DIPLOMA THESIS

“Multi-Particle Collision Dynamics” Simulation of Viscoelastic Fluids

A thesis by

David Toneian,

supervised by

Prof. Gerhard Kahl, Institute of Theoretical Physics, TU Wien,

Prof. Gerhard Gompper, Institute of Complex Systems 2 and Institute for Advanced Simulation 2, Forschungszentrum Jülich, and

Prof. Roland G. Winkler, Institute of Complex Systems 2 and Institute for Advanced Simulation 2, Forschungszentrum Jülich

submitted to the

TU Wien

in partial fulfillment of the requirements for the academic degree of “Diplom-Ingenieur” (equivalent to Master of Science).

2015-10-28

Abstract

In this thesis, the simulation technique called *Multi-Particle Collision Dynamics* (MPC) is extended such that it allows for the modeling of viscoelastic fluids. This is achieved by representing the fluid not by independent MPC particles, as would be the case in the original formulation of MPC, but rather by polymer-like aggregates of MPC particles. This way, the MPC particles do not propagate freely anymore, but instead are now subjected to an intra-polymer interaction potential.

Said potential introduces elastic degrees of freedom, and is chosen to be quadratic in the separation of nearest neighbors, which results in polymers that are described by the *Rouse model*. This choice allows one to find, starting from a generalized Navier-Stokes equation in the limit of low Reynolds numbers, an analytic solution for components of the velocity autocorrelation in Fourier space, as will be shown in this work.

Furthermore, a new implementation of the extended MPC algorithm, capable of highly parallel execution on a Graphics Processing Unit (GPU), is presented. Simulation data obtained from this implementation are compared to the theoretical results, and a remarkable agreement is demonstrated on both the qualitative and quantitative level for polymers containing up to 10 MPC particles. The velocity autocorrelation in Fourier space is examined for asymptotically large correlation times, and it is found that the fluid's behavior corresponds to that of a Newtonian fluid on sufficiently long time scales.

Finally, for the case of trimers, the relationship between the interaction strength (spring constant) and the root-mean-square bond length of the polymer is derived analytically for a shear flow with arbitrary shear rate.

Contents

1 Introduction and Motivation 9

1.1 Multiscale Systems 9

1.2 Multi-Particle Collision Dynamics 10

1.3 Viscoelastic Fluids 11

1.4 Scope and Organization of this Thesis 12

1.5 Acknowledgements 13

2 Fluid Dynamics 15

2.1 Fundamental Principles 15

2.1.1 The Continuity Equation 15

2.1.2 The Substantial Derivative 16

2.1.3 Euler's Equation for Ideal Fluids 17

2.1.4 Momentum Flux 18

2.1.5 The Navier-Stokes Equation 19

2.1.6 Incompressibility 22

2.2 Adaptions of the Navier-Stokes Equation 23

2.2.1 The Linearized Navier-Stokes Equation 23

2.2.2 Linear Viscoelasticity 24

3 Multi-Particle Collision Dynamics 27

3.1 Introduction: MPC for Simple Fluids 27

3.1.1 Stochastic Rotation Dynamics (SRD) 28

3.1.2 Grid Shift 30

3.1.3 Boundary Conditions 30

3.1.3.1 Periodic Boundary Conditions 31

3.1.3.2 Lees-Edwards Boundary Conditions 32

3.1.4 Maxwell-Boltzmann-Scaling Thermostat . . . 33
3.2 MPC for Complex Systems . . . 34

4 MPCDSim: A Novel GPU-Based MPC Implementation 37
4.1 Overview and Design Goals . . . 37
4.2 Simulation Parameters . . . 39
4.2.1 System Properties . . . 39
4.2.2 System Initialization . . . 40
4.3 Verification of Implementation Correctness . . . 41
4.3.1 Shear Flow of a Simple Fluid . . . 41
4.3.2 Grid Shift . . . 41

5 Center-of-Mass Velocity Autocorrelation Function 45
5.1 Theory . . . 45
5.1.1 Laplace-Domain Equation of Motion . . . 45
5.1.2 Solution for $\tilde{C}_{\mathrm{v}}^{\mathrm{T}}$ for Simple Fluids . . . 47
5.1.3 Solution for $\tilde{C}_{\mathrm{v}}^{\mathrm{T}}$ for Rouse Polymers . . . 48
5.1.3.1 Storage and Loss Moduli . . . 48
5.1.3.2 Relaxation Modulus . . . 49
5.1.3.3 Solution for $\tilde{C}_{\mathrm{v}}^{\mathrm{T}}(\mathbf{k}, t)$ for the Rouse Model . . . 51
5.1.3.4 Limit of Long Polymers . . . 53
5.1.3.5 Dimers . . . 54
5.2 Simulation Results . . . 56
5.2.1 Simple Fluid . . . 56
5.2.2 Dimers . . . 57
5.2.3 Decamers . . . 57
5.3 Numerical Results for $\tilde{C}_{\mathrm{v}}^{\mathrm{T}}$ . . . 57

6 Trimer MPC Fluid 65

7 Conclusion and Outlook 75

A Notation and Mathematical Conventions 77

B Properties of the Fourier Transformation 81
B.1 Fourier Transformation of Derivatives . . . 81

B.2 Fourier Transform of 1 and the Dirac Delta Function . . . 82

C Properties of the Laplace Transformation 83

C.1 Laplace Transformation of Derivatives . . . 83

C.2 Convolution Theorem . . . 84

D Tensors 85

D.1 The Kronecker Tensor δ_{ij} . . . 85

D.2 The Levi-Civita Tensor ε_{ijk} . . . 85

D.3 Symmetric and Anti-Symmetric Parts of a Tensor . . . 86

E Uniform Sampling from the 2-Sphere 87

Bibliography 89

Chapter 1

Introduction and Motivation

1.1 Multiscale Systems

In the field of soft matter physics, one is often interested in systems which contain entities of vastly different sizes and masses. Consequently, the involved time and length scales can span several orders of magnitude;[1,2] this poses a significant challenge for both theoretical treatment and computer simulations of such systems.

Some examples are *dispersions* or *suspensions*, where large physical units, such as colloids, are surrounded by much smaller and lighter molecules, such as water.[3] The aforementioned *colloids* are solid objects of typically spherical or rod-like shape,[4] and have sizes in the range of $1\,\mathrm{nm}$ to $10\,\mu\mathrm{m}$;[5] compared to a water molecule's diameter of about $0.3\,\mathrm{nm}$,[6] this is several orders of magnitude larger. Another frequently considered class of relatively big objects is that of polymers. A *polymers* is a large, often linear, molecule which is built by repeatedly linking a number of chemical units, called *monomers*, together.[7] The number of different types of units

[1] J. K. G. Dhont, G. Gompper, P. R. Lang, D. Richter, M. Ripoll, D. Willbold, and R. Zorn, eds. *Macromolecular Systems in Soft and Living Matter.* Vol. 20. Key Technologies. Forschungszentrum Jülich, 2011, Section 1.1.

[2] G. Gompper, T. Ihle, D. M. Kroll, and R. G. Winkler. *Multi-Particle Collision Dynamics: A Particle-Based Mesoscale Simulation Approach to the Hydrodynamics of Complex Fluids.* Advances in Polymer Science. Springer, 2008. Chapter 1.

[3] J. K. G. Dhont, G. Gompper, and D. Richter, eds. *Soft Matter: Complex Materials on Mesoscopic Scales.* Vol. 10. Matter and Materials. Forschungszentrum Jülich, 2002, Section 2.1.

[4] Dhont, Gompper, Lang, Richter, Ripoll, Willbold, and Zorn, *Macromolecular Systems in Soft and Living Matter*, Section 1.1.

[5] Dhont, Gompper, and Richter, *Soft Matter: Complex Materials on Mesoscopic Scales*, Section 2.1.

[6] Y. Huang, X. Zhang, Z. Ma, W. Li, Y. Zhou, J. Zhou, W. Zheng, and C. Q. Sun. "Size, separation, structural order, and mass density of molecules packing in water and ice". *Scientific Reports* 3 (2013), 3005.

[7] Dhont, Gompper, and Richter, *Soft Matter: Complex Materials on Mesoscopic Scales*, Section 2.3.

present in a polymer can range from one, as is the case in many synthetic polymers, to about twenty for proteins.[8] The molecular weights of such polymers can exceed $10^3\,\frac{\mathrm{kg}}{\mathrm{mol}}$.[9]

While the disparate length and time scales already present difficulties in their own right, other scenarios in soft matter physics can complicate the problem even further: microorganisms, for example, not only are comparatively very large and heavy, but may also have the capability of actively propelling themselves.[10] Description of these systems, called *active* since parts of the system exert forces on their surroundings autonomously, demands very powerful tools for the theoretical treatment and for computer simulations.

Some of the phenomena displayed by these systems may allow for a satisfactory description that does not take the solvent into account explicitly. However, some behaviors observed experimentally depend crucially on the hydrodynamic interactions the solvent mediates; for example, simple swimming organisms rely on these hydrodynamic interactions to synchronize the beating of their flagella, which is necessary to achieve efficient and directed movement.[11,12] It may therefore become unavoidable to treat both large and small constituents, if one desires to reproduce these behaviors in computer simulations.

1.2 Multi-Particle Collision Dynamics

Although there are a multitude of simulation schemes in existence, there is no single one among them that is viable in all situations. For example, if one cannot neglect the interplay of large, heavy, *macroscopic* constituents on the one hand and small, light, *microscopic* solvent particles on the other, simulation techniques where the energies and forces are calculated precisely in the way governed by physical principles (e.g. quantum or classical mechanics) may be very accurate, but quickly become so computationally expensive that they turn unfeasible for all but the tiniest systems. This is because the small and light-weight parts of the simulated objects typically exhibit dynamics governed by time scales much shorter than those of the larger constituents. In such cases, one might instead choose to employ a *mesoscopic* simulation method: here, one tries to design the simulation algorithm such that it captures the relevant

[8] R. B. Bird, R. C. Armstrong, and O. Hassager. *Fluid Mechanics.* 2nd ed. Vol. 1. Dynamics of Polymeric Liquids. John Wiley & Sons, 1987, § 2.1.

[9] Bird, Armstrong, and Hassager, *Fluid Mechanics*, § 2.1.

[10] E. Lauga and T. R. Powers. "The hydrodynamics of swimming microorganisms". *Reports on Progress in Physics* 72 (2009), 096601.

[11] M. Theers and R. G. Winkler. "Effects of thermal fluctuations and fluid compressibility on hydrodynamic synchronization of microrotors at finite oscillatory Reynolds number: a multiparticle collision dynamics simulation study". *Soft Matter* 10 (2014), 5894.

[12] E. Lauga and R. E. Goldstein. "Dance of the microswimmers". *Physics Today* 65 (2012), 30.

aspects of the dynamics at the different time and length scales involved, but discards the other aspects to gain computational efficiency.

Multi-particle collision dynamics, abbreviated *MPC* or *MPCD*, is such a mesoscopic technique. Introduced in 1999 by Malevanets and Kapral,[13,14] it is now commonly used in the simulation of fluid systems. MPC deals with discrete particles (as opposed to continua), and consists of two alternating phases:

In the *streaming phase*, MPC particles, which model either a mesoscopic volume of solvent or (part of) a macroscopic solute, are propagated ballistically, i.e. on trajectories influenced only by external forces such as as gravity or a magnetic field, ignoring all interactions with other MPC particles in the system. Exceptions are made for the case where specific particles model just part of a macroscopic unit, e.g. part of a membrane: there, one can define arbitrary interaction potentials with the other, connected parts, and perform a sufficient number of cheap *molecular dynamics* (*MD*) simulation steps with a small number of interaction partners.

In the following *collision phase*, the simulation volume is virtually partitioned into cubic cells. Each particle then interacts only with other particles occupying the same cell. In the original and most widely used variant of MPC, known as *stochastic rotation dynamics* (*SRD* for short), this interaction is achieved by choosing the corresponding cell's center-of-mass frame as the reference frame, and then rotating all particle velocities in this frame around a randomly chosen axis passing through the origin. This choice is made independently for each collision cell and simulation time-step.

This algorithm conserves the particle number, mass, and linear momentum. Together with a sufficient degree of isotropy, these properties lead to the reproduction of hydrodynamic behavior for a wide range of simulation parameters. The computational simplicity of the streaming and collision steps allows for highly parallel and efficient implementations, so that systems with $\sim 10^9$ MPC particles can be simulated.[15]

1.3 Viscoelastic Fluids

In what was described above, the MPC particles that represent the fluid (rather than the solutes), which usually is the vast majority of MPC particles, stream independently of each

[13] A. Malevanets and R. Kapral. "Mesoscopic model for solvent dynamics". *The Journal of Chemical Physics* 110 (1999), 8605.

[14] A. Malevanets and R. Kapral. "Solute molecular dynamics in a mesoscale solvent". *The Journal of Chemical Physics* 112 (2000), 7260.

[15] C.-C. Huang, G. Gompper, and R. G. Winkler. "Hydrodynamic correlations in multiparticle collision dynamics fluids". *Physical Review E* 86 (2012), 056711. Section I.

other. While this works very well for a range of systems, this model does not give rise to elastic properties of the solvent; the simulated fluid is purely viscous, i.e. Newtonian.

The elastic properties of viscoelastic fluids are, however, highly important for applications both in industry and biology.[16] In order to be able to incorporate elastic behavior into simulations, the MPC algorithm has been extended to couple fixed pairs of MPC fluid particles via harmonic springs to form dimers, and stream each dimer independently from the others, as opposed to streaming individual MPC particles without regard for each other.[17] While this approach has been shown to give rise to non-Newtonian phenomena,[18] it has not yet been studied extensively. A thorough understanding is, however, necessary to learn as much as possible from the simulation results.

1.4 Scope and Organization of this Thesis

In this thesis, a newly developed implementation of MPC, capable of execution on graphics cards (GPUs), is presented (chapter 4).

To verify its correctness, simulation results for the basic MPC algorithm are compared with known reference data and with theory, and it is found that there are no significant deviations.

The MPC simulation scheme is then extended to MPC polymers consisting of an arbitrary number of linked monomers. In chapter 5, a closed expression for the velocity autocorrelation function in Fourier space is found for linear polymers with harmonic interaction potentials between adjacent monomers. Said expression is discussed, and shown to describe the simulation results obtained to great accuracy.

Finally, for the case where the polymer consists of three monomers (i.e. is a trimer), analytical calculations are presented in chapter 6 that link the spring constants to the temperature, shear rate, viscosity, and hydrodynamic radius, under the condition that the mean squared end-to-end distance of the MPC polymer is fixed.

Additionally, to provide the necessary background, an introduction to the relevant aspects of fluid dynamics is given in chapter 2, along with a detailed description of the simulation method employed (chapter 3). Appendix A specifies the notation and mathematical conventions used in

[16] B. Kowalik and R. G. Winkler. “Multiparticle collision dynamics simulations of viscoelastic fluids: Shear-thinning Gaussian dumbbells”. *Journal of Chemical Physics* 138 (2013), 104903. Section I.

[17] Kowalik and Winkler, “Multiparticle collision dynamics simulations of viscoelastic fluids: Shear-thinning Gaussian dumbbells”, Section I.

[18] Kowalik and Winkler, “Multiparticle collision dynamics simulations of viscoelastic fluids: Shear-thinning Gaussian dumbbells”, Section I.

this thesis, and subsequent appendices provide further mathematical background information that is referred to in the text as needed.

Conclusions and a brief outlook are presented in the closing chapter 7.

1.5 Acknowledgements

This work would not have been possible without the extremely proficient and friendly support by my advisors, Prof. Gerhard Kahl, Prof. Roland G. Winkler, and Prof. Gerhard Gompper, to whom I would like to express my sincerest gratitude. I would also like to thank the Forschungszentrum Jülich in general and the Institute of Complex Systems 2 in particular, as well as the TU Wien, for generous financial support, and for providing invaluable study and research opportunities such as the IHRS BioSoft Guest Student Program and the IFF Spring School. Also, I am thankful for the computing resources provided by the Institute of Complex Systems 2 and Institute for Advanced Simulation 2 at the Forschungszentrum Jülich, as well as the Jülich Supercomputing Centre.

