.

Ein solches Nichtnewtonsches Verhalten weisen z. B. Polymerketten auf, die nicht als Kugel angenähert werden können. Das Shear Thinning tritt bei Polymerketten bei relativ hohen Scherraten auf, wie in Abbildung 7.6 a) aufgezeigt. Dabei ist der Effekt sowohl von der Scherrate $\dot{\gamma}$ als auch von Kettenlänge N_p und der Monomerdichte ρ abhängig, was in Abbildung 7.6 a) und b) demonstriert wird.

Der Umstand der Abhängigkeit der Viskosität von der Kettenlänge bei der Simulation von Polymeren, das Shear Thinning, muss beim Fit der Tikhonov-Polynome an die Simulationsdaten berücksichtigt werden. Dadurch wird sich der Polynomgrad erhöhen. Wie zuvor bei der Erweiterung auf die Dichte ρ in Abschnitt 8.2.2 muss zudem eine Erweiterung auf eine zusätzliche Dimension für N_p durchgeführt werden.

Problematisch in späteren Simulationen möglicher komplexer Geometrien wie einem 4 – 1- oder einem H-Kanal ist bei Polymerketten, dass sie sich bei simpler Scherung, die zur Zeit realisiert wird, entlang der Scherung ausrichten, sich aber zusätzlich aufgrund des Beitrags der reinen Rotation zum Verzerrungsgeschwindigkeitstensor \mathbf{E} immer wieder drehen, wie in Abbildung 7.8 gezeigt. Im Fall der reinen Scherung, die bei den Übergangspunkten solcher komplexeren Geometrien vorliegen würde, liegt ein anderes Verhalten

vor. Bei dieser Form der Scherung wird das Simulationsgebiet von zwei gegenüberliegenden Seiten zusammengepresst, bis aus einem quadratischen Gebiet (in zwei Dimensionen) ein immer schmaler werdendes Rechteck wird.

Die beiden Scherarten und der der reinen Scherung zugrundeliegende Algorithmus zur Simulation sind in [70] thematisiert. Um komplexe Geometrien mit Polymerketten behandeln zu können, muss eine weitere Simulationmethode der MD-Simulation implementiert werden. Außerdem muss der Fit der Eigenwertdekomposition auf einen zusätzlichen Tensor erweitert werden, der die einfache Scherung mitberücksichtigt. Zudem müssen wie in [69] ein zusätzlicher Tensor sowie zusätzliche Parameter λ_i hinzugefügt werden, wobei $i = \{0, \dots, 3\}$, um allgemeine Scherungen beschreiben zu können.

Insgesamt lässt sich sagen, dass die Parameter der Simulationen mit Molekulardynamik als Grundlage angemessen optimiert wurden, um ein hybrides Simulationsschema aus MD- und CFD-Simulation zu entwickeln. Dabei wird auf mehrere kleine Systeme mit einer Teilchenzahl $N \approx 1\,000$ in einer Simulationsbox mit Seitenlängen $L_i > 5\sigma$ bei einem Zeitschritt $\Delta t = 10^{-3}$ bei hohen Scherraten gearbeitet.

Neben den bisher betrachteten Systemen der Kugelkolloide und der gelösten Polymerketten stellen auch Simulationen von aktiven Mikroschwimmern [78, 79] einen interessanten Gegenstand der Untersuchungen mittels der Strömungssimulation dar. Auch Gemische verschiedener Teilchensorten besitzen eine von den bisherigen Simulationen unterschiedliche Dynamik. Diese Systeme sowie kristallbildende Systeme oder Systeme nahe einem kritischen Punkt könnten Gegenstand zukünftiger MD-Simulationen sein.

Eine weitere Optimierung könnte durch Umstellen der Programmierbasis von einem von Alexander Winkler für seine Doktorarbeit [36] und intern in der Gruppe *Komet331* weitergeführten C++-Code auf das Softwarepaket *ESPReso++* [80] erreicht werden, das am Max-Planck-Institut für Polymerforschung (MPIP) in Mainz entwickelt wird. Da dieses Softwarepaket wie das zur Zeit entwickelte Interface zwischen den beiden Skalen über ein Benutzer-Interface in der Programmiersprache *Python* angesprochen werden kann und auf einem C++-Kern basiert, könnte die Kommunikation zwischen den beiden Skalen vereinfacht werden.

Auf der Makro-Ebene sollen komplexe Geometrien wie der 4 – 1- und der H-Kanal realisiert werden. Diese komplexen Geometrien erlauben die Untersuchung der Dynamik der betrachteten Flüssigkeit, sowie eine Analyse der Durchmischung zweier unterschiedlicher Stoffe.

Vor allem bei der Untersuchung der Phasenseparation müssen die hier vorgestellten Navier-Stokes-Gleichungen um einen Spannungsterm ergänzt werden, der die Oberflächenspannung zweier unterschiedlicher Phasen berücksichtigt. Zudem sollen die Gleichungen von inkompressiblen Fluiden auf kompressible Gase erweitert werden.

Dr. Leonid Yelash und Dr. Kai Werth haben bereits ein Skript in *Python* ausgearbeitet, welches als Master die beiden Skalen ansprechen und Informationen von einer Skala zur anderen leitet. Da mithilfe der Programmiersprache *Python* gearbeitet wird, die Mikroskala in C++ und die Makroskala in C# geschrieben ist, bedeutet dies viel Programmieraufwand, um eine problemlose Kommunikation zu gewährleisten. Diese Schnittstelle wird in naher Zukunft optimiert werden.

Teil VI

APPENDIX

MODELLSYSTEM

A.1 VERGLEICH MIT VORHERIGEN ERGEBNISSEN

Um die Funktionalität des Programms zu überprüfen, war es notwendig, die Ergebnisse mit bisherigen Ergebnissen unabhängiger Quellen zu vergleichen. In diesem Abschnitt werden die Ergebnisse dieses Vergleichs gezeigt. Dabei wurde zum einen das Monomersystem, zum anderen ein Polymersystem mit bereits veröffentlichten Daten verglichen.

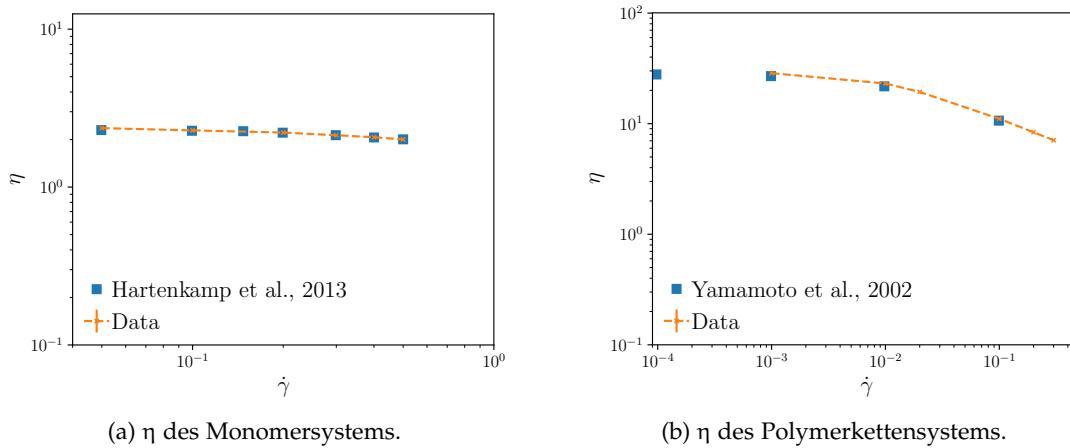


Abb. A.1: Vergleich von η mit unabhängigen Quellen. Dabei wird in (a) die Viskosität des Monomersystems bei einer Dichte von $\rho = 0.8442$ bei einer Temperatur von $k_B T = 0.722$ im Vergleich zu Daten von Hartenkamp et al., [81], gezeigt. Die Teilchenzahl ist dabei $N = 512$. Darüber ergibt sich eine Seitenlänge von $L = 8.464634$. Das System wurde mit dem isokinetischen Thermostat gekühlt. Die Daten stimmen gut überein. In (b) wird die Viskosität des Polymersystems mit Daten von Yamamoto et al., [68], verglichen. Das Polymersystem hat eine Monomerdichte von $\rho = 1.0$ und eine Kettenlänge von $N_P = 10$. Die Temperatur beträgt $k_B T = 1.0$. Auch hier stimmen die Daten gut überein.