Also, I would like to thank very much (in alphabetical order) Thorsten Auth, Thomas Eisenstecken, Maud Formanek, Raphael Hornung, Angela Koffler, Dimitrios Kyranas, Marisol Ripoll, Sebastian Rode, Mario Theers, Anoop Varghese, Elmar Westphal, and Jens Winkelmann for helpful discussions.

Chapter 2

Fluid Dynamics

2.1 Fundamental Principles

This section, which heavily borrows from Landau and Lifshitz,[19] is intended to give a quick review of the foundations of fluid dynamics most relevant to this thesis. Subsequent chapters will extend the material presented here.

2.1.1 The Continuity Equation[20]

Let $\varrho\left(\mathbf{r},t\right)$ be the density of the fluid at the position $\mathbf{r}$ and time t, and let there be an arbitrary, but fixed, non-degenerate 3-dimensional volume V, the 2-dimensional boundary of which is denoted by ∂V. The infinitesimal boundary surface element is referred to by the vector $\mathrm{d}\mathbf{f}\left(\mathbf{r}\right)$, the magnitude of which is equal to the surface element's area, while the vector's direction is perpendicular to the surface, pointing outwards of the volume V. Finally, let $\mathbf{v}\left(\mathbf{r},t\right)$ be the fluid's velocity field.

The fluid mass m_V inside the volume V is given by

$$m_V = \int\limits_V \varrho\left(\mathbf{r},t\right)\,\mathrm{d}^3r\,. \tag{2.1.1}$$

The amount of mass per time flowing out of V through a particular surface element $\mathrm{d}\mathbf{f}$ is

[19] L. D. Landau and E. M. Lifshitz. *Fluid Mechanics.* 2nd ed. Vol. 6. Course of Theoretical Physics. Pergamon Press, 1987.

[20] Landau and Lifshitz, *Fluid Mechanics*, § 1.

given by the component of the mass flux density $\varrho\mathbf{v}$ along $\mathrm{d}\mathbf{f}$,

$$\varrho\left(\mathbf{r},t\right)v_i\left(\mathbf{r},t\right)\,\mathrm{d}f_i\,, \tag{2.1.2}$$

so that the total amount of mass per time leaving V is

$$\oint\limits_{\partial V}\varrho\left(\mathbf{r},t\right)v_i\left(\mathbf{r},t\right)\,\mathrm{d}f_i\,. \tag{2.1.3}$$

The postulate of conservation of mass is formulated by equating this quantity to the negative time derivative of m_V, i.e. the mass flowing out of V per unit time. This leads to the integral form of the continuity equation, which states that mass can neither be created nor destroyed:

$$-\int\limits_V\partial_t\varrho\,\mathrm{d}^3r=\oint\limits_{\partial V}\varrho v_i\,\mathrm{d}f_i\,. \tag{2.1.4}$$

One can use Gauss's theorem to rewrite the right-hand side of the equation above as

$$\oint\limits_{\partial V}\varrho v_i\,\mathrm{d}f_i=\int\limits_V\partial_i\left(\varrho v_i\right)\,\mathrm{d}^3r \tag{2.1.5}$$

to obtain

$$\int\limits_V\left(\partial_i\left(\varrho v_i\right)+\partial_t\varrho\right)\,\mathrm{d}^3r=0\,. \tag{2.1.6}$$

Since this equation holds for arbitrary volumes V, the integrand has to be 0; thus, one arrives at the differential form of the continuity equation:

$$\partial_i\left(\varrho v_i\right)+\partial_t\varrho=0\,. \tag{2.1.7}$$

2.1.2 The Substantial Derivative[21]

When talking about time derivatives of vector fields that depend on the position and on time, such as the velocity field $\mathbf{v}\left(\mathbf{r},t\right)$, one sometimes refers to the *partial derivative*,

$$\partial_t\mathbf{v}\left(\mathbf{r},t\right)\ , \tag{2.1.8}$$

[21] Landau and Lifshitz, *Fluid Mechanics*, § 2.

which describes the rate of change of $\mathbf{v}(\mathbf{r}, t)$ at a fixed point in space $\mathbf{r}$. Alternatively, one may refer to the *substantial derivative*

$$\mathrm{D}_t \mathbf{v}(\mathbf{r}, t)\ . \tag{2.1.9}$$

To define its meaning, one can imagine having a volume of fluid that is small compared to the volume the entire fluid occupies, but large compared to the fluid molecules. This mesoscopic fluid volume, sometimes called *fluid element*, can be thought to be marked in a way that distinguishes it from all the other parts of the fluid, e.g. by being dyed with a specific color. As time passes, the fluid element may not only change its shape – an effect that shall not be of concern here, provided one can still ascribe a position to the fluid element in a meaningful way – but may also have its position (i.e. its center of mass) moved in space.

It is the substantial derivative that describes how the velocity of the fluid element changes with time, and it has two contributions: On the one hand, the fluid element at position $\mathbf{r}$ witnesses how the velocity of the fluid at the fixed point $\mathbf{r}$ changes. On the other hand, since the fluid element is being transported from $\mathbf{r}(t)$ at time t to another position $\mathbf{r}(t + \mathrm{d}t)$ at time $t + \mathrm{d}t$, it also witnesses the fluid's velocity $\mathbf{v}(\mathbf{r}(t + \mathrm{d}t))$ at that new position.

Formally, the substantial derivative can be understood as the total time derivative of the fluid element's velocity as a function of time, $\mathbf{v}(\mathbf{r}(t), t)$, which can be found using the chain rule:

$$\begin{aligned} \mathrm{D}_t v_i &= \partial_t v_i + (\partial_j v_i)(\partial_t r_j) \\ &= \partial_t v_i + v_j \partial_j v_i\,. \end{aligned} \tag{2.1.10}$$

Since this relation holds for any $\mathbf{r}$- and t-dependent vector field for which the right-hand side of the above relation is defined, one can declare

$$\mathrm{D}_t := \partial_t + v_j \partial_j\,. \tag{2.1.11}$$

2.1.3 Euler's Equation for Ideal Fluids[22]

Given a fixed volume V with boundary ∂V and differential surface element $\mathrm{d}\mathbf{f}$, the force on the fluid volume due to (isotropic) pressure $p(\mathbf{r}, t)$ of the surrounding medium is given by

$$-\oint\limits_{\partial V} p(\mathbf{r}, t)\ \mathrm{d}f_i\,, \tag{2.1.12}$$

[22] Landau and Lifshitz, *Fluid Mechanics*, § 2.

which can be re-written, using Gauss's theorem, as

$$-\oint\limits_{\partial V} p\left(\mathbf{r},t\right)\,\mathrm{d}f_i = -\int\limits_{V} \partial_i p\left(\mathbf{r},t\right)\,\mathrm{d}^3 r\,. \tag{2.1.13}$$

If V is taken to be infinitesimally small, the force density acting on V due to pressure is thus

$$-\partial_i p\left(\mathbf{r},t\right)\,. \tag{2.1.14}$$

With other external force densities f_i^{ext} acting on this volume element, the total force density then equals, according to Newton's second law, mass density times acceleration:

$$\varrho \mathrm{D}_t v_i = f_i^{\mathrm{ext}} - \partial_i p\,. \tag{2.1.15}$$

Insertion of the definition of the substantial derivative (2.1.11) and division by ϱ then leads to Euler's equation for *ideal fluids*, i.e. fluids in which energy dissipation is negligible:

$$\partial_t v_i + v_j \partial_j v_i = \varrho^{-1}\left(f_i^{\mathrm{ext}} - \partial_i p\right)\,. \tag{2.1.16}$$

2.1.4 Momentum Flux[23]

In order to study the evolution of the fluid's momentum density ϱv_i, one can use the continuity equation (2.1.7) to calculate

$$\begin{aligned}\partial_t\left(\varrho v_i\right) &= \varrho\partial_t v_i + v_i \partial_t \varrho \\ &= \varrho\partial_t v_i - v_i \partial_j\left(\varrho v_j\right)\,.\end{aligned} \tag{2.1.17}$$

Euler's equation (2.1.16) then allows the following reformulation:

$$\begin{aligned}\partial_t\left(\varrho v_i\right) &= f_i^{\mathrm{ext}} - \partial_i p - \varrho v_j \partial_j v_i - v_i \partial_j\left(\varrho v_j\right) \\ &= f_i^{\mathrm{ext}} - \partial_i p - \partial_j\left(\varrho v_i v_j\right) \\ &= f_i^{\mathrm{ext}} - \delta_{ij}\partial_j p - \partial_j\left(\varrho v_i v_j\right)\,.\end{aligned} \tag{2.1.18}$$

Defining the *momentum flux density tensor*

$$\Pi_{ij} := p\delta_{ij} + \varrho v_i v_j\,, \tag{2.1.19}$$

[23] Landau and Lifshitz, *Fluid Mechanics*, § 7.

one arrives at

$$\partial_t \left(\varrho v_i\right) = f_i^{\mathrm{ext}} - \partial_j \Pi_{ij} \,. \tag{2.1.20}$$

The physical meaning of Π_{ij} can be understood by integrating equation (2.1.20) for $f_i^{\mathrm{ext}} = 0$ over an arbitrary, fixed volume V:

$$\begin{aligned} \partial_t \int\limits_V \varrho v_i \,\mathrm{d}^3 r &= - \int\limits_V \partial_j \Pi_{ij} \,\mathrm{d}^3 r \\ &= - \oint\limits_{\partial V} \Pi_{ij} \,\mathrm{d} f_j \,. \end{aligned} \tag{2.1.21}$$

Here, the left-hand side describes the rate of change of the i-component of the momentum in V (not accounting for external forces), while the right-hand side is a surface integral over Π_{ij}. Thus, Π_{ij} is the i-component of the momentum flowing out of the volume through a unit surface perpendicular to the j-axis during unit time. Then, with n_j being a unit vector, the i-component of the momentum flowing through a unit surface perpendicular to n_j per unit time is given by $\Pi_{ij} n_j$.

2.1.5 The Navier-Stokes Equation[24,25]

The previous sections described an ideal fluid, where no internal friction occurs. In order to introduce viscosity, one has to subtract a term σ'_{ij}, called *viscous stress tensor*, from the momentum flux density tensor (2.1.19):

$$\Pi_{ij} := p\delta_{ij} + \varrho v_i v_j - \sigma'_{ij} \,. \tag{2.1.22}$$

Defining the *stress tensor*

$$\sigma_{ij} := -p\delta_{ij} + \sigma'_{ij} \,, \tag{2.1.23}$$

one can also write

$$\Pi_{ij} = \varrho v_i v_j - \sigma_{ij} \,. \tag{2.1.24}$$

To derive an expression for σ'_{ij}, one has to note there is no friction between neighboring fluid elements that move with the same velocity in the same direction; thus, in the case of locally constant velocities, one must have $\sigma'_{ij} = 0$. Therefore, σ'_{ij} can neither have a constant

[24] Landau and Lifshitz, *Fluid Mechanics*, § 15.

[25] G. K. Batchelor. *An Introduction to Fluid Dynamics.* Cambridge Mathematical Library. Cambridge University Press, 2000, § 3.3.

contribution, nor can it depend on $\mathbf{v}$ directly. We adopt the conventional ansatz, which assumes that σ'_{ij} depends linearly on the first spatial derivatives of $\mathbf{v}$, so that

$$\sigma'_{ij} = A_{ijkl}\partial_k v_l \,, \tag{2.1.25}$$

where the A_{ijkl} do not depend on $\mathbf{v}$ or any of its derivatives.

If the entire fluid undergoes rotational movement around an axis $\boldsymbol{\omega}$ as if the fluid was a rigid body, i.e. if $\mathbf{v}(\mathbf{r},t) = \boldsymbol{\omega} \times \mathbf{r}(t)$ with constant $\boldsymbol{\omega}$, then there cannot be friction, since in the co-rotating frame of reference, the fluid is at rest. So, in this case, one also has to have $\sigma'_{ij} = 0$; consequently,

$$\begin{aligned} \sigma'_{ij} &= A_{ijkl}\partial_k \varepsilon_{lmn}\omega_m r_n \\ &= A_{ijkl}\varepsilon_{lmn}\omega_m \partial_k r_n \\ &= A_{ijkl}\varepsilon_{lmn}\omega_m \delta_{kn} \\ &= A_{ijkl}\varepsilon_{mkl}\omega_m \overset{!}{=} 0 \,. \end{aligned} \tag{2.1.26}$$

Decomposing

$$A_{ijkl} = \frac{1}{2}\left(A_{ijkl} + A_{ijlk}\right) + \frac{1}{2}\left(A_{ijkl} - A_{ijlk}\right) \tag{2.1.27}$$

into a part

$$A_{ij(kl)} = \frac{1}{2}\left(A_{ijkl} + A_{ijlk}\right) \tag{2.1.28}$$

symmetric in the indices k and l, and a part

$$A_{ij[kl]} = \frac{1}{2}\left(A_{ijkl} - A_{ijlk}\right) \tag{2.1.29}$$

anti-symmetric in k and l, and noticing that the contraction of the anti-symmetric Levi-Civita tensor ε_{ijk} with a symmetric tensor S_{jk} gives 0 (cf. appendices D.2 and D.3), one arrives at

$$A_{ij[kl]}\varepsilon_{mkl}\omega_m = 0 \tag{2.1.30}$$

for arbitrary $\boldsymbol{\omega}$. However, this means (see (D.3.4)) that

$$A_{ij[kl]} = 0 \,, \tag{2.1.31}$$

so that A_{ijkl} is symmetric in the indices k and l.

If one imposes isotropy on A_{ijkl}, i.e. if A_{ijkl} is to be indifferent to rotations of the Cartesian coordinate system, its most general form can be written as[26]

$$A_{ijkl} = s_1 \delta_{ij} \delta_{kl} + s_2 \left(\delta_{ik} \delta_{jl} + \delta_{il} \delta_{jk} \right) + s_3 \left(\delta_{ik} \delta_{jl} - \delta_{il} \delta_{jk} \right) \tag{2.1.32}$$

with scalars s_1, s_2, and s_3, which are arbitrary *a priori*.

The requirement of symmetry in the indices k and l found in (2.1.31), however, requires that $s_3 = 0$. With this, the ansatz (2.1.25) becomes

$$\begin{aligned} \sigma'_{ij} &= s_1 \delta_{ij} \delta_{kl} \partial_k v_l + s_2 \left(\delta_{ik} \delta_{jl} + \delta_{il} \delta_{jk} \right) \partial_k v_l \\ &= s_1 \delta_{ij} \partial_k v_k + s_2 \left(\partial_i v_j + \partial_j v_i \right) \\ &= \left(s_1 + \frac{2}{3} s_2 \right) \delta_{ij} \partial_k v_k + s_2 \left(\partial_i v_j + \partial_j v_i - \frac{2}{3} \delta_{ij} \partial_k v_k \right) . \end{aligned} \tag{2.1.33}$$

Introducing the *dynamic viscosity*[27] $\eta := s_2$ and the *second viscosity* $\zeta := s_1 + \frac{2}{3} s_2$, the viscous stress tensor σ'_{ij} can then be re-written in the form

$$\sigma'_{ij} = \zeta \delta_{ij} \partial_k v_k + \eta \left(\partial_i v_j + \partial_j v_i - \frac{2}{3} \delta_{ij} \partial_k v_k \right) . \tag{2.1.34}$$

This representation shows that the coefficient of η becomes zero if i and j are contracted, so that the trace of the viscous stress tensor becomes

$$\sigma'_{ii} = 3 \zeta \partial_i v_i \, . \tag{2.1.35}$$

The expression

$$\dot{\gamma}_{ij} := \partial_i v_j + \partial_j v_i \tag{2.1.36}$$

is called the *rate-of-strain tensor* or *rate-of-deformation tensor*,[28] which can be used to rewrite (2.1.34) as

$$\sigma'_{ij} = \left(\zeta - \frac{2}{3} \eta \right) \delta_{ij} \partial_k v_k + \eta \dot{\gamma}_{ij} \, . \tag{2.1.37}$$

This new contribution σ'_{ij} to Π_{ij}, introduced in order to take viscosity into account, leads to an additional term in Euler's equation (2.1.16), which now reads

$$\varrho \left(\partial_t v_i + v_j \partial_j v_i \right) = f_i^{\text{ext}} - \partial_i p + \partial_j \sigma'_{ij} \, . \tag{2.1.38}$$

[26] H. Jeffreys. *Cartesian Tensors.* Cambridge University Press, 1931, Chapter VII, equation (20).

[27] For the dynamic viscosity, both the symbols η and μ are used in the literature.

[28] Bird, Armstrong, and Hassager, *Fluid Mechanics*, § 1.2.

In the case where $f_i^{\text{ext}} = 0$, $\partial_i \eta = 0$, and $\partial_i \zeta = 0$, this relation can be simplified to

$$\begin{aligned} \varrho \left(\partial_t v_i + v_j \partial_j v_i\right) &= -\partial_i p + \zeta \partial_i \partial_k v_k + \eta \left(\partial_j \partial_i v_j + \partial_j \partial_j v_i - \frac{2}{3} \partial_i \partial_k v_k\right) \\ &= -\partial_i p + \eta \partial_j \partial_j v_i + \left(\zeta + \frac{\eta}{3}\right) \partial_i \partial_j v_j \end{aligned} \tag{2.1.39}$$

which is known as the *Navier-Stokes equation.* For a general external force density f_i^{ext}, this relation becomes

$$\varrho \left(\partial_t v_i + v_j \partial_j v_i\right) = f_i^{\text{ext}} - \partial_i p + \eta \partial_j \partial_j v_i + \left(\zeta + \frac{\eta}{3}\right) \partial_i \partial_j v_j \,. \tag{2.1.40}$$

2.1.6 Incompressibility[29]

If a fluid element's density is constant throughout time, i.e. if

$$\mathrm{D}_t \varrho = 0 \,, \tag{2.1.41}$$

the flow is called *incompressible*; if this condition is met for all types of flow for a given liquid, that liquid itself is called incompressible. Rewriting the continuity equation (2.1.7) as

$$\begin{aligned} 0 &= v_i \partial_i \varrho + \varrho \partial_i v_i + \partial_t \varrho \\ &= \varrho \partial_i v_i + \mathrm{D}_t \varrho \,, \end{aligned} \tag{2.1.42}$$

it is evident that the incompressibility condition (2.1.41) can be equivalently formulated as

$$\partial_i v_i = 0 \,. \tag{2.1.43}$$

For incompressible flows, the viscous stress tensor becomes

$$\begin{aligned} \sigma'_{ij} &= \eta \left(\partial_i v_j + \partial_j v_i\right) \\ &= \eta \dot{\gamma}_{ij} \end{aligned} \tag{2.1.44}$$

and therefore the stress tensor reads

$$\begin{aligned} \sigma_{ij} &= -p \delta_{ij} + \eta \left(\partial_i v_j + \partial_j v_i\right) \\ &= -p \delta_{ij} + \eta \dot{\gamma}_{ij} \,. \end{aligned} \tag{2.1.45}$$

[29] Landau and Lifshitz, *Fluid Mechanics*, § 15.

The external-force Navier-Stokes equation reduces to

$$\varrho \partial_t v_i + \varrho v_j \partial_j v_i = f_i^{\text{ext}} - \partial_i p + \eta \partial_j \partial_j v_i \,, \tag{2.1.46}$$

or equivalently,

$$\partial_t v_i + v_j \partial_j v_i = \varrho^{-1} f_i^{\text{ext}} - \varrho^{-1} \partial_i p + \nu \partial_j \partial_j v_i \,, \tag{2.1.47}$$

where the *kinematic viscosity* $\nu := \eta/\varrho$ has been introduced.

2.2 Adaptions of the Navier-Stokes Equation

2.2.1 The Linearized Navier-Stokes Equation

The dimensionless Reynolds number[30,31] $\mathrm{Re} := ul/\nu$, which is defined in terms of the kinematic viscosity ν, a characteristic length l (such as a linear extent of some rigid boundary), and a characteristic velocity u (such as the steady speed of said moving rigid boundary), serves as a measure for the importance of the inertial forces, compared to pressure and viscous forces.[32,33] For example, the Reynolds number's order of magnitude for a human swimming in water is about $\mathrm{Re} \approx 10^4$, while for an organism about $2\,\mu\mathrm{m}$ in size, such as the bacterium *E. coli*, the Reynolds number is $\mathrm{Re} \approx 10^{-4}$ or even less.[34]

In terms of these characteristic scales u, l, and ν, the summand $v_j \partial_j v_i$ in equation (2.1.47), is roughly of the order of magnitude[35] u^2/l, while the term $\nu \partial_j \partial_j v_i$ is of order of magnitude $\nu u/l^2$; the ratio of these two (crude) estimates is exactly the Reynolds number Re. So, for small Reynolds numbers, i.e. for $\mathrm{Re} \ll 1$, the term $\varrho v_j \partial_j v_i$ becomes small compared to $\eta \partial_j \partial_j v_i$, and thus can be neglected in equation (2.1.47). Omission of this term, which is non-linear in $\mathbf{v}$, thus leads to the *linearized Navier-Stokes equation*[36]

$$\varrho \partial_t v_i = f_i^{\text{ext}} - \partial_i p + \eta \partial_j \partial_j v_i \,. \tag{2.2.1}$$

[30] Landau and Lifshitz, *Fluid Mechanics*, § 19.
[31] Batchelor, *An Introduction to Fluid Dynamics*, § 4.7.
[32] Batchelor, *An Introduction to Fluid Dynamics*, § 4.8.
[33] E. M. Purcell. “Life at low Reynolds number”. *American Journal of Physics* 45 (1977), 3.
[34] Purcell, “Life at low Reynolds number”.
[35] Landau and Lifshitz, *Fluid Mechanics*, § 20.
[36] Landau and Lifshitz, *Fluid Mechanics*, § 20.

2.2.2 Linear Viscoelasticity

One can generalize the derivation of the Navier-Stokes equation for incompressible fluids (see sections 2.1.5 and 2.1.6), by introducing an explicit time-dependence in the dynamic viscosity η and in the rate-of-strain tensor $\dot{\gamma}_{ij}$, and connecting them in a convolution term.[37,38,39] Then, instead of (2.1.45), the relationship between σ_{ij} and $\dot{\gamma}_{ij}$ reads

$$\sigma_{ij}\left(\mathbf{r},t\right) = -p\left(\mathbf{r},t\right)\delta_{ij} + \int\limits_{t'=-\infty}^{t} G\left(t-t'\right)\dot{\gamma}_{ij}\left(\mathbf{r},t'\right)\,\mathrm{d}t'\,, \tag{2.2.2}$$

where the real quantity $G\left(\Delta t\right)$ is called the *relaxation modulus*; it is assumed to be independent of the spatial coordinates $\mathbf{r}$. $G\left(\Delta t\right)$ describes how strain rates that occur during at a time Δt prior to t influence the stress of the fluid at time t. The range of integration, $t' \in (-\infty, t]$, is chosen accordingly, since it corresponds to $\Delta t \in [0, \infty]$, so that the system's state is determined only by its past, but not by its future. $G\left(\Delta t\right)$ is positive and monotonically decreasing[40] to 0 with increasing Δt, provided that one deals with viscoelastic *liquids*; the condition $\lim_{\Delta t \to \infty}\left(G\left(\Delta t\right)\right) = 0$ means that the system is stress-free if a strain was placed on it only in the infinitely far past. This property is characteristic of liquids;[41,42] for a viscoelastic *solid*, this assumption does not hold.

A quantity related to the relaxation modulus is the *complex modulus*[43]

$$G^*\left(\omega\right) = \mathrm{i}\omega\int\limits_{t=0}^{\infty} G\left(t\right)\exp\left(-\mathrm{i}\omega t\right)\,\mathrm{d}t\,, \tag{2.2.3}$$

[37] Bird, Armstrong, and Hassager, *Fluid Mechanics*, § 5.2d, equation (5.2-18).

[38] J. D. Ferry. *Viscoelastic Properties of Polymers.* 3rd ed. John Wiley & Sons, 1980, Section 1.B4, equation (7).

[39] Ferry, *Viscoelastic Properties of Polymers*, Section 1.G1, equation (59).

[40] Bird, Armstrong, and Hassager, *Fluid Mechanics*, § 5.2d.

[41] Ferry, *Viscoelastic Properties of Polymers*, Section 1.B4.

[42] Ferry, *Viscoelastic Properties of Polymers*, Section 1.G.

[43] M. Doi and S. F. Edwards. *The Theory of Polymer Dynamics.* Oxford University Press, 1994, Chapter 7.3, equation (7.22).

which is decomposed[44] into its real part, called the *storage modulus*[45]

$$\begin{aligned} G'(\omega) &= \mathrm{Re}\left(G^*(\omega)\right) \\ &= \omega \int\limits_{t=0}^{\infty} G(t) \sin(\omega t)\, \mathrm{d}t\,, \end{aligned} \tag{2.2.4}$$

and its imaginary part, called the *loss modulus*:[46]

$$\begin{aligned} G''(\omega) &= \mathrm{Im}\left(G^*(\omega)\right) \\ &= \omega \int\limits_{t=0}^{\infty} G(t) \cos(\omega t)\, \mathrm{d}t\,. \end{aligned} \tag{2.2.5}$$

With equation (2.2.2), the correspondingly extended Euler equation (2.1.38) reads, under the incompressibility condition $\partial_i v_i = 0$,

$$\begin{aligned} \varrho\left(\partial_t v_i + v_j \partial_j v_i\right) &= f_i^{\mathrm{ext}} - \partial_i p + \int\limits_{t'=-\infty}^{t} G(t-t')\, \partial_j \dot{\gamma}_{ij}(t')\, \mathrm{d}t' \\ &= f_i^{\mathrm{ext}} - \partial_i p + \int\limits_{t'=-\infty}^{t} G(t-t')\, \partial_j \partial_j v_i(t')\, \mathrm{d}t'\,, \end{aligned} \tag{2.2.6}$$

where some dependencies on $\mathbf{r}$ and t have been omitted for notational clarity. With this, the linearized Navier-Stokes equation (2.2.1) for linearly viscoelastic fluids reads

$$\varrho \partial_t v_i = f_i^{\mathrm{ext}} - \partial_i p + \int\limits_{t'=-\infty}^{t} G(t-t')\, \partial_j \partial_j v_i(t')\, \mathrm{d}t'\,. \tag{2.2.7}$$

[44] Doi and Edwards, *The Theory of Polymer Dynamics*, Chapter 4.5.1.

[45] Doi and Edwards, *The Theory of Polymer Dynamics*, Equation (7.23).

[46] Doi and Edwards, *The Theory of Polymer Dynamics*, Equation (7.23).

Chapter 3

Multi-Particle Collision Dynamics

3.1 Introduction: MPC for Simple Fluids

Multi-Particle Collision Dynamics (MPC or MPCD for short) is a class of particle-based simulation techniques, introduced by Malevantes and Kapral.[47]

It is instructional to consider first the simplest variant of an MPC system in three spatial dimensions: Let the coordinate system be Cartesian, with the axes named x, y, and z, or alternatively 1, 2, and 3, respectively. Let the primary simulation volume be a cuboid of side lengths $L_i = l_i a$, $i \in \{1, 2, 3\}$, where $l_i \in \mathbb{N}_+$. So, L_i is a positive integer multiple of a characteristic length a, which serves as the length scale of the system and as such is set to unity. The origin of the coordinate system is chosen such that it coincides with a corner of the primary simulation volume, and the axes are aligned such that the primary simulation volume lies entirely within the first octant of the coordinate system.

The fluid is modeled by a number $N_{\text{MPC}} \in \mathbb{N}_+$ of point-like MPC particles. Each of them is an abstract representation of a volume of the fluid that is large compared to the individual fluid molecules, but small compared to $V := L_1 L_2 L_3$, the volume of the simulated system. Each MPC particle $i \in \mathbb{I} := \{x \in \mathbb{N}_+ \mid x \leq N_{\text{MPC}}\}$ has a position, taking on continuous values $\mathbf{r}_i \in \mathbb{R}^3$, which, depending on the boundary conditions applied and the way these are implemented, may be further restricted to some continuous subset of $\mathbb{R}^3$. The MPC particle velocities, $\mathbf{v}_i \in \mathbb{R}^3$, are unconstrained. For simple fluids, each MPC particle has the same mass $m_i = m$, which is the system's reference mass and is set to 1.

The MPC algorithm then consists of two alternating phases. In the *streaming phase*, each MPC particle is propagated independently from all other particles. With the fixed propagation

[47] Malevanets and Kapral, "Mesoscopic model for solvent dynamics".

time being called Δt_{MPC}, and ignoring here, for the sake of simplicity, both the possible presence of external force fields and the issue of boundary conditions (see section 3.1.3), the streaming step thus is simply an update of the particle positions in accordance with ballistic motion:

$$\mathbf{r}_i\left(t+\Delta t_{\mathrm{MPC}}\right)=\mathbf{r}_i\left(t\right)+\mathbf{v}_i\left(t\right)\cdot\Delta t_{\mathrm{MPC}}\,. \tag{3.1.1}$$

For the subsequent *collision phase*, the primary simulation volume is partitioned into cubic *collision cells*, the side lengths of which are what defines the system's length scale a. The collision cells are tiled such that each point in the simulation volume can be uniquely assigned to one collision cell. Then, all MPC particles that are momentarily located within a given collision cell interact with one another (but not with any particles outside that collision cell), in a way that conserves the total mass, linear momentum, and energy contained in each individual collision cell.[48]

With these properties, the only missing ingredient essential to the derivation of the Navier-Stokes equation (cf. section 2.1) is spatial isotropy. Since the orientation of the collision cell cubes distinguishes three directions, the system is not strictly isotropic. However, it is conventional to assume that isotropy is satisfied to a sufficient degree, such that, for suitable simulation parameters (such as the simulation time-step Δt_{MPC}), one can expect MPC to reproduce Navier-Stokes-like behavior.[49,50,51]

3.1.1 Stochastic Rotation Dynamics (SRD)

In the initial and most common variant of MPC, called *stochastic rotation dynamics*[52] and abbreviated *SRD*, the collision rule is as follows: For each collision cell c, the largest set $\mathbb{I}_c \subseteq \mathbb{I}$ of MPC particle indices with the property that for each $i \in \mathbb{I}_c$, $\mathbf{r}_i$ lies in the collision cell c, is identified. Then, the collision cell's center-of-mass velocity

$$\boldsymbol{\mathcal{V}}_c := \frac{\sum_{i\in\mathbb{I}_c} m_i \mathbf{v}_i}{\sum_{i\in\mathbb{I}_c} m_i} \tag{3.1.2}$$

[48] Gompper, Ihle, Kroll, and Winkler, *Multi-Particle Collision Dynamics: A Particle-Based Mesoscale Simulation Approach to the Hydrodynamics of Complex Fluids*, Chapter 1.

[49] U. Frisch, B. Hasslacher, and Y. Pomeau. "Lattice-Gas Automata for the Navier-Stokes Equation". *Physical Review Letters* 56 (1986), 1505.

[50] Gompper, Ihle, Kroll, and Winkler, *Multi-Particle Collision Dynamics: A Particle-Based Mesoscale Simulation Approach to the Hydrodynamics of Complex Fluids*, Chapter 1.

[51] T. Ihle and D. M. Kroll. "Stochastic rotation dynamics. I. Formalism, Galilean invariance, and Green-Kubo relations". *Physical Review E* 67 (2003), 066705. Section I.A.