In Abbildung A.1 (a) ist die Viskosität η des Monomersystems bei einer Temperatur von $k_B T = 0.722$ und einer Dichte von $\rho = 0.844$ im Vergleich zu Daten von Hartenkamp et al., [81], gezeigt. Das System wurde mit dem isokinetischen Thermostat gekühlt. Das Scherprofil wird sowohl durch die Lees-Edwards-Randbedingungen, als auch den SLLOD-Algorithmus induziert. Wie man sieht, stimmen die Daten gut überein. In Abbildung A.1 (b) wird die Viskosität des Polymersystems mit Daten von Yamamoto et al., [68], verglichen. Das Polymersystem bei einer Monomerdichte von $\rho = 1.0$ enthält 100 Ketten mit je einer Kettenlänge von $N_P = 10$. Die Temperatur beträgt $k_B T = 1.0$. Auch hier stimmen die Daten gut überein und der Verlauf des Ausdünnungseffekts des Shear-Thinnings ist reproduzierbar.

A.2 POLYMERKETTEN

Simulationsergebnisse der Polymerkettensysteme, die über den Rahmen dieser Arbeit hinausgehen, sind in diesem Abschnitt aufgeführt. So sind zum einen die Ergebnisse des Vergleichs verschiedener Kettenlängen bei einer Dichte von $\rho = 0.4$ gezeigt, zum anderen Ergebnisse der Variation der Federkonstante k . Dieser ist ein Parameter des FENE-Potentials und wurde in der Arbeit, wenn nicht anders erwähnt, auf $k = 30$ gesetzt.

A.2.1 Dichte $\rho = 0.4$

Neben den Systemen mit einer Monomerdichte von $\rho = 0.8$ wurden auch Simulationen bei einer Monomerdichte von $\rho = 0.4$ durchgeführt. In diesem Abschnitt werden die Ergebnisse gezeigt.

Die hier vorgestellten Ergebnisse sind bei einer Monomerdichte von $\rho = 0.4$ aufgenommen. Die Simulationsbox hat eine Ausdehnung von $V = 10 \times 10 \times 10 \sigma^3$ mit 40 Polymerketten mit je einer Kettenlänge von $N_P = 10$. Die Anzahl an Polymerketten für die übrigen Kettenlängen folgt analog.

Das System wird mit unterschiedlichen Scherraten geschert und dabei mithilfe des isokinetischen Thermostats gekühlt. Zur korrekten Einstellung der Scherprofile wird der SLLOD-Algorithmus zu Hilfe genommen.

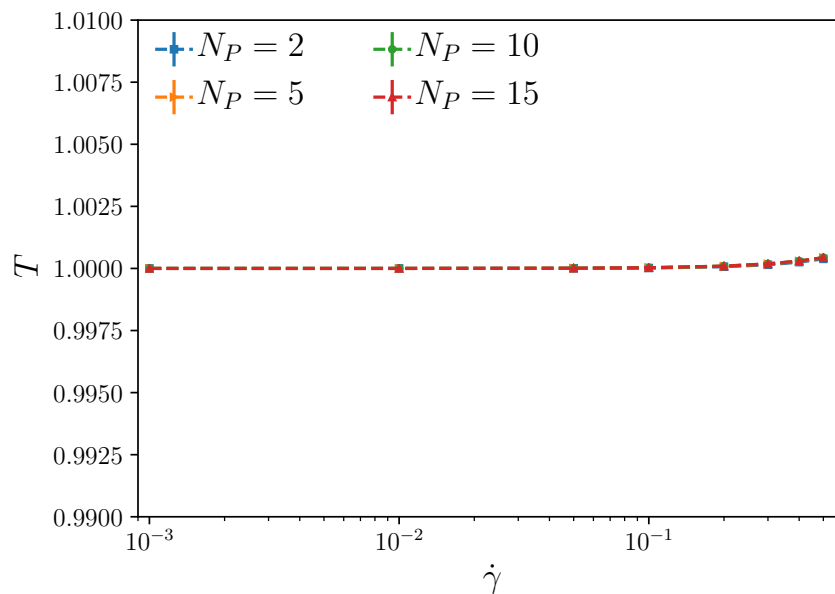


Abb. A.2: Temperaturverlauf für Polymerketten unterschiedlicher Kettenlänge N_P in Abhängigkeit der Scherrate $\dot{\gamma}$ bei einer Monomerdichte von $\rho = 0.4$. Wie bei den Monomersystemen und dem im Hauptteil der Arbeit betrachteten System mit $\rho = 0.8$ ist auch hier lediglich eine minimale Temperaturerhöhung durch erhöhte Scherung zu verzeichnen. Der Temperaturanstieg ist dabei sogar unter 0,01 %. Dies ist auf die geringe Dichte zurückzuführen.

Zunächst wird der Temperaturverlauf der Simulationen für Polymerketten betrachtet. Die ist notwendig, um auszuschließen, dass die Kettenlänge der Polymere einen Einfluss auf die Temperatur des Systems hat.

In Abbildung A.2 ist der Temperaturverlauf für Kettenlängen $N_P = 2, 5, 10$ und 15 in Abhängigkeit der Scherrate dargestellt. Im Vergleich zum Temperaturanstieg bei einer Monomerdichte von $\rho = 0.8$ in Abbildung 7.4 in Abschnitt 7.3 ist dieser bei der niedrigeren Dichte noch geringer und liegt bei unter 0.01% bei hohen Scherraten $\dot{\gamma} \geq 0.4$.

Zudem erkennt man auch hier, dass die Temperatur nicht durch die Kettenlänge beeinflusst wird. Die einzelnen Kurven liegen gut übereinander.

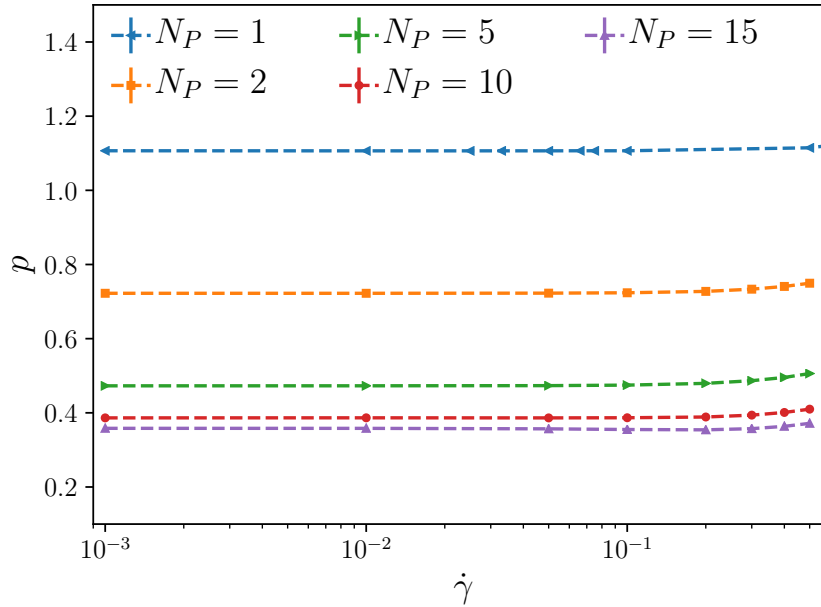


Abb. A.3: p in Abhängigkeit der Scherrate für ein System konstanter Dichte ($\rho = 0.4$) für verschiedene Kettenlängen $N_P = 1, 2, 5, 10$ und 15 . Auch hier ist deutlich zu sehen, dass der Druck mit zunehmender Kettenlänge abnimmt.