[52] T. Ihle and D. M. Kroll. "Stochastic rotation dynamics: A Galilean-invariant mesoscopic model for fluid flow". *Physical Review E* 63 (2001), 020201.

is calculated, where $\mathbf{v}_i$ is the i-th MPC particle velocity prior to the collision. With this, the MPC particle velocities in the center-of-mass frame of the collision cell are obtained:

$$\boldsymbol{\mathfrak{v}}_i := \mathbf{v}_i - \boldsymbol{\mathcal{V}}_c \,. \tag{3.1.3}$$

Next, in the collision cell's center-of-mass frame, the $\boldsymbol{\mathfrak{v}}_i$ are rotated by a fixed angle α around a randomly chosen unit vector $\mathbf{R}$; it is uniformly sampled from $\mathbb{S}^2$ (cf. appendix E), independently so for each collision cell and time-step. Decomposing $\boldsymbol{\mathfrak{v}}_i = \boldsymbol{\mathfrak{v}}_i^{\perp} + \boldsymbol{\mathfrak{v}}_i^{\parallel}$ uniquely into a part $\boldsymbol{\mathfrak{v}}_i^{\parallel} := (\boldsymbol{\mathfrak{v}}_i \cdot \mathbf{R})\,\mathbf{R}$ parallel to $\mathbf{R}$ and a part $\boldsymbol{\mathfrak{v}}_i^{\perp} := \boldsymbol{\mathfrak{v}}_i - \boldsymbol{\mathfrak{v}}_i^{\parallel}$ perpendicular to $\mathbf{R}$, the result of the rotation of $\boldsymbol{\mathfrak{v}}_i$ around $\mathbf{R}$ is[53]

$$\begin{aligned}
\boldsymbol{\mathfrak{v}}_i' &:= \boldsymbol{\mathfrak{v}}_i^{\parallel} + \cos(\alpha)\,\boldsymbol{\mathfrak{v}}_i^{\perp} + \left|\boldsymbol{\mathfrak{v}}_i^{\perp}\right| \sin(\alpha)\,\frac{\mathbf{R} \times \boldsymbol{\mathfrak{v}}_i}{\left|\mathbf{R} \times \boldsymbol{\mathfrak{v}}_i\right|} \\
&= \boldsymbol{\mathfrak{v}}_i^{\parallel} + \cos(\alpha)\,\boldsymbol{\mathfrak{v}}_i^{\perp} + \sin(\alpha)\,(\mathbf{R} \times \boldsymbol{\mathfrak{v}}_i) \\
&= (\boldsymbol{\mathfrak{v}}_i \cdot \mathbf{R})\,\mathbf{R} + \cos(\alpha)\,(\boldsymbol{\mathfrak{v}}_i - (\boldsymbol{\mathfrak{v}}_i \cdot \mathbf{R})\,\mathbf{R}) + \sin(\alpha)\,(\mathbf{R} \times \boldsymbol{\mathfrak{v}}_i)\ ,
\end{aligned} \tag{3.1.4}$$

since $\left|\mathbf{R} \times \boldsymbol{\mathfrak{v}}_i\right| = \left|\boldsymbol{\mathfrak{v}}_i^{\perp}\right|$. The post-collision velocities obtained in the simulation system's frame of reference then are finally

$$\mathbf{v}_i' := \boldsymbol{\mathcal{V}}_c + \boldsymbol{\mathfrak{v}}_i' \,. \tag{3.1.5}$$

Since the collision cell's center-of-mass frame is defined by the condition that the sum of all linear momenta in that reference frame is zero, i.e. $\sum_{i \in \mathbb{I}_c} m_i \boldsymbol{\mathfrak{v}}_i = 0$, rotation of all $\boldsymbol{\mathfrak{v}}_i$ around the same axis conserves the linear momentum of the entire collision cell. Also, the collision cell's mass content is conserved, so that energy is unchanged as well. Finally, since the procedure is evidently isotropic, all the conditions for an MPC collision step are satisfied (cf. section 3.1).

It should be noted, however, that SRD violates conservation of angular momentum. While this deficiency can be repaired at the cost of computational efficiency,[54] in most simulations where MPC particles move slowly compared to the speed of sound[55] the deviation from Navier-Stokes-type behavior due to unphysical changes in angular momentum during the collision step is negligible.[56]

[53] D. Koks. *Explorations in Mathematical Physics.* 1st ed. Springer, 2006. Chapter 4.2.

[54] H. Noguchi, N. Kikuchi, and G. Gompper. "Particle-based mesoscale hydrodynamic techniques". *Europhysics Letters* (2007).

[55] Landau and Lifshitz, *Fluid Mechanics*, § 44.

[56] Gompper, Ihle, Kroll, and Winkler, *Multi-Particle Collision Dynamics: A Particle-Based Mesoscale Simulation Approach to the Hydrodynamics of Complex Fluids*, Chapter 2.1.1.

3.1.2 Grid Shift[57,58,59]

If one uses a fixed grid to define the collision cells, one breaks the system's symmetry under Galilei transformations: Imagine a fluid in a situation where the mean free path λ during the time-step Δt_{MPC} is substantially smaller than the collision cell size a. Then, given a particular collision cell c, the set of MPC particles $\mathbb{I}_c(t)$ in that cell at time t is going to contain mostly the same members as $\mathbb{I}_c(t + \Delta t_{\text{MPC}})$, the set of MPC particles in the same collision cell c in the next time-step. Statistically, the states of the MPC particles in $\mathbb{I}_c(t)$ are therefore going to be correlated over a timespan large compared to the streaming time-step Δt_{MPC}.

However, if one superimposes a global, fixed, and non-zero velocity $\mathbf{u}$ on the entire system, the correlation time changes in general, since now, the sets $\mathbb{I}_c(t)$ and $\mathbb{I}_c(t + \Delta t_{\text{MPC}})$ may share less members. This means that the statistical properties of a system's MPC particles depend on the observer's inertial frame, thus breaking Galilean symmetry.

In the case where λ is large compared to a, this effect is negligible, since then $\mathbb{I}_c(t)$ and $\mathbb{I}_c(t + \Delta t_{\text{MPC}})$ are mostly disjoint for arbitrary $\mathbf{u}$.

This deficiency of broken Galilean symmetry can be eliminated by independently sampling three random numbers X_1, X_2, and X_3 from $U\left(\left[-\frac{a}{2}, \frac{a}{2}\right]\right)$ and shifting either the entire collision cell grid by $\mathbf{X} = (X_1, X_2, X_3)$ with reference to its fixed position in the previous scenario, or equivalently, by shifting the positions of all MPC particles by $-\mathbf{X}$ (for the handling of boundary conditions see section 3.1.3).

This grid shift does not, however, fix the broken rotational symmetry due to the grid's cubic unit cell, which distinguishes three spatial directions in the system. Like most of the literature, the present work will not deal with this issue further.

3.1.3 Boundary Conditions

The boundary conditions used in this work are *periodic boundary conditions* and *Lees-Edwards boundary conditions*, both described below. Of course, others can be used with MPC as well, such as *no-slip boundary conditions* in scenarios where extended solids, such as confining walls, are involved: there, one imposes that the velocity component tangential to the surface of the solid-fluid interface is continuous across the interface, i.e. the fluid does not slip along the solid's

[57] Ihle and Kroll, "Stochastic rotation dynamics: A Galilean-invariant mesoscopic model for fluid flow".

[58] Ihle and Kroll, "Stochastic rotation dynamics. I. Formalism, Galilean invariance, and Green-Kubo relations", Section A.

[59] Gompper, Ihle, Kroll, and Winkler, *Multi-Particle Collision Dynamics: A Particle-Based Mesoscale Simulation Approach to the Hydrodynamics of Complex Fluids*, Chapter 2.1.

surface – a condition experimentally verified under certain conditions.[60] However, implementing these boundary conditions without introducing artifacts is non-trivial.[61,62]

3.1.3.1 Periodic Boundary Conditions

Periodic boundary conditions are defined by the property that, during the collision step, each MPC particle i has its position vector components $r_{i,j}$, $j \in \{1, 2, 3\}$, mapped to the *image coordinates*

$$r'_{i,j} := r_{i,j} + c_{i,j} L_j \, , \tag{3.1.6}$$

where the $c_{i,j} \in \mathbb{Z}$ are chosen such that $r'_{i,j} \in [0, L_j)$.

For the collision step, it is only these image coordinates that are of importance. However, the real MPC particle coordinates $r_{i,j}$ are not replaced by these image coordinates, so that during the streaming steps, the MPC particles effectively move in an unbounded system, that is only virtually folded back into the primary simulation volume for MPC collisions.

This procedure makes the handling of distances easier, for example when observing diffusion behavior: if one is interested in how far a certain MPC particle travels during a given time interval, one does not have to keep track of how often the primary simulation volume's boundaries have been crossed.

There is a possible downside to having the MPC particles propagate without bounds: if particles move far away from the origin, the numerical accuracy of the floating-point variables storing the positions degrades, as the density of representable real numbers decreases with increasing modulus.[63,64] However, this effect would be noticeable only for simulations running many orders of magnitude longer than what was necessary in this work.

[60] Batchelor, *An Introduction to Fluid Dynamics*, Chapter 3.3.

[61] A. Lamura, G. Gompper, T. Ihle, and D. M. Kroll. "Multi-particle collision dynamics: Flow around a circular and a square cylinder". *Europhysics Letters* 56 (2001), 319.

[62] A. Lamura and G. Gompper. "Numerical study of the flow around a cylinder using multi-particle collision dynamics". *The European Physical Journal E - Soft Matter* 9 (2002), 477.

[63] *IEEE Standard for Floating-Point Arithmetic.* Institute of Electrical and Electronics Engineers.

[64] D. Goldberg. "What every computer scientist should know about floating-point arithmetic". *ACM Computing Surveys* 23 (1991), 5.

3.1.3.2 Lees-Edwards Boundary Conditions[65,66]

Lees and Edwards[67] introduced a generalization of the periodic boundary conditions that enables one to impose a specific, possibly time-dependent, uniform shear on the simulated system.

Let the shear act in the x-y-plane in the sense that with increasing y, the mean velocity v_x along the x direction increases, and let

$$\dot{\gamma} := \frac{\mathrm{d}v_x}{\mathrm{d}r_y} \tag{3.1.7}$$

denote the shear rate, which is time-independent in this work.

Imagine, at the beginning of the simulation, having periodic images of the primary simulation volume along the x, y, and z directions. Label the primary simulation volume by $n_x = n_y = n_z = 0$; the image adjacent to that along the positive x direction is labeled $n_x = 1$, $n_y = n_z = 0$, and so on. Then, as the simulation time progresses, imagine the layers with y-label $n_y = 1$ moving with a constant velocity

$$u := \dot{\gamma} L_y \,, \tag{3.1.8}$$

with L_y being the length of the primary simulation volume along the y direction. Similarly, the $n_y = -1$ images move with $-u$ along the x direction, and in general, an image with label n_y moves with $n_y u$ along the x direction. So, after a simulation time t, the coordinates of the origins of the images are $(n_x L_x + n_y u t, n_y L_y, n_z L_z)$ with respect to the origin of the primary simulation cell.

Recall that the MPC particles are thought to propagate in unbounded space during the streaming phase, and that it is necessary to temporarily fold the MPC particle coordinates into the primary simulation volume for the collision phase. To achieve this, one first finds, for each MPC particle i, which simulation volume image it falls into, by computing

$$n_{i,j} := \left\lfloor \frac{r_{i,j}}{L_j} \right\rfloor \tag{3.1.9}$$

[65] M. P. Allen and D. J. Tildesley. *Computer Simulation of Liquids.* Reprint by The Ipswich Book Co Ltd in 1991. Clarendon Press, 1987, Chapter 8.2.

[66] Kowalik and Winkler, "Multiparticle collision dynamics simulations of viscoelastic fluids: Shear-thinning Gaussian dumbbells", Appendix A.

[67] A. W. Lees and S. F. Edwards. "The computer study of transport processes under extreme conditions". *Journal of Physics C: Solid State Physics* 5 (1972), 1921.

for $j \in \{1, 2, 3\}$. Then, the MPC particle's image coordinates are calculated as

$$r'_{i,j} = r_{i,j} - n_{i,j} L_j - n_{i,2} n_y u t \delta_{j,1} \,, \tag{3.1.10}$$

where δ_{ij} is the Kronecker symbol. Additionally, the MPC particle's velocity along the x direction is temporarily changed to

$$v'_{i,j}(t) := v_{i,j}(t) + n_y u \delta_{j,1} \tag{3.1.11}$$

to account for the additional speed gained through the applied shear. Next, the MPC collision algorithm is executed with the image positions $\mathbf{r}'_i$ and image velocities $\mathbf{v}'_i$. Finally, the changed MPC particle velocities are transformed from the image in the primary simulation volume to the proper position in unbounded space, by undoing the transformation (3.1.11):

$$v_{i,j}(t + \Delta t_{\mathrm{MPC}}) := v'_{i,j}(t + \Delta t_{\mathrm{MPC}}) - n_y u \delta_{j,1} \,. \tag{3.1.12}$$

The Lees-Edwards boundary conditions have the periodic boundary conditions as the special case $\dot{\gamma} = 0$, as can easily be seen.

3.1.4 Maxwell-Boltzmann-Scaling Thermostat

The MPC algorithm conserves energy, and thus models a microcanonical ensemble. However, many interesting phenomena, such as the dynamics of polymers in solution, arise due to thermal fluctuations.[68] In order to introduce those fluctuations into the simulation scheme, one can employ various methods to realize a canonical ensemble where the system temperature T is fixed. Another reason one might need to couple the system to a heat bath is that, if external forces act on the MPC particles, one may introduce energy into the system that may accumulate if not removed via thermal contact.

In this work, the *Maxwell-Boltzmann-Scaling* thermostat, introduced by Huang et al.,[69] is implemented, which operates locally on the collision cells: For each collision cell c with member particle indices in $\mathbb{I}_c$, let N_c be the number of MPC particles currently being in that collision cell. If $N_c = 1$, no action is performed, since then, no change in energy can be accomplished without violating momentum conservation. In the case $N_c > 1$, a random variable E'_k is sampled from

[68] Gompper, Ihle, Kroll, and Winkler, *Multi-Particle Collision Dynamics: A Particle-Based Mesoscale Simulation Approach to the Hydrodynamics of Complex Fluids*, Chapter 1.

[69] C. Huang, A. Chatterji, G. Sutmann, G. Gompper, and R. G. Winkler. “Cell-level canonical sampling by velocity scaling for multiparticle collision dynamics simulations”. *Journal of Computational Physics* 229 (2010), 168.

the gamma distribution (A.11), with the distribution parameters being $a = \frac{f}{2}$ and $b = k_B T$; here, $f = 3\left(N_c - 1\right)$ is the number of velocity degrees of freedom left in the collision cell's center-of-mass frame, so that

$$E'_k \sim f_\Gamma\left(E'_k; \frac{f}{2}, k_B T\right) = \frac{1}{E'_k \Gamma\left(\frac{f}{2}\right)} \left(\frac{E'_k}{k_B T}\right)^{\frac{f}{2}} \exp\left(-\frac{E'_k}{k_B T}\right) . \tag{3.1.13}$$

Equation (3.1.13) is derived from the Maxwell-Boltzmann distribution of three-dimensional velocity vectors, with the constraint of momentum conservation.

With the center-of-mass velocity of MPC particle i being $\mathfrak{v}_i$, the current center-of-mass kinetic energy

$$E_k := \frac{1}{2} m \sum_{i \in \mathbb{I}_c} \mathfrak{v}_i^2 \tag{3.1.14}$$

is calculated, a random E'_k is chosen according to equation (3.1.13), and the center-of-mass velocities are scaled via

$$\mathfrak{v}_i \mapsto \sqrt{\frac{E'_k}{E_k}} \mathfrak{v}_i \, , \tag{3.1.15}$$

with the result that after scaling, the center-of-mass kinetic energy of the collision cell c is E'_k.

3.2 MPC for Complex Systems

MPC can easily be extended to simulate more complex systems than just fluids of one species, with applications ranging from the modeling of binary fluids to systems of biological cells and vesicles.[70]

In this work, the emphasis lies on the dynamics of linear polymers and the behavior of viscoelastic fluids. Here, polymers are represented by linking together several MPC particles by suitable potentials.

While it is easy to find an analytic solution for the streaming-step propagation of two MPC particles coupled by a harmonic potential,[71] the solutions become increasingly more complicated, or even inaccessible, as soon as one generalizes this to other interaction potentials and/or more constituents per polymer. So, instead of trying to find exact solutions in terms of closed expressions, the motion of the individual MPC particles is approximately calculated by

[70] Gompper, Ihle, Kroll, and Winkler, *Multi-Particle Collision Dynamics: A Particle-Based Mesoscale Simulation Approach to the Hydrodynamics of Complex Fluids*.

[71] Kowalik and Winkler, "Multiparticle collision dynamics simulations of viscoelastic fluids: Shear-thinning Gaussian dumbbells", Section III.A.

the *velocity Verlet* algorithm as follows.[72] Let $\mathbf{r}_i\left(t\right)$, $\mathbf{v}_i\left(t\right)$, and $\mathbf{a}_i\left(t\right)$ be the position, velocity, and acceleration of the i-th MPC particle at time t, respectively, and let Δt be the simulation time-step. Then, the updated positions $\mathbf{r}_i\left(t+\Delta t\right)$ are calculated as

$$\mathbf{r}_i\left(t+\Delta t\right) = \mathbf{r}_i\left(t\right) + \mathbf{v}_i\left(t\right)\Delta t + \frac{1}{2}\mathbf{a}_i\left(t\right)\left(\Delta t\right)^2 \,. \tag{3.2.1}$$

The updated velocities are computed via

$$\mathbf{v}_i\left(t+\Delta t\right) = \mathbf{v}_i\left(t\right) + \frac{\mathbf{a}_i\left(t\right) + \mathbf{a}_i\left(t+\Delta t\right)}{2}\Delta t \,. \tag{3.2.2}$$

Of course, for the accelerations $\mathbf{a}_i\left(t+\Delta t\right)$ to be available, the updated positions $\mathbf{r}_i\left(t+\Delta t\right)$ of all interaction partners have to be computed beforehand, since

$$\mathbf{a}_i\left(t\right) := -\frac{\partial}{\partial \mathbf{r}_i\left(t\right)} U\left(\mathbf{r}_1\left(t\right), \mathbf{r}_2\left(t\right), \ldots\right) \,, \tag{3.2.3}$$

where $U\left(\mathbf{r}_1\left(t\right), \mathbf{r}_2\left(t\right), \ldots\right)$ is the system's potential energy at time t. U evidently depends on the positions of the interacting particles, but the velocity Verlet algorithm in unsuitable if there is a dependence on the velocities; since the calculation of the updated velocities (3.2.2) requires knowledge of the updated accelerations, having the latter be influenced by the former would create a circular dependency.

For the velocity Verlet algorithm, being equivalent to the original Verlet algorithm,[73] an error of $\left(\Delta t\right)^4$ in the updated positions can be estimated, and an error of $\left(\Delta t\right)^2$ in the prediction of the updated velocities.[74] While velocity verlet exhibits inaccuracies in the estimation of the velocity and a violation of conservation of energy on long timescales,[75] these deficiencies are not too worrisome in SRD simulations, because first, the velocities are subjected to a random rotation during the SRD collision step anyway, and second, the system energy and velocities undergo random rescalings due to the thermostat (see section 3.1.4).

The main advantage of the velocity Verlet algorithm is its conceptual and computational simplicity. In particular, the number of values one needs to store and access are small, so that the algorithm does not additionally constrain the system size one is able to simulate by requiring storage of otherwise unneeded values. Furthermore, since accessing memory that is not local to the GPU's computation unit costs a considerable amount of time, minimizing the amount

[72] D. Frenkel and B. Smit. *Understanding Molecular Simulation. From Algorithms to Applications.* 2nd ed. Academic Press, 2002, Chapter 4.3.1.

[73] Frenkel and Smit, *Understanding Molecular Simulation*, Chapter 4.3.1.

[74] Frenkel and Smit, *Understanding Molecular Simulation*, Chapter 4.2.3.

[75] Frenkel and Smit, *Understanding Molecular Simulation*, Chapter 4.3.

of data needed to update a given polymer is beneficial for the simulation performance.

Chapter 4

MPCDSim: A Novel GPU-Based MPC Implementation

4.1 Overview and Design Goals

The basis of this thesis is an implementation of the SRD variant of MPC; uncreatively named *MPCDSim*, it has been developed from ground up, with the following design goals in mind:

- **Correctness**
  The implementation has to produce correct data, in the sense that the simulation follows the MPC algorithm as specified in chapter 3.

- **Performance**
  One advantage of MPC is its computational efficiency, which allows for large and complex systems to be simulated over extended periods of simulation-time (i.e. number of simulation steps times the simulation time-step Δt_{MPC}). However, one can easily negate this algorithmic advantage by making unfavorable code design choices which may reduce computational performance (i.e. real time taken per simulation step) drastically. While optimization of computing performance was not a primary objective, considerable effort was spent on identifying and implementing the optimizations that were characterized by the highest ratio of anticipated impact on computing time vs. developer time used.

- **Massive Parallelization**
  Since in the streaming step, (fixed groups of) MPC particles move independently from the others, and since the collision step involves only on the order of 1 to 10 MPC particles,

MPC is inherently a very parallel algorithm; the majority of computational work can be easily broken up into independent tasks, each of which requires only a small amount of computing power and memory. This makes MPC very well-suited for the use of *Graphics Processing Units* (abbreviated *GPUs*), which have, depending on the model, several thousands of computing cores working in parallel; in order to utilize them most efficiently, ideally one would schedule on the order of 10^4 threads for execution on each GPU.[76]

- **Clean and Easily Extensible Software Architecture**
  The software was designed, using the paradigm of object-oriented programming, in such a way that different tasks are handled by different parts of the code, and vice versa. This way, the code modularizes into components that are small, specialized, and easily comprehensible. Components can readily be changed, e.g. in order to test a new algorithm, without disturbing the remainder of the program. Also, new components, like a new measurement of a dynamical system property, can be added without concern for the inner workings of the bulk of the software package.

- **Documentation**
  In order for the black-box approach to work, the application programming interfaces (APIs) of the code have to be specified; for example, a given function's documentation should describe the purpose of passed parameters and conditions for their validity, the returned value, possible side effects, assumptions made about the state of the program upon entering the function, and error conditions.

  The documentation is embedded into the source code in the form of comments, and is written in a syntax compatible the program *doxygen*,[77] which allows for automatic generation of manuals in various formats, such as HTML and PDF.

- **Code Legibility**
  It was attempted to make the code as easily readable as possible: for names of symbols, such as variables and functions, descriptive names have been chosen over short ones. Also, effort was spent to make routines small and compact; if a function grew too large, oftentimes that was due to the function doing two separable tasks, which then were refactored into smaller functions. This way, even readers unfamiliar with the code should have an,

[76] A. V. Adinetz. "GPUs and Other Non-Standard Hardware". *Computing Solids: Models, ab-initio methods and supercomputing.* Ed. by S. Blügel, N. Helbig, V. Meden, and D. Wortmann. Vol. 74. Key Technologies. Forschungszentrum Jülich, 2014. Chap. D5, Chapter 2.2.

[77] D. van Heesch. *Doxygen.* URL: `http://www.stack.nl/~dimitri/doxygen/manual/index.html`.

at least vague, idea what any particular code passage does. If the functionality still was non-obvious, for example due to algorithmic or mathematical complexities, appropriate comments were inserted into the code explaining the passage's purpose.

- **Traceability**
  As features are added and errors are corrected, the source code undergoes a lot of changes. To ensure that data produced by any given version of the software can be checked for systematic errors, and in order to provide the possibility to trace potential errors to where they were introduced, the source control management program *git*[78] was used.

While the degree of success in achieving the design goals has to be judged subjectively, there certainly is potential for improvement regarding performance and scalability, at least for simulations of simple fluids:[79] The implementation presented here is currently incapable of using more than one GPU in parallel. This is primarily a constraint on the available GPU memory, and thus on the system size. However, since the majority of applications for *MPCDSim* do not require any given simulation to run for extended periods of simulation-time (say, $2t$), multi-GPU systems can still be used efficiently by running multiple instances of *MPCDSim* in parallel, and thus improving the statistics of the data gathered, by providing two distinct realizations of the simulated system, each contributing a simulation-time of t.

Furthermore, *MPCDSim* currently utilizes only one CPU thread, since the performance bottleneck lies with the GPU and in the communication with it; it should be relatively easy, however, to implement CPU-multithreading if needed.

4.2 Simulation Parameters

4.2.1 System Properties

Unless noted otherwise, the simulation parameters for results shown were as follows:

The primary system volume had dimensions $L_x = L_y = L_z = 30a$. The SRD rotation angle α was fixed at 2.27, corresponding to about 130°. The temperature was set to $k_B T = 1$, the MPC streaming time was $\Delta t_{\text{MPC}} = 0.1$. The statistical average number of MPC particles per collision cell was set to $\overline{N_c} = 10$.

[78] L. Torvalds et al. *git (Version 1.9.1)*. URL: `https://git-scm.com`.

[79] E. Westphal, S. Singh, C.-C. Huang, G. Gompper, and R. G. Winkler. "Multiparticle collision dynamics: GPU accelerated particle-based mesoscale hydrodynamic simulations". *Computer Physics Communications* 185 (2014), 495.

4.2.2 System Initialization

When initializing the simulation, $N_{\text{MPC}} := \overline{N_c} L_x L_y L_z a^{-3}$ MPC particles are created in total. With N being the number of springs per MPC polymer, i.e. $N+1$ being the number of MPC particles per polymer, this populates the simulation volume with $N_{\text{Polymer}} := N_{\text{MPC}} / (N+1)$ MPC polymers; the simulation parameters are chosen such that $N_{\text{Polymer}} \in \mathbb{N}_+$.

For each MPC polymer, the position $\mathbf{r}_1$ of the first MPC particle is sampled randomly from a uniform distribution $U([0, L_x) \times [0, L_y) \times [0, L_z))$ over the primary simulation volume. For each subsequent MPC particle $i+1$ of that same MPC polymer, the initial position $\mathbf{r}_{i+1}$ is set to be $\mathbf{r}_i + s\mathbf{X}$, where $\mathbf{X} \sim U\left([0,1)^3\right)$, and s is a scaling factor that is described below. The thusly chosen $\mathbf{r}_i$ is rejected if it lies outside of the primary simulation volume. If this would not be done, the newly generated particle would be mapped onto a potentially very distant location due to the boundary conditions employed, which could create unphysically highly stretched polymer bonds. On the other hand, this rejection mechanism is cause for a relatively depleted zone around the borders of the primary simulation volume. However, this imbalance in the mass distribution is expected to average out after the warmup phase described below.

The scaling parameter s is chosen to be the root-mean-square bond length l in the case for dimers, since there, this quantity can be readily obtained from the spring constant and vice versa. For larger polymers, calculation of the l is more involved (see chapter 6); for spring constants on the order of unity, it is expected that setting $s = 1$ for simplicity's sake still allows for the springs to reach their equilibrium state during simulation warmup.

The initial velocities of the MPC particles are chosen randomly and independently, with each Cartesian component being drawn from a normal distribution with zero mean and unity variance. This immediately yields the Maxwell-Boltzmann distribution for the magnitudes of the velocities, which is the correct distribution in systems where there is no external force; for other systems, the warmup time defined below has to be chosen such that a steady state can develop during warumup. The magnitudes of the velocities, which depend on the fluid's temperature, are scaled from their initialization value to the physically correct one by the thermostat applied (see section 3.1.4).

After the system has been set up by choosing the initial positions and velocities of all MPC particles, a number of the usual MPC simulation steps is performed in what is called the *warmup phase* of the simulation. The goal of these steps is to let the system evolve to a point where the artifacts introduced by the initialization algorithm described above have averaged out and a steady state is reached, so that measurements of quantities one is interested in can be expected to accurately represent the physical behavior of an ideally-prepared system. Consequently, all measurements are performed only after the warmup phase has concluded.

4.3 Verification of Implementation Correctness

In this section, *MPCDSim* is tested in some scenarios where published results and theoretical predictions are available, in order to test the implementation for deficiencies. Of course, the absence of observed deviations is not a proof of the correctness of *MPCDSim*, but is reassuring nevertheless. Also, the tests shown are of diagnostic help in situations where indeed faults are introduced into the code during development.

4.3.1 Shear Flow of a Simple Fluid

As a first test, a simulation of a simple fluid in shear flow is presented in figure 4.1. The simulation data was taken by dividing the primary simulation volume along the y axis into intervals of length $\frac{a}{10}$, and for each of these subvolumes, averaging over all MPC particle velocities for about $3 \cdot 10^5$ MPC time-steps.

The results are in excellent agreement with the theoretical prediction, which is that in the central layer $y = \frac{1}{2} L_y$ of the simulation volume, the shear flow is zero, while its magnitude increases linearly with increasing distance from that central layer. Mathematically, this can be formulated as

$$v_x = - \left(y - \frac{1}{2} L_y + \frac{1}{2} \frac{a}{10} \right) \dot{\gamma} \, . \tag{4.3.1}$$

Since in the sampled subvolume identified by the coordinate y_0, one averages over all MPC particles in the interval $y \in \left[y_0, y_0 + \frac{a}{10} \right)$, one effectively samples from the average MPC particle position $y_0 + \frac{1}{2} \frac{a}{10}$; hence the corresponding correctional term $\frac{1}{2} \frac{a}{10}$ in equation (4.3.1).

4.3.2 Grid Shift

In order to see that the results in section 4.3.1 depend on the MPC particle decorrelation achieved by the grid shift procedure as described in section 3.1.2, figure 4.2 shows the same situation as figure 4.1, except that the grid shift has been disabled entirely in one case, and reduced in its effect in another case. The almost-vertical sections of the data graph coincide with the boundaries of the collision cells. The zig-zag pattern is due to the fact that within each collision cell, the average particle velocities are nearly constant. Since the collision cell boundaries are fixed without the grid shift, “communication” between different collision cells is achieved only through MPC particles crossing cell boundaries, which is a comparatively rare event; relaxation into the theoretically predicted equilibrium state would take vast amounts of computing time. However, with grid shift enabled, the interaction partners for one given MPC particle change frequently, so that equilibrium can be reached quickly.

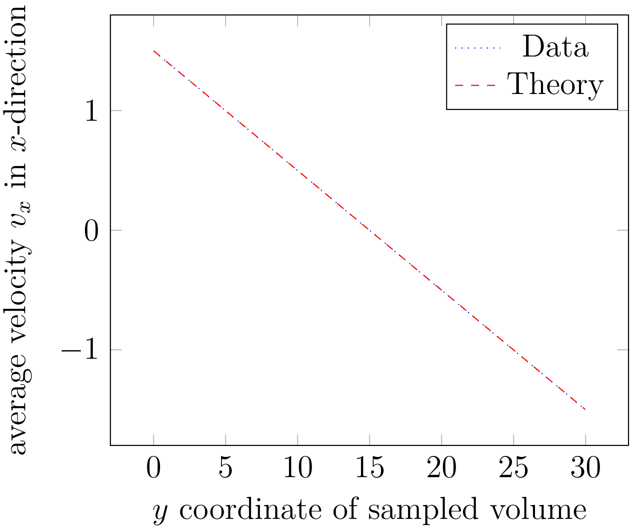

Figure 4.1: Results of a simulation of a simple fluid in shear flow with shear rate $\dot{\gamma} = 0.1$, showing how the x-component v_x of the fluid velocity depends on the y coordinate. About $3 \cdot 10^5$ time-steps were simulated.

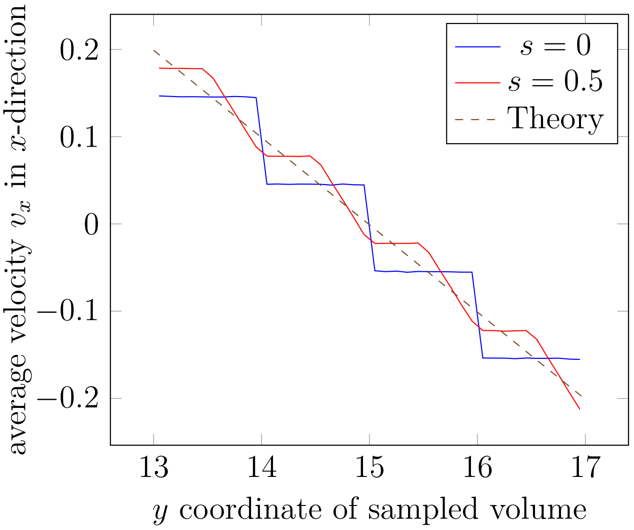

Figure 4.2: Results of simulations with varying "strength factors" s for the grid shift. This scaling factor s has the effect that, instead of sampling the grid shift lengths from $U\left(\left[-\frac{a}{2}, \frac{a}{2}\right]\right)$, the distribution function is given by $U\left(\left[-s\frac{a}{2}, s\frac{a}{2}\right]\right)$. The data for $s = 1$ (i.e. full grid shift) is not shown, as it is virtually indistinguishable from the theory curve. $s = 0$ corresponds to having no grid shift at all, and $s = 0.5$ is an in-between case that only serves to illustrate how the simulation reaches equilibrium faster with s approaching 1 from below. The imposed shear rate in this figure is $\dot{\gamma} = 0.1$, and about $3 \cdot 10^5$ time-steps were simulated.