Der in Abbildung A.3 dargestellte Druck p weist auch für diese Dichte eine leichte Abhängigkeit von der Scherrate $\dot{\gamma}$ auf. Je höher die Kettenlänge ist, desto niedriger ist der Druck, wie zu erwarten. Dies ist auf die geringere Polymerdichte zurückzuführen. Die Druckerhöhung für steigende Scherraten $\dot{\gamma} \geq 0.3$ ist für höhere Kettenlängen $N_P \geq 5$ signifikanter.

In Abbildung A.4 werden in (a) σ_{13} und in (b) die davon abgeleitete Viskosität η in Abhängigkeit der Scherrate $\dot{\gamma}$ gezeigt. Dabei werden verschiedene Kettenlängen $N_P = 1, 2, 5, 10$ und 15 simuliert. Aufgrund der niedrigeren Dichte ist der Effekt des *Shear-Thinnings* erst ab einer Kettenlänge von $N_P = 10$ bei hohen Scherraten deutlich zu erkennen. Bei niedrigerer Kettenlänge ist der Effekt kaum wahrnehmbar.

Wie zuvor auch erscheint es, dass die Viskosität bei hohen Scherraten $\dot{\gamma} \geq 0.3$ auf den Wert des Monomersystems konvergiert.

Auch hier kann man beobachten, dass niedrige Scherraten zu einem hohen Fehler in der Viskosität führt, obwohl die gleiche Statistik vorliegt und die Simulation unter gleichen Bedingungen stattgefunden haben. Bei dieser niedrigen Dichte liegen Dimere (also Polymerketten mit einer Kettenlänge von $N_P = 2$, auch Hanteln genannt) und Monomere sehr dicht beieinander, gerade durch den Shear-Thinning-Effekt.

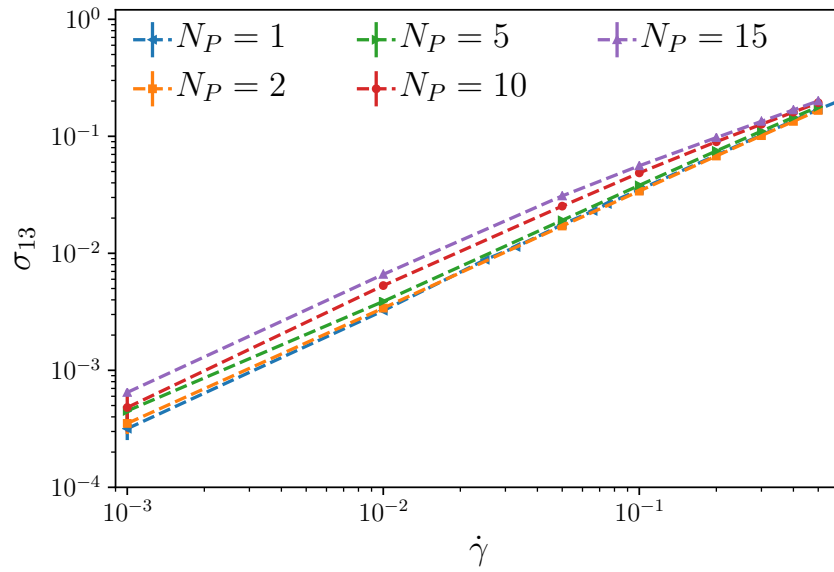
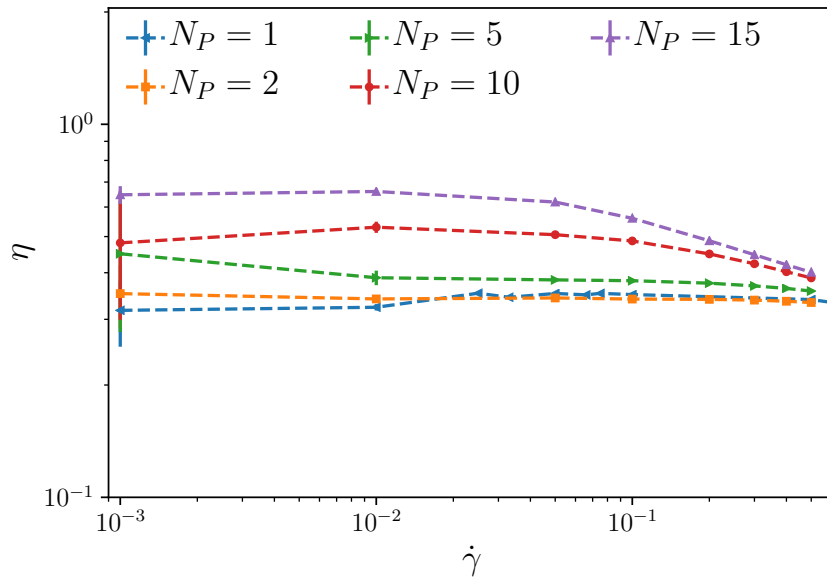
(a) σ_{13} der Polymerkettensysteme für $\rho = 0.4$.(b) η der Polymerkettensysteme für $\rho = 0.4$.

Abb. A.4: (a) σ_{13} und (b) η in Abhängigkeit der Scherrate für ein System konstanter Dichte ($\rho = 0.4$) für verschiedene Kettenlängen $N_P = 1, 2, 5, 10$ und 15 . Aufgrund der niedrigeren Dichte ist der Effekt des *Shear Thinnings* erst ab einer Kettenlänge von $N_P = 10$ deutlich zu erkennen.

A.2.2 Variation des Parameters k im FENE-Potential

Ein Parameter, der im Rahmen dieser Doktorarbeit nur eine geringe Rolle spielt, ist die Federkonstante k im FENE-Potential. Erhöht man den Parameter k , so ist die Ruhelage der Monomere und damit der ideale oder bevorzugte Abstand der Monomere zueinander minimiert.

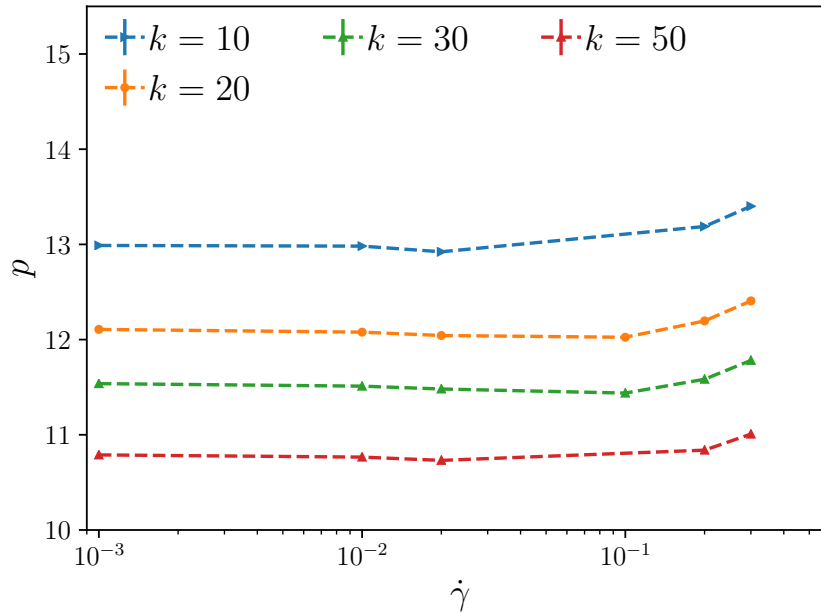


Abb. A.5: Druck p in Abhängigkeit der Scherrate für verschiedene Federkonstanten $k = 10, 20, 30$ und 50 . Aufgrund des niedrigeren bevorzugten Abstands der Polymerketten und den damit verbundenen Freiraum im Fluid ist der Druck für höhere Werte von k geringer. Der Verlauf des Drucks selbst ist dabei unabhängig von k .

In Abbildung A.5 ist der Druck p gegen die Scherrate $\dot{\gamma}$ für verschiedene Federkonstanten k aufgetragen. Beim Druck zeigt sich deutlich, dass eine Kette mit höherer Federkonstante einen geringeren Druck aufweist. Dieser Effekt ist bei σ_{13} und der Viskosität η weniger ausgeprägt, wie in Abbildungen A.6 und A.7 zu erkennen.