Chapter 5

Center-of-Mass Velocity Autocorrelation Function

5.1 Theory

5.1.1 Laplace-Domain Equation of Motion

The linearized Navier-Stokes equation for linearly viscoelastic fluids (2.2.7) reads, in the absence of external forces ($\mathbf{f}^{\text{ext}} = 0$),

$$\varrho \partial_t v_i \left(\mathbf{r}, t\right) = -\partial_i p \left(\mathbf{r}, t\right) + \int\limits_{t'=-\infty}^{t} G\left(t - t'\right) \partial_j \partial_j v_i \left(\mathbf{r}, t'\right) \, \mathrm{d}t' \, . \tag{5.1.1}$$

In order to make the mathematical treatment easier later on, let the lower integration bound be changed to 0:[80]

$$\varrho \partial_t v_i \left(\mathbf{r}, t\right) = -\partial_i p \left(\mathbf{r}, t\right) + \int\limits_{t'=0}^{t} G\left(t - t'\right) \partial_j \partial_j v_i \left(\mathbf{r}, t'\right) \, \mathrm{d}t' \tag{5.1.2}$$

This change corresponds to a system which is in equilibrium at times $t \leq 0$. For other systems, this altered equation is only an approximation, the quality of which depends on t and on how fast $G\left(\Delta t\right)$ decays with Δt. However, arbitrarily small errors can be guaranteed with

[80] J. Farago, H. Meyer, J. Baschnagel, and A. N. Semenov. “Mode-coupling approach to polymer diffusion in an unentangled melt. II. The effect of viscoelastic hydrodynamic interactions”. *Physical Review E* 85 (2012), 051807. Equation (27).

sufficiently large t, since $G\left(\Delta t\right)$ tends to 0 monotonically with Δt (cf. section 2.2.2).

The spatial Fourier transform $\mathbf{r} \to \mathbf{k}$ then leads to (cf. appendix B.1)

$$\varrho \partial_t \tilde{v}_i \left(\mathbf{k}, t\right) = -\mathrm{i} k_i \tilde{p} \left(\mathbf{k}, t\right) - k_j k_j \int\limits_{t'=0}^{t} G\left(t - t'\right) \tilde{v}_i \left(\mathbf{k}, t'\right) \, \mathrm{d}t' \, , \tag{5.1.3}$$

where the tilde denotes a Fourier-transformed quantity (see appendix A).

Let $\tilde{\mathbf{v}}$ be uniquely decomposed into a longitudinal part $\tilde{\mathbf{v}}^{\mathrm{L}}$ parallel to $\mathbf{k}$ and a transverse part $\tilde{\mathbf{v}}^{\mathrm{T}}$ perpendicular to $\mathbf{k}$ for a fixed $\mathbf{k}$, i.e. $\tilde{\mathbf{v}} = \tilde{\mathbf{v}}^{\mathrm{L}} + \tilde{\mathbf{v}}^{\mathrm{T}}$. Then, one can perform a projection of the vectors occurring in (5.1.3) onto the subspace perpendicular to $\mathbf{k}$ and obtain

$$\varrho \partial_t \tilde{v}_i^{\mathrm{T}} \left(\mathbf{k}, t\right) = -k^2 \int\limits_{t'=0}^{t} G\left(t - t'\right) \tilde{v}_i^{\mathrm{T}} \left(\mathbf{k}, t'\right) \, \mathrm{d}t' \, . \tag{5.1.4}$$

Multiplication with the t-independent term $\tilde{v}_i^{\mathrm{T}} \left(-\mathbf{k}, 0\right)$ and summation over the repeated index i yields

$$\varrho \partial_t \tilde{v}_i^{\mathrm{T}} \left(\mathbf{k}, t\right) \tilde{v}_i^{\mathrm{T}} \left(-\mathbf{k}, 0\right) = -k^2 \int\limits_{t'=0}^{t} G\left(t - t'\right) \tilde{v}_i^{\mathrm{T}} \left(\mathbf{k}, t'\right) \tilde{v}_i^{\mathrm{T}} \left(-\mathbf{k}, 0\right) \, \mathrm{d}t' \, . \tag{5.1.5}$$

Defining the velocity autocorrelation function in the Fourier subspace perpendicular to $\mathbf{k}$ as

$$\tilde{C}_{\mathrm{v}}^{\mathrm{T}} \left(\mathbf{k}, t\right) := \left\langle \tilde{\mathbf{v}}^{\mathrm{T}} \left(\mathbf{k}, t\right) \cdot \tilde{\mathbf{v}}^{\mathrm{T}} \left(-\mathbf{k}, 0\right) \right\rangle \, , \tag{5.1.6}$$

where the angle brackets denote statistical averaging over the stochastic value of $\tilde{\mathbf{v}}^{\mathrm{T}} \left(\mathbf{k}, t\right) \cdot \tilde{\mathbf{v}}^{\mathrm{T}} \left(-\mathbf{k}, 0\right)$, one can calculate the ensemble average on both sides of equation (5.1.5) to obtain the equation governing the temporal evolution of $\tilde{C}_{\mathrm{v}}^{\mathrm{T}} \left(\mathbf{k}, t\right)$:

$$\varrho \partial_t \tilde{C}_{\mathrm{v}}^{\mathrm{T}} \left(\mathbf{k}, t\right) = -k^2 \int\limits_{t'=0}^{t} G\left(t - t'\right) \tilde{C}_{\mathrm{v}}^{\mathrm{T}} \left(\mathbf{k}, t'\right) \, \mathrm{d}t' \, . \tag{5.1.7}$$

Due to the lower integration bound being 0, this equation can conveniently be subjected to a Laplace transform $t \to s$ (denoted by the decoration of a symbol with a hat) to arrive at an algebraic equation for $\hat{\tilde{C}}_{\mathrm{v}}^{\mathrm{T}}$ (cf. appendices C.1 and C.2),

$$\varrho \left(s \hat{\tilde{C}}_{\mathrm{v}}^{\mathrm{T}} \left(\mathbf{k}, s\right) - \tilde{C}_{\mathrm{v}}^{\mathrm{T}} \left(\mathbf{k}, 0\right) \right) = -k^2 \hat{G} \left(s\right) \hat{\tilde{C}}_{\mathrm{v}}^{\mathrm{T}} \left(\mathbf{k}, s\right) \, , \tag{5.1.8}$$

which can be rearranged to yield

$$\hat{\tilde{C}}_{\mathrm{v}}^{\mathrm{T}}\left(\mathbf{k}, s\right) = \frac{\varrho \tilde{C}_{\mathrm{v}}^{\mathrm{T}}\left(\mathbf{k}, 0\right)}{\varrho s + k^2 \hat{G}\left(s\right)} \,. \tag{5.1.9}$$

5.1.2 Solution for $\tilde{C}_{\mathrm{v}}^{\mathrm{T}}$ for Simple Fluids

For simple, purely viscous fluids, equation (2.2.2) has to reduce to equation (2.1.45). This is achieved by

$$G\left(t - t'\right) = \eta \delta \left(t - t'\right) , \tag{5.1.10}$$

which means that the stress tensor depends only on the instantaneous value of the shear rate, but not its history. Then, the Laplace transform of G is given by[81]

$$\hat{G}\left(s\right) = \eta \,, \tag{5.1.11}$$

so that equation (5.1.9) simplifies to

$$\hat{\tilde{C}}_{\mathrm{v}}^{\mathrm{T}}\left(\mathbf{k}, s\right) = \frac{\tilde{C}_{\mathrm{v}}^{\mathrm{T}}\left(\mathbf{k}, 0\right)}{s + k^2 \eta \varrho^{-1}} \,, \tag{5.1.12}$$

the inverse Laplace transform $s \to t$ of which is[82]

$$\tilde{C}_{\mathrm{v}}^{\mathrm{T}}\left(\mathbf{k}, t\right) = \tilde{C}_{\mathrm{v}}^{\mathrm{T}}\left(\mathbf{k}, 0\right) \exp\left(-\frac{k^2 \eta}{\varrho} t\right) \,. \tag{5.1.13}$$

This exponential decay is characteristic of simple fluids and can be used to determine the kinematic viscosity $\nu = \eta / \varrho$ from measurements of $\tilde{C}_{\mathrm{v}}^{\mathrm{T}}\left(\mathbf{k}, t\right) / \tilde{C}_{\mathrm{v}}^{\mathrm{T}}\left(\mathbf{k}, 0\right)$.

[81] P. Dyke. *An Introduction to Laplace Transforms and Fourier Series.* Ed. by M. A. J. Chaplain, K. Erdmann, A. MacIntyre, E. Süli, M. R. Tehranchi, and J. F. Toland. 2nd ed. Springer Undergraduate Mathematics Series. Springer, 2014. Chapter 2.6.

[82] F. Oberhettinger and L. Badii. *Tables of Laplace Transforms.* Springer, 1973. Rule II.2.2.

5.1.3 Solution for $\tilde{C}_{\mathrm{v}}^{\mathrm{T}}$ for Rouse Polymers

5.1.3.1 Storage and Loss Moduli

The *Rouse model*[83,84] describes a linear chain of $N+1$ identical monomers, connected via N massless harmonic springs, dissolved in a viscous solvent; the hydrodynamic interaction between the monomers is, however, neglected. Likewise, interactions due to excluded volume effects, such as forbidding the penetration of one polymer's bonds by another bond, are not considered.

The Rouse model is mathematically simple enough to allow one to compute an analytical solution. The storage modulus, defined in equation (2.2.4), can be found to be[85,86]

$$G'(\omega) = \varphi k_B T \sum_{p=1}^{N} \frac{\omega^2 \tau_p^2}{1+\omega^2 \tau_p^2}\,, \tag{5.1.14}$$

while the loss modulus (2.2.5) is[87,88,89]

$$G''(\omega) = \omega\eta + \varphi k_B T \sum_{p=1}^{N} \frac{\omega \tau_p}{1+\omega^2 \tau_p^2}\,. \tag{5.1.15}$$

Here, η is the dynamic viscosity of the solvent, and φ is the number density of polymer molecules, i.e. the number of polymer chains per unit volume. The *relaxation times* τ_p are given by[90]

$$\tau_p = \sigma^2 \left(24 B k_B T \sin^2 \left(\frac{p\pi}{2(N+1)} \right) \right)^{-1} > 0\,, \tag{5.1.16}$$

with $\sigma^2 > 0$ being the average squared distance of two consecutive monomers. $B > 0$ is the mobility, defined as v/F of the initial drift velocity v of a monomer when a small force F acts

[83] P. E. Rouse. "A Theory of the Linear Viscoelastic Properties of Dilute Solutions of Coiling Polymers". *The Journal of Chemical Physics* 21 (1953), 1272.

[84] Doi and Edwards, *The Theory of Polymer Dynamics*, Chapter 4.1.

[85] Rouse, "A Theory of the Linear Viscoelastic Properties of Dilute Solutions of Coiling Polymers", Equation (30a).

[86] Y.-G. Tao, I. O. Götze, and G. Gompper. "Multiparticle collision dynamics modeling of viscoelastic fluids". *The Journal of Chemical Physics* 128 (2008), 144902. arXiv: `0802.2200 [cond-mat.soft]`. Equation (30).

[87] Rouse, "A Theory of the Linear Viscoelastic Properties of Dilute Solutions of Coiling Polymers", Equation (30b).

[88] R. B. Bird, C. F. Curtiss, R. C. Armstrong, and O. Hassager. *Kinetic Theory.* 2nd ed. Vol. 2. Dynamics of Polymeric Liquids. John Wiley & Sons, 1987, Equations (13.4-21) and (13.4-22).

[89] Tao, Götze, and Gompper, "Multiparticle collision dynamics modeling of viscoelastic fluids", Equation (31).

[90] Rouse, "A Theory of the Linear Viscoelastic Properties of Dilute Solutions of Coiling Polymers", Equation (31).

on it; B is related[91] to the friction constant ζ via

$$B = \zeta^{-1}\,. \tag{5.1.17}$$

5.1.3.2 Relaxation Modulus

In order to retrieve the relaxation modulus $G\left(t\right)$ from the complex modulus $G^{*}\left(\omega\right)$ (cf. (2.2.3)),

$$G^{*}\left(\omega\right) = \mathrm{i}\omega \int\limits_{t'=0}^{\infty} G\left(t'\right) \exp\left(-\mathrm{i}\omega t'\right)\, \mathrm{d}t'\,, \tag{5.1.18}$$

one can multiply both sides of this equation by $-\mathrm{i}\omega^{-1}\exp\left(\mathrm{i}\omega t\right)$ and integrate ω over $\mathbb{R}$:

$$\int\limits_{\omega=-\infty}^{\infty} \frac{G^{*}\left(\omega\right)\exp\left(\mathrm{i}\omega t\right)}{\mathrm{i}\omega}\,\mathrm{d}\omega = \int\limits_{\omega=-\infty}^{\infty}\int\limits_{t'=0}^{\infty} G\left(t'\right)\exp\left(\mathrm{i}\omega\left(t-t'\right)\right)\,\mathrm{d}t'\,\mathrm{d}\omega\,. \tag{5.1.19}$$

One can carry out the integration over ω on the right hand side, using the representation (B.2.1) of the Dirac delta function, to obtain

$$\begin{aligned} \int\limits_{\omega=-\infty}^{\infty} \frac{G^{*}\left(\omega\right)\exp\left(\mathrm{i}\omega t\right)}{\mathrm{i}\omega}\,\mathrm{d}\omega &= 2\pi \int\limits_{t'=0}^{\infty} G\left(t'\right)\delta\left(t-t'\right)\mathrm{d}t' \\ &= 2\pi G\left(t\right)\theta\left(t\right)\,. \end{aligned} \tag{5.1.20}$$

In order to simplify the left-hand side, one can use the definitions (2.2.4) and (2.2.5) of the storage and loss moduli,

$$G^{*}\left(\omega\right) = G'\left(\omega\right) + \mathrm{i}G''\left(\omega\right)\,, \tag{5.1.21}$$

and insert the expressions (5.1.14) and (5.1.15):

$$G^{*}\left(\omega\right) = \varphi k_B T \sum_{p=1}^{N} \frac{\omega^2\tau_p^2}{1+\omega^2\tau_p^2} + \mathrm{i}\omega\eta + \mathrm{i}\varphi k_B T \sum_{p=1}^{N} \frac{\omega\tau_p}{1+\omega^2\tau_p^2}\,. \tag{5.1.22}$$

[91] Doi and Edwards, *The Theory of Polymer Dynamics*, Equation (3.5).

Division by iω then yields

$$\frac{G^{*}\left(\omega\right)}{\mathrm{i}\omega}=\eta+\varphi k_{B}T\sum_{p=1}^{N}\frac{\tau_{p}-\mathrm{i}\omega\tau_{p}^{2}}{1+\omega^{2}\tau_{p}^{2}}\,. \tag{5.1.23}$$

Writing the denominator as $1+\omega^{2}\tau_{p}^{2}=\left(1-\mathrm{i}\omega\tau_{p}\right)\left(1+\mathrm{i}\omega\tau_{p}\right)$, one can cancel one of the factors with the numerator, arriving at

$$\frac{G^{*}\left(\omega\right)}{\mathrm{i}\omega}=\eta+\varphi k_{B}T\sum_{p=1}^{N}\frac{1}{\tau_{p}^{-1}+\mathrm{i}\omega}\,. \tag{5.1.24}$$

The left-hand side of (5.1.20) is (up to a factor of $\sqrt{2\pi}$) an inverse Fourier transformation, the result of which is[92] (see also appendix B.2)

$$\begin{aligned}\int\limits_{\omega=-\infty}^{\infty}\frac{G^{*}\left(\omega\right)\exp\left(\mathrm{i}\omega t\right)}{\mathrm{i}\omega}\,\mathrm{d}\omega&=\sqrt{2\pi}\mathcal{F}_{\omega}^{-1}\left\{\frac{G^{*}\left(\omega\right)}{\mathrm{i}\omega}\right\}\left(t\right)\\&=2\pi\eta\delta\left(t\right)+\varphi k_{B}T\sum_{p=1}^{N}2\pi\exp\left(-\frac{t}{\tau_{p}}\right)\theta\left(t\right)\,.\end{aligned} \tag{5.1.25}$$

Comparison with the final result of (5.1.20) yields

$$G\left(t\right)\theta\left(t\right)=\eta\delta\left(t\right)+\varphi k_{B}T\sum_{p=1}^{N}\exp\left(-\frac{t}{\tau_{p}}\right)\theta\left(t\right)\,. \tag{5.1.26}$$

Since values for $t<0$ do not enter in the Laplace transform

$$\hat{G}\left(s\right)=\int\limits_{t=0}^{\infty}G\left(t\right)\exp\left(-st\right)\,\mathrm{d}t\,, \tag{5.1.27}$$

the Heaviside function can be replaced by unity in equation (5.1.26). The Laplace transform is then computed to be[93, 94]

$$\hat{G}\left(s\right)=\eta+\varphi k_{B}T\sum_{p=1}^{N}\frac{1}{\tau_{p}^{-1}+s}\,. \tag{5.1.28}$$

[92] F. Oberhettinger. *Tables of Fourier Transforms and Fourier Transforms of Distributions.* Springer, 1990. Rule III.3.42.

[93] Dyke, *An Introduction to Laplace Transforms and Fourier Series*, Chapter 2.6.

[94] Oberhettinger and Badii, *Tables of Laplace Transforms*, Rule I.5.1.

Insertion of this result into (5.1.9) leads to the result

$$\hat{\tilde{C}}_{\mathrm{v}}^{\mathrm{T}}(\mathbf{k}, s)=\frac{\varrho \tilde{C}_{\mathrm{v}}^{\mathrm{T}}(\mathbf{k}, 0)}{\varrho s+k^2\left(\eta+\varphi k_B T \sum_{p=1}^N\left(\tau_p^{-1}+s\right)^{-1}\right)}\,. \tag{5.1.29}$$

5.1.3.3 Solution for $\tilde{C}_{\mathrm{v}}^{\mathrm{T}}(\mathbf{k}, t)$ for the Rouse Model

In order to find the analytic solution for $\tilde{C}_{\mathrm{v}}^{\mathrm{T}}(\mathbf{k}, t)$ within the Rouse model, one has to compute the inverse Laplace transform of $\hat{\tilde{C}}_{\mathrm{v}}^{\mathrm{T}}(\mathbf{k}, s)$ in (5.1.29). Let

$$D(s):=\varrho s+k^2\left(\eta+\varphi k_B T \sum_{n=1}^N\left(\tau_n^{-1}+s\right)^{-1}\right) \tag{5.1.30}$$

be the denominator in (5.1.29). Multiplication with

$$W(s):=\prod_{p=1}^N\left(\tau_p^{-1}+s\right)\,, \tag{5.1.31}$$

which is a polynomial of degree N in s, yields

$$\begin{aligned} P(s) & :=D(s) W(s) \\ & =\left(\varrho s+k^2 \eta\right) \prod_{p=1}^N\left(\tau_p^{-1}+s\right)+k^2 \varphi k_B T \sum_{n=1}^N \frac{1}{\tau_n^{-1}+s} \prod_{p=1}^N\left(\tau_p^{-1}+s\right) \\ & =\left(\varrho s+k^2 \eta\right) W(s)+k^2 \varphi k_B T \sum_{n=1}^N \prod_{\substack{p=1 \\ p \neq n}}^N\left(\tau_p^{-1}+s\right)\,. \end{aligned} \tag{5.1.32}$$

Since this expression is a polynomial of degree $N+1$ in s, the coefficient of s^{N+1} being $\varrho \neq 0$, it can be written as

$$P(s)=\varrho \prod_{m=1}^{N+1}\left(s-P_m\right)\,, \tag{5.1.33}$$

where P_n are the roots of the polynomial $\varrho^{-1} P(s)$. Similarly, since $W(s)$ is a polynomial in s, the coefficient of s^n being denoted by W_n, $W(s)$ can be written as

$$W(s)=\sum_{n=0}^N W_n s^n\,. \tag{5.1.34}$$

Multiplying both the numerator and the denominator in (5.1.29) by $W(s)$, one finds

$$\begin{aligned}\hat{\tilde{C}}_{\mathrm{v}}^{\mathrm{T}}(\mathbf{k},s) &= \varrho\tilde{C}_{\mathrm{v}}^{\mathrm{T}}(\mathbf{k},0)\,\frac{W(s)}{P(s)} \\ &= \tilde{C}_{\mathrm{v}}^{\mathrm{T}}(\mathbf{k},0)\sum_{n=0}^{N} W_n \frac{s^n}{\prod_{m=1}^{N+1}(s-P_m)}\end{aligned} \tag{5.1.35}$$

Assuming that all of the P_n are distinct, i.e. that $P(s)$ only has simple roots, the inverse Laplace transform $s \to t$ of

$$s^n \prod_{m=1}^{N+1}(s-P_m)^{-1} \tag{5.1.36}$$

can be computed to be[95]

$$\mathcal{L}_s^{-1}\left\{\frac{s^n}{\prod_{m=1}^{N+1}(s-P_m)}\right\}(t) = \sum_{m=1}^{N+1} P_m^n \exp(P_m t) \prod_{\substack{q=1\\ q\neq m}}^{N+1}(P_m-P_q)^{-1}\ , \tag{5.1.37}$$

where the property $n \leq N$ was used. Thus, the velocity autocorrelation function in the Fourier subspace perpendicular to $\mathbf{k}$ is found to be

$$\tilde{C}_{\mathrm{v}}^{\mathrm{T}}(\mathbf{k},t) = \tilde{C}_{\mathrm{v}}^{\mathrm{T}}(\mathbf{k},0)\sum_{n=0}^{N} W_n \sum_{m=1}^{N+1} P_m^n \exp(P_m t) \prod_{\substack{q=1\\ q\neq m}}^{N+1}(P_m-P_q)^{-1}\ . \tag{5.1.38}$$

To make the dependency of $\tilde{C}_{\mathrm{v}}^{\mathrm{T}}(\mathbf{k},t)$ on the relaxation times τ_p explicit, it is necessary to express W_n and P_n in terms of the τ_p. As is evident from (5.1.31), $W_N = 1$. The other coefficients W_{N-n}, $n \neq 0$, can be calculated using Viète's formula[96]

$$W_{N-n} = \sum_{1\leq i_1<i_2<\ldots<i_n\leq N} \tau_{i_1}^{-1}\tau_{i_2}^{-1}\cdots\tau_{i_n}^{-1} > 0\,. \tag{5.1.39}$$

One can define the functions

$$W^{(n)}(s) := \frac{1}{\tau_n^{-1}+s}W(s) = \prod_{\substack{p=1\\ p\neq n}}^{N}\left(\tau_p^{-1}+s\right)\ , \tag{5.1.40}$$

which again are polynomials in s, but of degree $N-1$. Similar as for $W(s)$, their leading

[95] Oberhettinger and Badii, *Tables of Laplace Transforms*, Rule II.2.78.

[96] E. B. Vinberg. *A Course in Algebra.* Ed. by W. Craig, N. Ivanov, S. G. Krantz, and D. Saltman. Vol. 56. Graduate Studies in Mathematics. American Mathematical Society, 2003, Chapter 3.2.

coefficients are $W_{N-1}^{(n)} = 1$, and, for $m \neq 1$,

$$W_{N-m}^{(n)} = \sum_{\substack{1 \leq i_1 < i_2 < \ldots < i_{m-1} \leq N \\ n \notin \{i_1, i_2, \ldots, i_{m-1}\}}} \tau_{i_1}^{-1} \tau_{i_2}^{-1} \cdots \tau_{i_{m-1}}^{-1} > 0 \,. \tag{5.1.41}$$

With this in mind, one can rewrite (5.1.32) as

$$\varrho^{-1} P(s) = \left(s + \varrho^{-1} k^2 \eta\right) W(s) + \varrho^{-1} k^2 \varphi k_B T \sum_{n=1}^{N} W^{(n)}(s) \;. \tag{5.1.42}$$

Representing the polynomial $\varrho^{-1} P(s)$ by its coefficients Q_n,

$$\varrho^{-1} P(s) = \sum_{n=0}^{N+1} Q_n s^n \tag{5.1.43}$$

and comparing this with (5.1.42), one finds that

$$\begin{aligned} Q_{N+1} &= W_N = 1 \,, \\ Q_N &= W_{N-1} + \varrho^{-1} k^2 \eta W_N \,, \\ \forall n \in [1, N-1] : \quad Q_n &= W_{n-1} + \varrho^{-1} k^2 \eta W_n + \varrho^{-1} k^2 \varphi k_B T \sum_{m=1}^{N} W_n^{(m)} \,, \\ Q_0 &= \varrho^{-1} k^2 \eta W_0 + \varrho^{-1} k^2 \varphi k_B T \sum_{m=1}^{N} W_0^{(m)} \,. \end{aligned} \tag{5.1.44}$$

Since $\varrho > 0$, $\varphi > 0$, $k^2 > 0$, $\eta > 0$, and $k_B T > 0$, one finds that the Q_n all are real and positive.

5.1.3.4 Limit of Long Polymers

In the limit of large $N \gg 1$ and small $p \ll N$, one can approximate $N + 1 \approx N$ and expand the sine in (5.1.16) into its Taylor series, keeping only the linear term, to obtain

$$\tau_p \approx \frac{\zeta N^2 \sigma^2}{6 \pi^2 k_B T p^2} \,. \tag{5.1.45}$$

Evidently, within the approximation (5.1.45), $\tau_p = \tau_1 / p^2$ holds. From this, it can be seen that the relaxation times are ordered $\tau_1 > \tau_2 > \tau_3 > \ldots$, and that therefore, the longest relaxation times provide the major contributions to the storage and loss moduli.

5.1.3.5 Dimers

In the case of dimers, where $N = 1$, the only emerging relaxation time can be computed from (5.1.16) to be

$$\tau_1 = \frac{\sigma^2}{12Bk_BT}\,. \tag{5.1.46}$$

If the potential between two neighboring monomers, being a distance $\mathbf{R}$ apart from each other, is written as $K\mathbf{R}^2$ (K being the spring constant), then σ^2 is given by[97,98]

$$\sigma^2 = \frac{3k_BT}{2K}\,, \tag{5.1.47}$$

so that

$$\tau_1 = \frac{\zeta}{8K}\,. \tag{5.1.48}$$

For the storage modulus (5.1.14) and the loss modulus (5.1.15), one thus obtains

$$G'(\omega) = \varphi k_BT\frac{\omega^2\tau_1^2}{1+\omega^2\tau_1^2}\,, \tag{5.1.49}$$

$$G''(\omega) = \omega\eta + \varphi k_BT\frac{\omega\tau_1}{1+\omega^2\tau_1^2}\,, \tag{5.1.50}$$

and consequently, the Laplace-transformed relaxation modulus (5.1.28) is given by

$$\hat{G}(s) = \eta + \varphi k_BT\frac{1}{\tau_1^{-1}+s}\,. \tag{5.1.51}$$

The polynomial $P(s)$ defined in (5.1.32) now reads

$$\begin{aligned} P(s) &= \left(\varrho s + k^2\eta\right)\left(\tau_1^{-1}+s\right) + k^2\varphi k_BT \\ &= \varrho\left(s^2 + s\left(\tau_1^{-1}+k^2\varrho^{-1}\eta\right) + \left(k^2\varrho^{-1}\eta\tau_1^{-1} + k^2\varrho^{-1}\varphi k_BT\right)\right)\,, \end{aligned} \tag{5.1.52}$$

and hence its two roots P_1 and P_2 are (cf. equation (5.1.33))

$$P_n = -\frac{\tau_1^{-1}+k^2\varrho^{-1}\eta}{2} + (-1)^n\sqrt{\frac{\left(\tau_1^{-1}+k^2\varrho^{-1}\eta\right)^2}{4} - k^2\varrho^{-1}\left(\eta\tau_1^{-1}+\varphi k_BT\right)}\,. \tag{5.1.53}$$

The solution (5.1.38) for the velocity autocorrelation function in the Fourier subspace per-

[97] Doi and Edwards, *The Theory of Polymer Dynamics*, Equation (4.5).

[98] Kowalik and Winkler, "Multiparticle collision dynamics simulations of viscoelastic fluids: Shear-thinning Gaussian dumbbells", Equation (3) and section III.A.

pendicular to $\mathbf{k}$ can, with $W_0 = \tau_1^{-1}$ and $W_1 = 1$, thus be re-written as

$$\tilde{C}_{\mathrm{v}}^{\mathrm{T}}(\mathbf{k},t)\,\frac{P_1 - P_2}{\tilde{C}_{\mathrm{v}}^{\mathrm{T}}(\mathbf{k},0)} = \exp(P_1 t)\left(\tau_1^{-1} + P_1\right) - \exp(P_2 t)\left(\tau_1^{-1} + P_2\right)\ . \tag{5.1.54}$$

Defining

$$\begin{aligned} p_A &:= -\frac{\tau_1^{-1} + k^2\varrho^{-1}\eta}{2} \\ p_B &:= -\mathrm{i}\sqrt{\frac{\left(\tau_1^{-1} + k^2\varrho^{-1}\eta\right)^2}{4} - k^2\varrho^{-1}\left(\eta\tau_1^{-1} + \varphi k_B T\right)}\ , \end{aligned} \tag{5.1.55}$$

the roots P_1 and P_2 in equation (5.1.53) can be written more compactly as

$$\begin{aligned} P_1 &= p_A - \mathrm{i}p_B \\ P_2 &= p_A + \mathrm{i}p_B \end{aligned} \tag{5.1.56}$$

so that equation (5.1.54) can be cast into the form

$$\begin{aligned} \frac{\tilde{C}_{\mathrm{v}}^{\mathrm{T}}(\mathbf{k},t)}{\tilde{C}_{\mathrm{v}}^{\mathrm{T}}(\mathbf{k},0)} &= \exp(p_A t)\,\frac{\exp(-\mathrm{i}p_B t)\left(\tau_1^{-1} + p_A - \mathrm{i}p_B\right) - \exp(\mathrm{i}p_B t)\left(\tau_1^{-1} + p_A + \mathrm{i}p_B\right)}{-2\mathrm{i}p_B} \\ &= \frac{\exp(p_A t)}{2\mathrm{i}p_B}\left(\exp(\mathrm{i}p_B t)\left(\tau_1^{-1} + p_A + \mathrm{i}p_B\right) - \exp(-\mathrm{i}p_B t)\left(\tau_1^{-1} + p_A - \mathrm{i}p_B\right)\right) \\ &= \frac{\exp(p_A t)}{p_B}\left(\left(\tau_1^{-1} + p_A\right)\sin(p_B t) + p_B\cos(p_B t)\right)\ . \end{aligned} \tag{5.1.57}$$

Evidently, $p_A \in \mathbb{R}_-$. If

$$\frac{\left(\tau_1^{-1} + k^2\varrho^{-1}\eta\right)^2}{4} < k^2\varrho^{-1}\left(\eta\tau_1^{-1} + \varphi k_B T\right)\ , \tag{5.1.58}$$

then $p_B \in \mathbb{R}_+$, and equation (5.1.57) contains real quantities only.

Otherwise, if $\left(\tau_1^{-1} + k^2\varrho^{-1}\eta\right)^2 > 4k^2\varrho^{-1}\left(\eta\tau_1^{-1} + \varphi k_B T\right)$, p_B is purely imaginary. Then, let $p_C := \mathrm{i}p_B \in \mathbb{R}_+$, so that equation (5.1.57) can be written in terms of hyperbolic functions:

$$\frac{\tilde{C}_{\mathrm{v}}^{\mathrm{T}}(\mathbf{k},t)}{\tilde{C}_{\mathrm{v}}^{\mathrm{T}}(\mathbf{k},0)} = \frac{\exp(p_A t)}{p_C}\left(\left(\tau_1^{-1} + p_A\right)\sinh(p_C t) + p_C\cosh(p_C t)\right)\ . \tag{5.1.59}$$

Asymptotically, for constant $\mathbf{k}$ and $t \to \infty$, one thus has the proportionality

$$\tilde{C}_{\mathrm{v}}^{\mathrm{T}}(\mathbf{k},t) \sim \exp\left((p_A + p_C)\,t\right) \tag{5.1.60}$$

which tends to 0 if and only if $-p_A > p_C$, i.e. in the case

$$\frac{\tau_1^{-1} + k^2 \varrho^{-1} \eta}{2} > \sqrt{\frac{\left(\tau_1^{-1} + k^2 \varrho^{-1} \eta\right)^2}{4} - k^2 \varrho^{-1} \left(\eta \tau_1^{-1} + \varphi k_B T\right)} \tag{5.1.61}$$

which evidently is a tautology as long as $k \neq 0$.