Während die Linearität von σ_{13} bezüglich der Scherrate aufgrund des Shear Thinnings in dem simulierten Bereich nicht zu sehen ist, weisen alle simulierten Federkonstanten einen annähernd parallelen Verlauf für hohe Scherraten auf. Erst bei niedrigeren Scherraten ist der Effekt der Federkonstante relevant und ändert den Verlauf der Viskositätskurve.

Dabei liegen die beiden Kurven für $k = 20$ und $k = 30$ auch in diesem kleinen Scherratenbereich annähernd parallel, während der Unterschied zu Systemen mit anderen Federkonstanten signifikanter ist. Eine hohe Federkonstante von $k = 50$ weist den höchsten Unterschied in Form eines Einknickens der Kurve auf. In diesem Bereich sollten weitere Simulationen durchgeführt werden, um den Verlauf näher zu untersuchen.

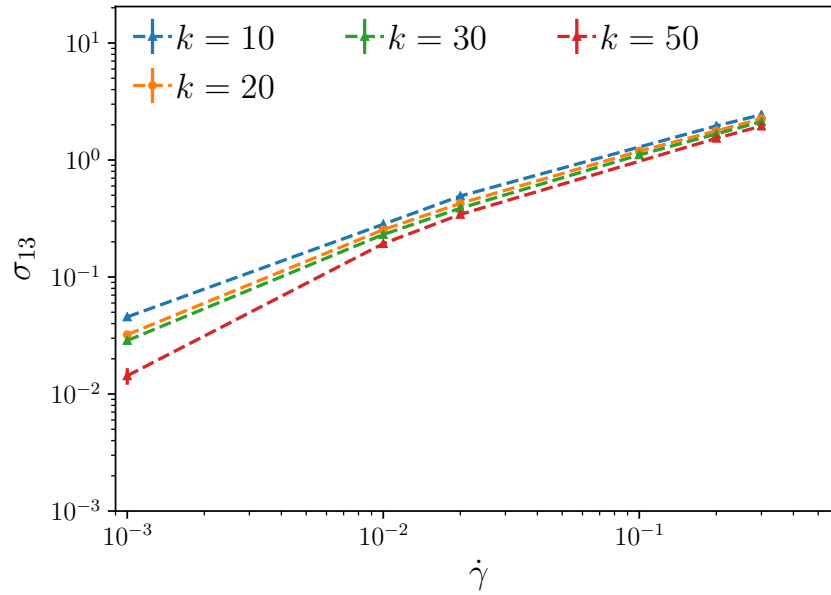


Abb. A.6: Stresstensorkomponente σ_{13} in Abhängigkeit der Scherrate für verschiedene Federkonstanten $k = 10, 20, 30$ und 50 . Aufgrund der geringeren Ausdehnung der Polymerketten ist σ_{13} für höhere Werte von k geringer. Der Unterschied zwischen den Kurven fällt dabei geringer aus als beim Druck in Abbildung A.5. Der Verlauf der Stresstensorkomponente ist ebenfalls unabhängig von k .

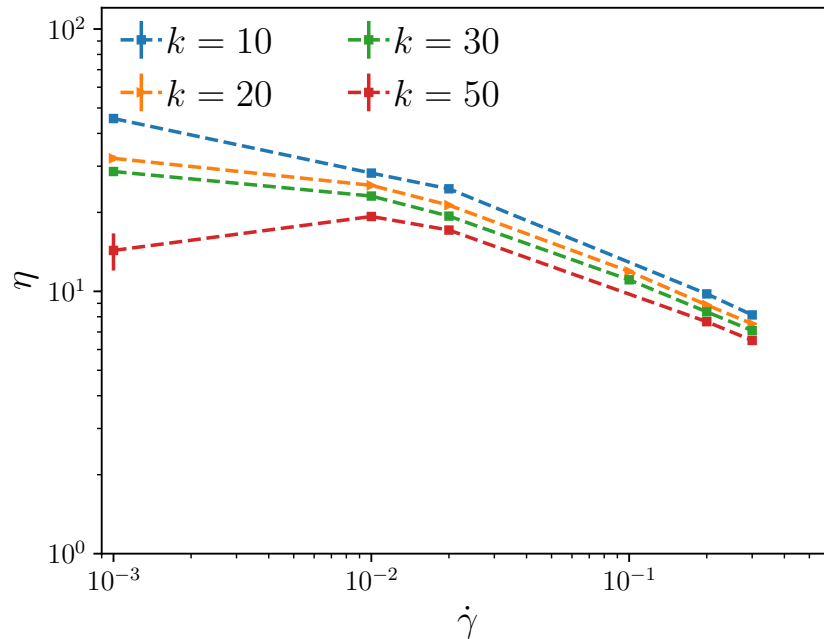


Abb. A.7: η in Abhängigkeit der Scherrate für verschiedene Federkonstanten $k = 10, 20, 30$ und 50 . Aufgrund der geringeren Ausdehnung der Polymerketten ist die Viskosität für höhere Werte von k geringer. Der Verlauf der Viskosität ist dabei unabhängig von k . Wie bei den vorherigen Ergebnissen in Abschnitt 7.4 ist das Shear-Thinning ab einer Scherrate von $\dot{\gamma} \approx 0.001$ ersichtlich. Die Wahl der Federkonstante beeinträchtigt bei den untersuchten Werten den Verlauf der Viskosität kaum. Für hohe Scherraten nähern sich die beiden Kurven stärker an als für niedrige Scherraten.

Eine einfache Alternative zur diskontinuierlichen Galerkin-Methode bieten Finite-Differenzen-Verfahren. Einige Testreihen um schnelle Ergebnisse zu erhalten, unter anderem in den Master-Arbeiten von Sergej Schelle und Hannah Vogel [82, 83], verwenden ein gestaffeltes Gitter, wie von [84] in der MAC-Methode vorgeschlagen wurde. Da dieses Gitter zwar nicht explizit in der vorliegenden Arbeit Verwendung fand, in Vorarbeiten aber angewendet wurde, ist sie aus Gründen der Vollständigkeit an dieser Stelle erwähnt.

In dieser Methode werden Variablen nur an bestimmten, vordefinierten Punkten des Gitters bestimmt. Die Gitterpunkte bei diesem Finite-Differenzen-Verfahren sind dabei (x_i, y_j) , wobei die Zählvariablen i und j von 1 bis n_x bzw. n_y gehen und die x - bzw. y -Position im Gitternetz anzeigen. Die Gitterpunkte stellen demnach die Koordinaten der Zellecken des Gitters dar. Die in der dG-Methode verwendete Ortsdiskretisierung (vgl. Abschnitt 5.2.2) gilt also nicht.

Der Druck p wird bei der Lösung der Navier-Stokes-Gleichungen in den Zellmitten bestimmt, also bei $(x_{i+\frac{1}{2}}, y_{j+\frac{1}{2}})$. Für den Druck werden am Rand des Gebiets $\partial\Omega$ Neumann-Bedingungen vorgegeben. Die x -Komponente der Geschwindigkeit wird an der Stelle $(x_i, y_{j+\frac{1}{2}})$ berechnet, die y -Komponente bei $(x_{i+\frac{1}{2}}, y_j)$. Dabei gelten für die Geschwindigkeiten am Rand des Gebiets $\partial\Omega$ Dirichlet-Randbedingungen. Dieses Schema ist in Abbildung B.1 dargestellt.

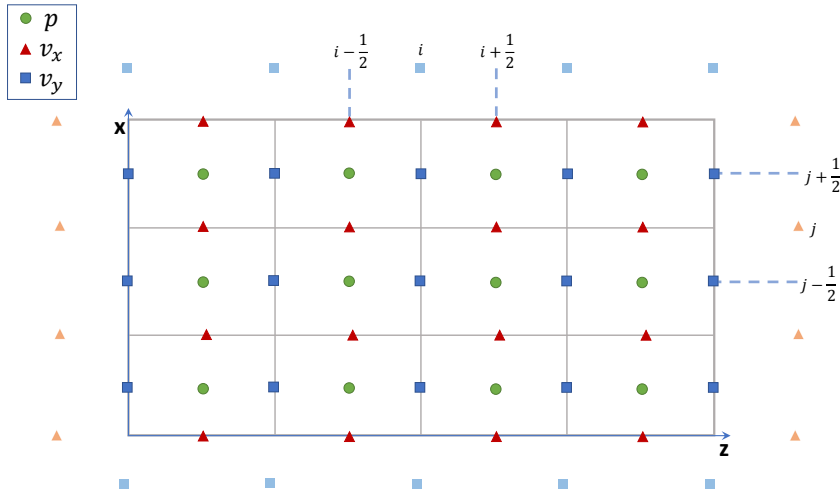


Abb. B.1: Das gestaffelte MAC-Gitter. Bei dieser Methode werden die Geschwindigkeiten in x - und z -Richtung sowie der Druck nur an bestimmten, vordefinierten Punkten des Gitters berechnet. i und j sind dabei Zählvariablen in x - und z -Richtung und bezeichnen z.B. den i -ten Gitterpunkt in z -Richtung.

Durch die Staffelung des Gitters liegen einige Punkte nur randnah, nicht direkt auf dem Rand, wie von den Dirichlet-Randbedingungen vorgegeben. Um diese gewährleisten zu können, werden für randnahe Punkte virtuelle Punkte hinzugefügt, die außerhalb des Rechengebietes liegen, wie in Abbildung B.1 gezeigt. Die auf dem Randpunkt zwischen

$i + \frac{1}{2}$ und $i + \frac{3}{2}$, bzw. $j + \frac{1}{2}$ und $j + \frac{3}{2}$ vorgeschriebene Geschwindigkeit sei dabei $v_{x;N}$, bzw. $v_{y;N}$, so wird verlangt:

$$v_{x;N} = \frac{v_{x;i,j+\frac{1}{2}} + v_{x;i,j+\frac{3}{2}}}{2}, \quad (B.1)$$

$$v_{y;N} = \frac{v_{y;i+\frac{1}{2},j} + v_{y;i+\frac{3}{2},j}}{2}. \quad (B.2)$$

Für den Druck gelten die Neumann-Randbedingungen, also $\frac{\partial p}{\partial n} = 0$. Dies ist erfüllt, wenn

$$\frac{p_{i+\frac{1}{2},j+\frac{3}{2}} - p_{i+\frac{1}{2},j+\frac{1}{2}}}{h_y} = 0 \quad (B.3)$$

$$\Rightarrow p_{i+\frac{1}{2},j+\frac{3}{2}} = p_{i+\frac{1}{2},j+\frac{1}{2}} \quad (B.4)$$

Analog gilt die Rechnung für die seitlichen Ränder.

Durch die Staffelung des Gitters kann als Approximation der ersten Ableitung die Differenz benachbarter Punkte betrachtet werden:

$$\left(\frac{\partial v_x}{\partial x} \right)_{i+\frac{1}{2},j+\frac{1}{2}} \approx \frac{v_{x;i+1,j+\frac{1}{2}} - v_{x;i,j+\frac{1}{2}}}{h_x} \quad (B.5)$$

in x -Richtung, bzw. in y -Richtung:

$$\left(\frac{\partial v_y}{\partial y} \right)_{i+\frac{1}{2},j+\frac{1}{2}} \approx \frac{v_{y;i+\frac{1}{2},j} - v_{y;i+\frac{1}{2},j+1}}{h_y}. \quad (B.6)$$

Auf diese Weise sind die Ableitungen der Geschwindigkeiten für die Zellmitten bestimmt, also an den Punkten, an denen der Druck bestimmt wird.

Der Laplace-Operator an einem (inneren) Punkt $p_{i+\frac{1}{2},j+\frac{1}{2}}$ kann approximiert werden über:

$$\begin{aligned} \Delta p_{i+\frac{1}{2},j+\frac{1}{2}} &= \frac{\partial^2 p_{i+\frac{1}{2},j+\frac{1}{2}}}{\partial x^2} + \frac{\partial^2 p_{i+\frac{1}{2},j+\frac{1}{2}}}{\partial y^2} \\ &\approx \frac{p_{i-\frac{1}{2},j+\frac{1}{2}} - 2p_{i+\frac{1}{2},j+\frac{1}{2}} + p_{i+\frac{3}{2},j+\frac{1}{2}}}{h_x^2} \\ &\quad + \frac{p_{i+\frac{1}{2},j-\frac{1}{2}} - 2p_{i+\frac{1}{2},j+\frac{1}{2}} + p_{i+\frac{1}{2},j+\frac{3}{2}}}{h_y^2}. \end{aligned} \quad (B.7)$$

Dabei können beteiligte Punkte auch Randpunkte sein, die der Gleichung B.4 genügen. Diese Approximation gilt analog für den Laplace-Operator der Geschwindigkeiten $v_{x,y}$, sofern die Simulation nicht-hybrid ist und der Operator als zusätzliche Information benötigt wird.

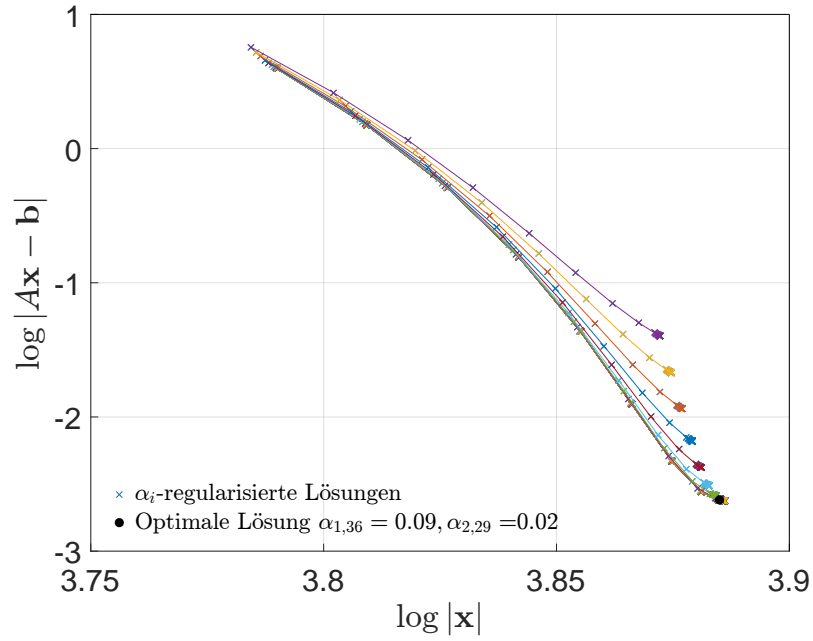
Für die Hybridsimulationen werden die Simulationsboxen für die Mikroskala auf den Gitterpunkten selbst, also (x_i, y_j) angesiedelt und wie in Kapitel 8.1 beschrieben, entsprechend des vorliegenden Flusses gedreht.

In diesem Kapitel sind die L-Kurven für verschiedene Polynomgrade P_{\max} und zwei unterschiedliche Basen aufgezeigt. Die beiden analysierten Polynombasen sind die in dieser Arbeit verwendeten Chebyshev-Polynome und Monome, die in Gleichung (8.9) bzw. Gleichung (8.8) vorgestellt sind.

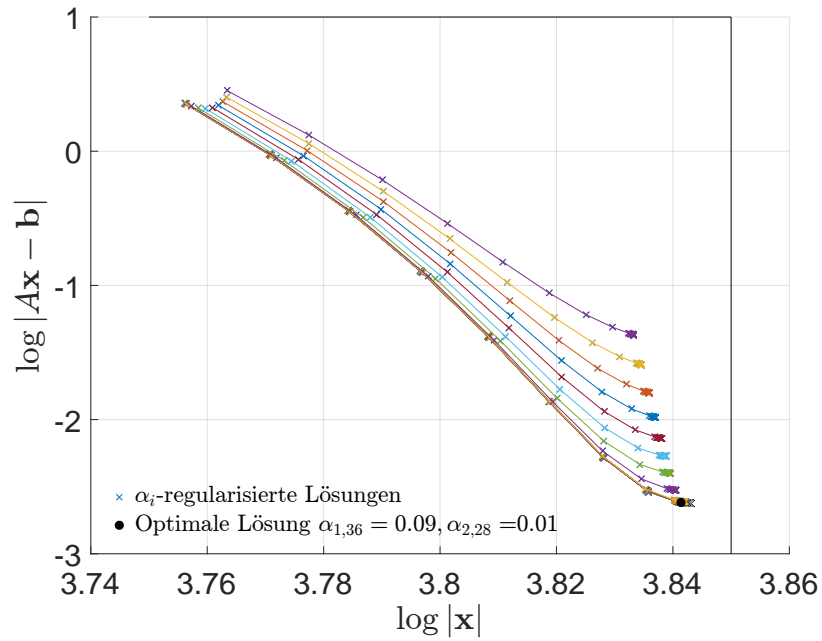
Hier werden lediglich die L-Kurven der Näherung $f(\lambda)$ für den gesamten Bereich $\lambda \in [-\frac{\gamma}{2}, \frac{\gamma}{2}]$ gezeigt. Für jeden vorgenommenen Fit, sowohl des gesamten Bereichs, als auch den beiden einzelnen Ästen $f(\lambda_1)$ mit $\lambda_1 < 0$ und $f(\lambda_2)$ mit $\lambda_2 > 0$, sowie $\sigma_{22}(\gamma)$ kann die L-Kurve betrachtet werden. Sie ist ein nützliches Werkzeug und gibt Aufschluss darüber, ob die einzelnen Parameter gut gewählt sind und die Lösung das Problem angemessen gut beschreiben.

In Abbildung C.1 ist der maximale Polynomgrad auf $P_{\max} = 4$ gesetzt. In a) ist die L-Kurve für Chebyshev-Polynome gezeigt, während b) die zugehörige L-Kurve der Monome aufzeigt. Hier ist ersichtlich, dass auch bei Monomen die L-Kurve je nach Wahl der Regularisierungsparameter die eher untypische Form annehmen kann, bei der der horizontale Teil des L kaum sichtbar ist.

Bei höherem erlaubten Polynomgrad $P_{\max} = 8$ steigt für die Chebyshev-Polynom-Basis der eigentlich senkrechte Teil des L stärker an, wie in Abbildung C.2 a) zu erkennen ist. Auch hier ist der horizontale Teil des L kaum zu sehen. Bei der L-Kurve der Regularisierung mit Monom-Basis in Abbildung C.2 b) ist die Form der L-Kurve gut nachzuvollziehen. Die optimalen Werte für α sind für dieses Problem $\alpha_1 = 0.09$ und $\alpha_2 = 0.005$. Die beiden Strafterme unterscheiden sich in ihrer Wichtung also um etwa eine Größenordnung.

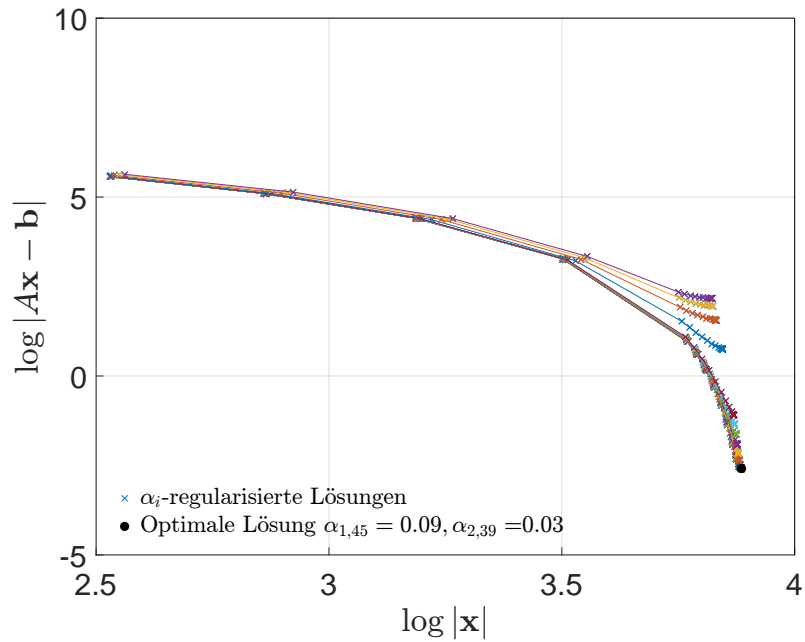


(a) Die L-Kurve des Tikhonov Fits mit Chebyshev-Polynomen.

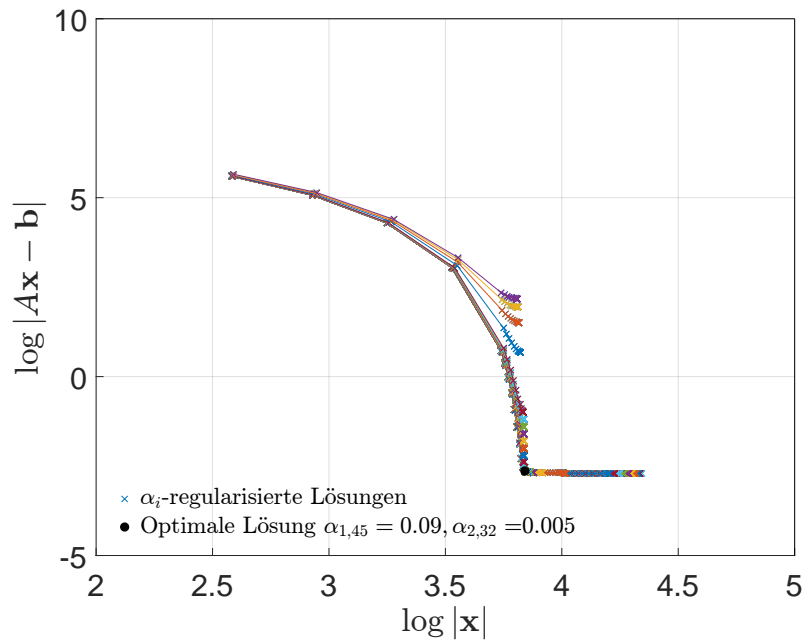


(b) Die L-Kurve des Tikhonov Fits mit Monomen.

Abb. C.1: L-Kurve des Tikhonov-Fits (a) Chebychev-Polynomen und (b) mit Monomen. Die Kurven sind im Hilbertraum \mathcal{H}_1 berechnet. Die L-Kurve weist für die verwendeten Parameter weder in (a) noch in (b) die typische Form auf, wie sie in Abbildung C.2 (b) für die Monomer-Basis deutlich zu erkennen ist. Der maximal erlaubte Polynomgrad ist auf $P_{\max} = 4$ gesetzt.



(a) Die L-Kurve des Tikhonov Fits mit Chebyshev-Polynomen.



(b) Die L-Kurve des Tikhonov Fits mit Monomen.

Abb. C.2: L-Kurve des Tikhonov-Fits (a) Chebyshev-Polynomen und (b) mit Monomen. Die Kurven sind im Hilbertraum \mathcal{H}_1 berechnet. Die L-Kurve der Chebyshev-Polynome weist nicht die typische Form auf, wie sie in (b) für Monome deutlich zu erkennen ist. Der maximal erlaubte Polynomgrad ist auf $P_{\max} = 8$ gesetzt.

- [1] Emamy, N.; Lukáčová-Medvid'ová, M.; Stalter, S.; Virnau, P.; Yelash, L. Reduced-order hybrid multiscale method combining the Molecular Dynamics and the Discontinuous-Galerkin method. *VII International Conference on Computational Methods for Coupled Problems in Science and Engineering Coupled Problems 2017* (Eds. Papadrakakis, Onate and Schrefle) **2017**, pp. 1–15.
- [2] Stalter, S.; Yelash, L.; Emamy, N.; Statt, A.; Hanke, M.; Lukáčová-Medvid'ová, M.; Virnau, P. Molecular dynamics simulations in hybrid particle-continuum schemes: Pitfalls and caveats. *Computer Physics Communications* **2018**, 224, 198–208.
- [3] Wolfram Research, Inc.. Mathematica Version 11.3, 2018. Champaign, IL, 2018.
- [4] Humphrey, W.; Dalke, A.; Schulten, K. VMD – Visual Molecular Dynamics. *Journal of Molecular Graphics* **1996**, 14, 33–38.
- [5] Stone, J. An Efficient Library for Parallel Ray Tracing and Animation. Masterarbeit, Computer Science Department, University of Missouri-Rolla, 1998.
- [6] Reedtz-Runge, S. Crystal structure of Glucagon-Like Peptide-1 in complex with the extracellular domain of the Glucagon-Like Peptide-1 Receptor, 2009.
- [7] Frenkel, D.; Smit, B. *Understanding Molecular Simulations*; Academic Press, 1996.
- [8] Allen, M.P.; Tildesley, D.J. *Computer Simulation of Liquids* (Oxford Science Publications); Clarendon Press, 1989.
- [9] Wendt, J.F. *Computational Fluid Dynamics*; Springer Berlin Heidelberg, 2009.
- [10] E, W.; Engquist, B.; Li, X.; Ren, W.; Vanden-Eijnden, E. Heterogeneous multiscale methods: A review. *Comm. Comp. Phys.* **2007**, 2, 367–450.
- [11] E, W.; ; Enquist, B. The heterogeneous mutli-scale method. *Commun. Math. Sci.* **2003**, 1, 87–133.
- [12] E, W.; Li, X. Analysis of the heterogeneous multiscale method for gas dynamics. *Methods Appl. Anal.* **2004**, 11, 557–572.
- [13] E, W.; Ming, P.; Zhang, P. Analysis of the heterogeneous multiscale method for elliptic homogenization problems. *J. American Math. Soc.* **2005**, 18, 121–157.
- [14] Ren, W. Analytical and numerical study of coupled atomistic-continuum methods for fluids. *J. Comput. Phys.* **2007**, 227, 1353–1371.
- [15] Ren, W.; E, W. Heterogeneous multiscale method for the modeling of complex fluids and micro-fluidics. *Journal of Computational Physics* **2005**, 204, 1–26.
- [16] Yasuda, S.; Yamamoto, R. Multiscale modeling and simulation for polymer melt flows between parallel plates. *Phys. Rev. E* **2010**, 81.

- [17] Kremer, K.; Grest, G.S. Dynamics of entangled linear polymer melts: A molecular-dynamics simulation. *The Journal of Chemical Physics* **1990**, 92, 5057–5086.
- [18] Metropolis, N.; Ulam, S. The Monte Carlo Method. *Journal of the American Statistical Association* **1949**, 44, 335–341.
- [19] Laurendeau, N.M. *Statistical Thermodynamics: Fundamentals and Applications*; CAMBRIDGE UNIV PR, 2005.
- [20] Becker, R. *Theory of Heat*; Springer, 1967.
- [21] Altenbach, H. *Kontinuumsmechanik: Einführung in die materialunabhängigen und materialabhängigen Gleichungen (German Edition)*; Springer, 2012.
- [22] Lees, A.; Edwards, S. The computer study of transport processes under extreme conditions. *J. Phys. C.* **1972**, 5, 1921–1929.
- [23] Hoover, W.G.; Evans, D.J.; Hickman, R.B.; Ladd, A.J.C.; Ashurst, W.T.; Moran, B. Lennard-Jones triple-point bulk and shear viscosities. Green-Kubo theory, Hamiltonian mechanics, and nonequilibrium molecular dynamics. *Physical Review A* **1980**, 22, 1690–1697.
- [24] Evans, D.J.; Morriss, G.P. Nonlinear-response theory for steady planar Couette flow. *Phys. Rev. A* **1984**, 30, 1528–1530.
- [25] Leimkuhler, B.; Matthews, C. Robust and efficient configurational molecular sampling via Langevin dynamics. *Journal of Chemical Physics* **2013**, 138, 174102–.
- [26] Schwabl, F. *Statistische Mechanik*; Springer-Verlag GmbH, 2006.
- [27] Clausius, R. Über einen auf die Wärme anwendbaren mechanischen Satz. *Annalen der Physik* **1870**, 217, 124–130.
- [28] Goldstein, H. *Klassische Mechanik*; 1983.
- [29] Nosé, S. Constant-temperature molecular dynamics. *Journal of Physics: Condensed Matter* **1990**, 2, SA115.
- [30] Kubo, R. The fluctuation-dissipation theorem. *Reports on Progress in Physics* **1966**, 29, 255.
- [31] Lowe, C.P. An alternative approach to dissipative particle dynamics. *Europhys. Lett.* **1999**, 47, 145–151.
- [32] Koopman, E.A.; Lowe, C.P. Advantages of a Lowe-Andersen thermostat in molecular dynamics simulations. *The Journal of Chemical Physics* **2006**, 124, 204103.
- [33] Kemnitz, E.; Fishedick, A.; Grubert, L.; Hartmann, A.; Hennig, H. *Basiswissen Schule - Chemie Abitur*; Bibliograph. Instit. GmbH, 2015.
- [34] Harvey, S.C.; Tan, R.K.Z.; Cheatham, T.E. The flying ice cube: Velocity rescaling in molecular dynamics leads to violation of energy equipartition. *Journal of Computational Chemistry* **1998**, 19, 726–740.
- [35] Trefz, B. Computer simulations of the statistical behaviour of active particles. PhD thesis, 2016.

- [36] Winkler, A. Computer Simulations of Colloidal Fluids in Confinement. Dissertation, 2012.
- [37] Brünger, A.; Brooks, C.L.; Karplus, M. Stochastic boundary conditions for molecular dynamics simulations of ST2 water. *Chemical Physics Letters* **1984**, *105*, 495–500.
- [38] Schlick, T. *Molecular Modeling and Simulation: An Interdisciplinary Guide*; Springer New York, 2010.
- [39] Weeks, J.D.; Chandler, D.; Andersen, H.C. Role of Repulsive Forces in Determining the Equilibrium Structure of Simple Liquids. *The Journal of Chemical Physics* **1971**, *54*, 5237–5247.
- [40] Carnahan, N.F.; Starling, K.E. Equation of State for Nonattracting Rigid Spheres. *The Journal of Chemical Physics* **1969**, *51*, 635–636.
- [41] Barker, J.A.; Henderson, D. Perturbation Theory and Equation of State for Fluids. II. A Successful Theory of Liquids. *Journal of Chemical Physics* **1967**, *47*, 4714–4720.
- [42] Tang, Y. Role of the Barker–Henderson diameter in thermodynamics. *The Journal of Chemical Physics* **2002**, *116*, 6694–6700.
- [43] Schöpe, H.J. Physikalische Eigenschaften kolloidaler Festkörper. Dissertation, Johannes Gutenberg-Universität Mainz, 2000.
- [44] Asakura, S.; Oosawa, F. Interaction between particles suspended in solutions of macromolecules. *Journal of Polymer Science* **1958**, *33*, 183–192.
- [45] Zausch, J.; Virnau, P.; Binder, K.; Horbach, J.; Vink, R.L. Statics and dynamics of colloid-polymer mixtures near their critical point of phase separation: A computer simulation study of a continuous Asakura–Oosawa model. *The Journal of Chemical Physics* **2009**, *130*, 064906.
- [46] Grest, G.S.; Kremer, K. Molecular dynamics simulation for polymers in the presence of a heat bath. *Physical Review A* **1986**, *33*, 3628–3631.
- [47] Demtröder, W. *Experimentalphysik 1: Mechanik und Wärme (Springer-Lehrbuch) (German Edition)*; Springer Spektrum, 2008.
- [48] Kummer, F. The BoSSS Discontinuous Galerkin solver for incompressible fluid dynamics and an extension to singular equations. Dissertation, 2011.
- [49] Müller, B.; Kummer, F.; Oberlack, M.; Wang, Y. Simple multidimensional integration of discontinuous functions with application to level set methods. *International Journal for Numerical Methods in Engineering* **2012**, *92*, 637–651.
- [50] Griebel, M.; Dornsheifer, T.; Neunhoffer, T. *Numerical Simulation in Fluid Dynamics: A Practical Introduction (Monographs on Mathematical Modeling and Computation)*; SIAM: Society for Industrial and Applied Mathematics, 1997.
- [51] Chorin, A.J. The numerical solution of the Navier-Stokes equations for an incompressible fluid. *Bull. Amer. Math. Soc.* **1967**, *73*, 928–931.

- [52] Bian, X.; Deng, M.; Tang, Y.H.; Karniadakis, G. Analysis of hydrodynamic fluctuations in heterogeneous adjacent multidomains in shear flow. *Phys. Rev. E* **2016**, *93*, 033312.
- [53] Emamy, N.; Kummer, F.; Mrosek, M.; Karcher, M.; Oberlack, M. Implicit-explicit and explicit projection schemes for the unsteady incompressible Navier-Stokes equations using a high-order dG method. *Comput. Fluids* **2017**, *154*, 285 – 295.
- [54] Butt, R. *Introduction to Numerical Analysis Using MATLAB*; JONES & BARTLETT PUBL INC, 2009.
- [55] Reis, G.A.; Tasso, I.V.M.; Souza, L.F.; Cuminato, J.A. A compact finite differences exact projection method for the Navier-Stokes equations on a staggered grid with fourth-order spatial precision. *Comput. Fluids* **2015**, *118*, 19–31.
- [56] Sanderse, B.; Koren, B. Accuracy analysis of explicit Runge-Kutta methods applied to the incompressible Navier-Stokes equations. *J. Comput. Phys.* **2012**, *231*, 3041–3063.
- [57] Emamy, N. Numerical simulation of deformation of a droplet in a stationary electric field using DG. Dissertation, TU Darmstadt, Darmstadt, 2014.
- [58] Ferrer, E.; Moxey, D.; Willden, R.H.J.; Sherwin, S.J. Stability of projection methods for incompressible flows using high order pressure-velocity pairs of same degree: continuous and discontinuous Galerkin formulations. *Commun. Comput. Phys.* **2014**, *16*, 817–840.
- [59] Steinmoeller, D.T.; Stastna, M.; Lamb, K.G. A short note on the discontinuous Galerkin discretization of the pressure projection operator in incompressible flow. *J. Comput. Phys.* **2013**, *251*, 480–486.
- [60] Pietro, D.A.D.; Ern, A. *Mathematical Aspects of Discontinuous Galerkin Methods (Mathématiques et Applications)*; Springer, 2011.
- [61] Arnold, D.N.; Brezzi, F.; Cockburn, B.; Marini, L.D. Unified Analysis of Discontinuous Galerkin Methods for Elliptic Problems. *SIAM Journal on Numerical Analysis* **2002**, *39*, 1749–1779.
- [62] Shahbazi, K. An explicit expression for the penalty parameter of the interior penalty method. *J. Comput. Phys.* **2005**, *205*, 401–407.
- [63] Hillewaert, K. Development of the discontinuous Galerkin method for high-resolution, large scale CFD and acoustics in industrial geometries. Dissertation, 2013.
- [64] Landau, D.P.; Binder, K. *A Guide to Monte Carlo Simulations in Statistical Physics*; Cambridge University Press, 2014.
- [65] Müller-Krumbhaar, H.; Binder, K. Dynamic properties of the Monte Carlo method in statistical mechanics. *Journal of Statistical Physics* **1973**, *8*, 1–24.
- [66] Schmidt, G.; Nakatani, A.I.; Butler, P.D.; Karim, A.; Han, C.C. Shear Orientation of Viscoelastic Polymer-Clay Solutions Probed by Flow Birefringence and SANS. *Macromolecules* **2000**, *33*, 7219–7222.

- [67] Zhou, W.J.; Kornfield, J.A.; Ugaz, V.M.; Burghardt, W.R.; Link, D.R.; Clark, N.A. Dynamics and Shear Orientation Behavior of a Main-Chain Thermotropic Liquid Crystalline Polymer. *Macromolecules* **1999**, *32*, 5581–5593.
- [68] Yamamoto, R.; Onuki, A. Dynamics and rheology of a supercooled polymer melt in shear flow. *The Journal of Chemical Physics* **2002**, *117*, 2359–2367.
- [69] Giusteri, G.G.; Seto, R. A theoretical framework for steady-state rheometry in generic flow conditions. *Journal of Rheology* **2018**, *62*, 713–723.
- [70] Seto, R.; Giusteri, G.G.; Martiniello, A. Microstructure and thickening of dense suspensions under extensional and shear flows. [\[1706.01745v2\]](#).
- [71] Barnes, H.A. Shear-Thickening (“Dilatancy”) in Suspensions of Nonaggregating Solid Particles Dispersed in Newtonian Liquids. *Journal of Rheology* **1989**, *33*, 329–366.
- [72] The MathWorks, Inc., Natick, Massachusetts, United States.. MATLAB and Statistics Toolbox Release 2016a, 2016.
- [73] Anderson, E.; Bai, Z.; Bischof, C.; Blackford, S.; Demmel, J.; Dongarra, J.; Du Croz, J.; Greenbaum, A.; Hammarling, S.; McKenney, A.; Sorensen, D. *LAPACK Users’ Guide*, third ed.; Society for Industrial and Applied Mathematics: Philadelphia, PA, 1999.
- [74] Temlyakov, V. Greedy approximation. *Acta Numer.* **2008**, *17*, 235–409.
- [75] Todd, B.; Davis, P. Nonequilibrium Molecular Dynamics Simulations of Planar Elongational Flow with Spatially and Temporally Periodic Boundary Conditions. *Phys. Rev. Lett.* **1998**, *81*, 1118–1121.
- [76] Hansen, P.C. *The L-Curve and Its Use in the Numerical Treatment of Inverse Problems*; Vol. 4, 2001; pp. 119–142.
- [77] Zausch, J. Dynamics, Rheology and Critical Properties of Colloidal Fluid Mixtures: Molecular Dynamics Studies in Equilibrium and Under Shear. PhD thesis, 2008.
- [78] Reinmüller, A.; Schöpe, H.J.; Palberg, T. Self-Organized Cooperative Swimming at Low Reynolds Numbers. *Langmuir* **2013**, *29*, 1738–1742.
- [79] Trefz, B.; Siebert, J.T.; Speck, T.; Binder, K.; Virnau, P. Estimation of the critical behavior in an active colloidal system with Vicsek-like interactions. *The Journal of Chemical Physics* **2017**, *146*, 074901.
- [80] Halverson, J.D.; Brandes, T.; Lenz, O.; Arnold, A.; Bevc, S.; Starchenko, V.; Kremer, K.; Stuehn, T.; Reith, D. ESPResSo++: A modern multiscale simulation package for soft matter systems. *Computer Physics Communications* **2013**, *184*, 1129–1149.
- [81] Hartkamp, R.; Todd, B.D.; Luding, S. A constitutive framework for the non-Newtonian pressure tensor of a simple fluid under planar flows. *The Journal of Chemical Physics* **2013**, *138*, 244508.
- [82] Schelle, S. Hybrid-Simulation der Molekulardynamik und Computational Fluid-dynamik. Masterarbeit, 2016.

- [83] Vogel, H. Stochastische Galerkin-Methode bei Fluid Dynamik Modellen. Masterarbeit, 2018.
- [84] Harlow, F.H.; Welch, J.E. Numerical Calculation of Time-Dependent Viscous Incompressible Flow of Fluid with Free Surface. *Physics of Fluids* **1965**, *8*, 2182.