In this large-t limit (5.1.60) of $\tilde{C}_{\mathrm{v}}^{\mathrm{T}}(\mathbf{k}, t)$, it is clear that it is the small k-values that provide the dominant contributions. A Taylor expansion of p_C around $k = 0$ up to order k^2 yields

$$p_C \approx \frac{1}{2\tau_1} - k^2 \varrho^{-1} \left(\varphi k_B T \tau_1 + \frac{\eta}{2}\right) , \tag{5.1.62}$$

so that, for large t,

$$C_{\mathrm{v}}^{\mathrm{T}}(\mathbf{k}, t) \sim \exp\left(-k^2 \varrho^{-1} \left(\eta + \varphi k_B T \tau_1\right) t\right) . \tag{5.1.63}$$

Comparison with the case of simple fluids, i.e. equation (5.1.13), shows that in the large-t limit, the viscous behavior dominates over the elastic contributions; the latter only influence the apparent viscosity, providing an additional term $\varphi k_B T \tau_1$.

5.2 Simulation Results

5.2.1 Simple Fluid

Figures 5.1 and 5.2 show data obtained from a simulation of a simple fluid, along with the theoretical prediction in equation (5.1.13). For the latter, the kinematic viscosity $\nu = \eta/\varrho$ has been computed to be $\nu = 0.8705$ from the MPC particle mass $m = 1$, the system temperature $k_B T = 1$, the MPC time-step $\Delta t_{\mathrm{MPC}} = 0.1$, and the SRD rotation angle $\alpha = 2.27$, via[99]

$$\begin{aligned} \nu = &\left(\frac{5m}{\left(m - 1 + \exp\left(-m\right)\right)\left(2 - \cos\left(\alpha\right) - \cos\left(2\alpha\right)\right)} - 1\right) \frac{k_B T \Delta t_{\mathrm{MPC}}}{2m} \\ &+ \frac{m - 1 + \exp\left(-m\right)}{18m} \left(1 - \cos\left(\alpha\right)\right) \frac{a^2}{\Delta t_{\mathrm{MPC}}} \end{aligned} \tag{5.2.1}$$

For $t \leq 40$, the simulation data is in excellent agreement both with the theory curve and with the literature.[100] For larger times, the signal-to-noise ratio in the simulation measurements gets gradually worse, so that the sample size is probably too small for the statistical average

[99] Gompper, Ihle, Kroll, and Winkler, *Multi-Particle Collision Dynamics: A Particle-Based Mesoscale Simulation Approach to the Hydrodynamics of Complex Fluids*, Table 1.

[100] Huang, Gompper, and Winkler, "Hydrodynamic correlations in multiparticle collision dynamics fluids", Fig. 2.

to approach its real mean value. This would serve not only as an explanation for the deviation of the data from the theoretical prediction, but is also hinted at by the fact that the data for $\mathbf{k}$ along different axes of the Cartesian coordinate system diverge for larger t, even though the simulation does not distinguish between the different directions.

5.2.2 Dimers

Figures 5.3 to 5.5 show the absolute value of $\tilde{C}_{\mathrm{v}}^{\mathrm{T}}(\mathbf{k},t)/\tilde{C}_{\mathrm{v}}^{\mathrm{T}}(\mathbf{k},0)$ as a function of t in dimeric fluid simulations.

Comparison of the data with the theoretical prediction (5.1.57) again shows very good agreement for $t < t_0$ for some t_0, where t_0 is getting smaller with increasing k. The relaxation time $\tau_1 \approx 13.4$ is measured from the decay of the correlation of the dimer's normal mode coordinates.[101] Once thusly obtained, it can be used in equation (5.1.57) to show that not only does the predicted envelope match the data, but also that the oscillation frequency and amplitude, into which τ_1 enters, matches the simulation.

5.2.3 Decamers

Figures 5.6 to 5.8 show simulation results for a system consisting of decamers, with varying spring constants K.

Figure 5.9 can be used to judge the quality of the prediction that is equation (5.1.38); again, for small enough t, the agreement of the qualitative features is very satisfactory. The first relaxation time τ_1 has been obtained by fitting equation (5.1.38) to the data, so that the agreement on the quantitative level is not verified independently.

5.3 Numerical Results for $\tilde{C}_{\mathrm{v}}^{\mathrm{T}}$

In order to study the properties of the theoretical solution (5.1.38) in general and the long-time limit in particular, one can calculate the solutions numerically for various parameters and compare them to one another.

In figure 5.10, the effect of varying the spring count N for a fixed first relaxation time τ_1 is shown. One notable property is that the slope of the envelope is not a monotonic function of the spring count. Another is that, as predicted by (5.1.38), the curve $N = 5$ demonstrates, after

[101] I. Teraoka. *Polymer Solutions. An Introduction to Physical Properties.* John Wiley & Sons, 2002, Chapter 3.4.2.

sufficiently long time, that one oscillation frequency is suppressed at a time. The frequency that persists the longest is the one related to the longest relaxation time τ_1.

Figures 5.11 and 5.12 show, for fixed N, the effect of varying τ_1. While in the case $N = 1$ the most noticeable change is the change of the slope of the envelope, the case $N = 10$ offers a richer spectrum of behaviors. It is interesting to note that, contrary to the long-time asymptote (5.1.63) for $N = 1$, the slope does not always steepen with increasing τ_1.

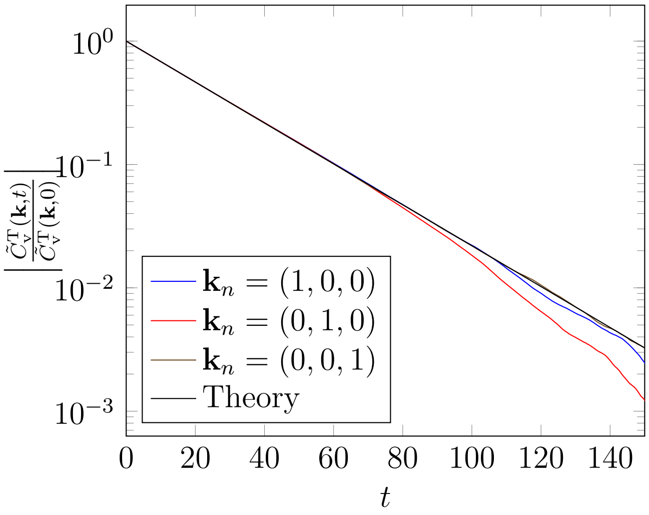

Figure 5.1: Simulation results for simple fluids without shear. The horizontal axis represents the time parameter t, while the vertical axis shows the absolute value of $\tilde{C}_{\mathrm{v}}^{\mathrm{T}}(\mathbf{k},t)$ normalized by $\tilde{C}_{\mathrm{v}}^{\mathrm{T}}(\mathbf{k},0)$. The vector $\mathbf{k}$ is related to $\mathbf{k}_n$ by $k_i = 2\pi k_{n,i}/L_i$. The theory curve is obtained from equation (5.1.13), with the kinematic viscosity $\nu = \eta/\varrho = 0.8705$ and $|\mathbf{k}_n| = 1$. The simulation data is averaged over about $1.13 \cdot 10^{7}$ simulation steps.

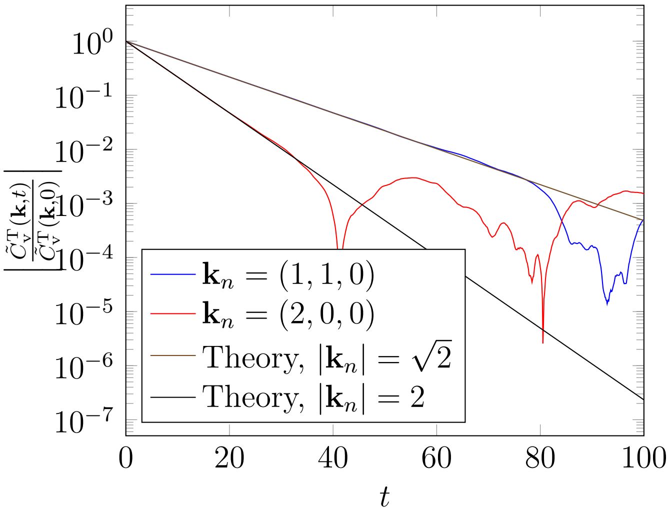

Figure 5.2: Depiction of simulation data and theory for the same kind of scenario as in figure 5.1, but with other values for $\mathbf{k}_n$, and thus k.

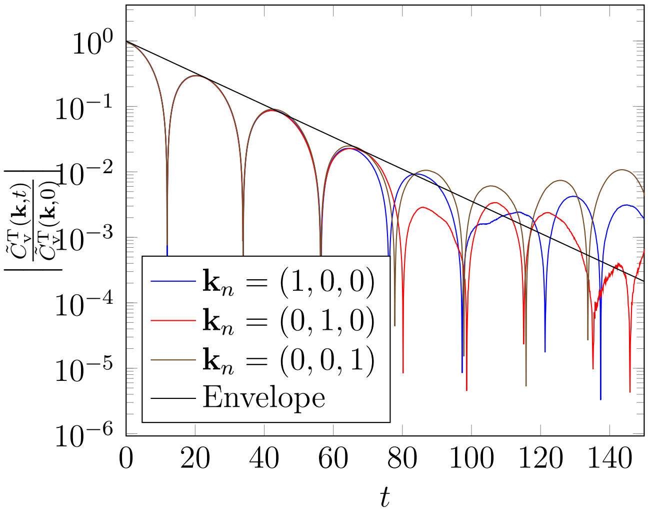

Figure 5.3: Simulation results for a dimer fluid without shear (spring count $N = 1$). The horizontal axis represents the time parameter t, while the vertical axis shows the absolute value of $\tilde{C}_{\mathrm{v}}^{\mathrm{T}}(\mathbf{k},t)$ normalized by $\tilde{C}_{\mathrm{v}}^{\mathrm{T}}(\mathbf{k},0)$. The vector $\mathbf{k}$ is related to $\mathbf{k}_n$ by $k_i = 2\pi k_{n,i}/L_i$. The theoretical envelope $\exp\left(-\left(\tau_1^{-1} + k^2\nu\right)t/2\right)$ is extracted from equations (5.1.57) and (5.1.55), with $|\mathbf{k}_n| = 1$, $\nu = \eta/\varrho = 0.8705$, and $\tau_1 \approx 13.4$. The root-mean-square bond length was chosen to be $l = 3$, which corresponds to spring constant $K = 1/3$. The data is sampled from about $3 \cdot 10^5$ simulation steps.

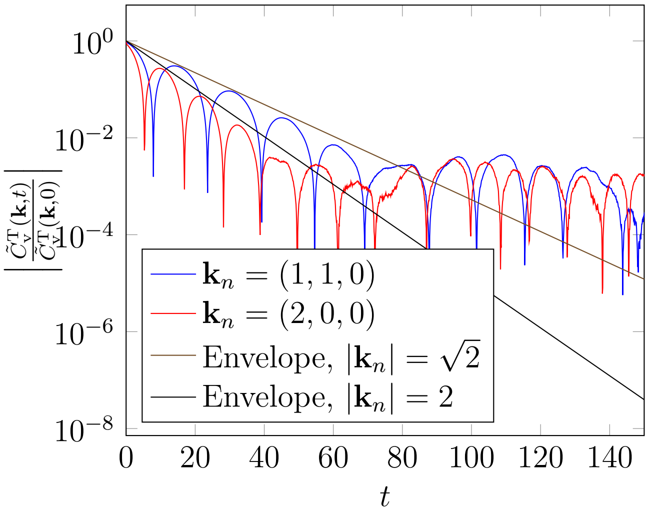

Figure 5.4: The same situation as in figure 5.4, but with other values of $\mathbf{k}_n$, and thus k.

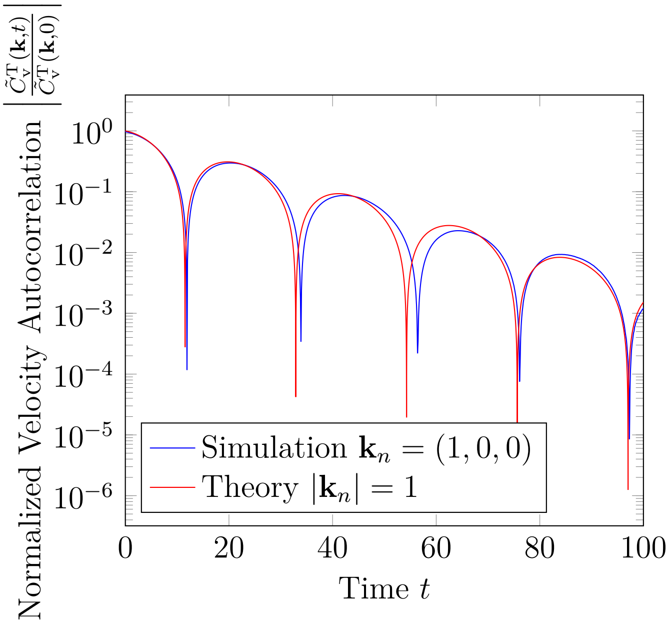

Figure 5.5: One of the three simulation curves from figure 5.4, with the addition of the theoretical prediction from equation (5.1.57).

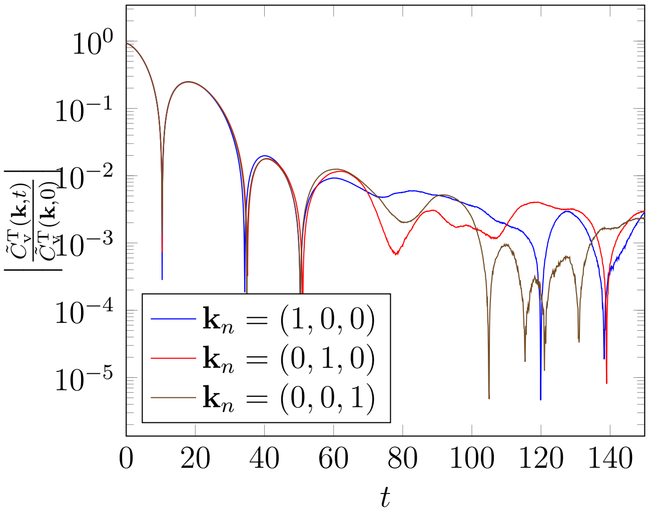

Figure 5.6: Simulation results for a decamer fluid (spring count $N = 9$) with spring constant $K = 1$. The horizontal axis represents the time parameter t, while the vertical axis shows the absolute value of $\tilde{C}_\mathrm{v}^\mathrm{T}(\mathbf{k},t)$ normalized by $\tilde{C}_\mathrm{v}^\mathrm{T}(\mathbf{k},0)$. The vector $\mathbf{k}$ is related to $\mathbf{k}_n$ by $k_i = 2\pi k_{n,i}/L_i$. The data is sampled from about $5 \cdot 10^6$ simulation steps.

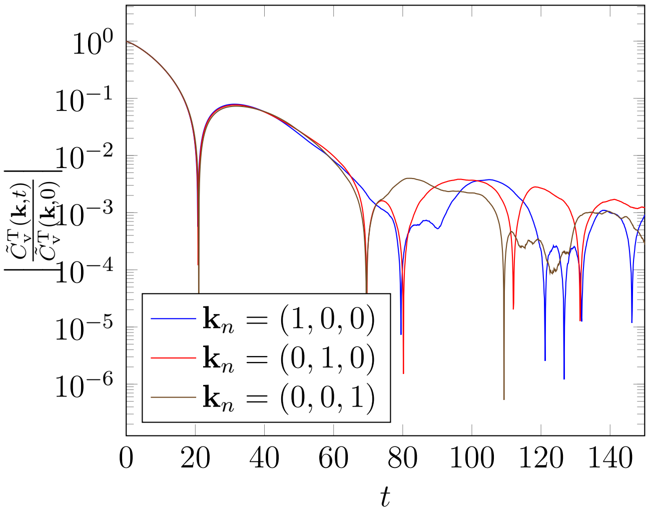

Figure 5.7: Simulation data for the same scenario as in figure 5.6, but with a spring constant $K = 5$.

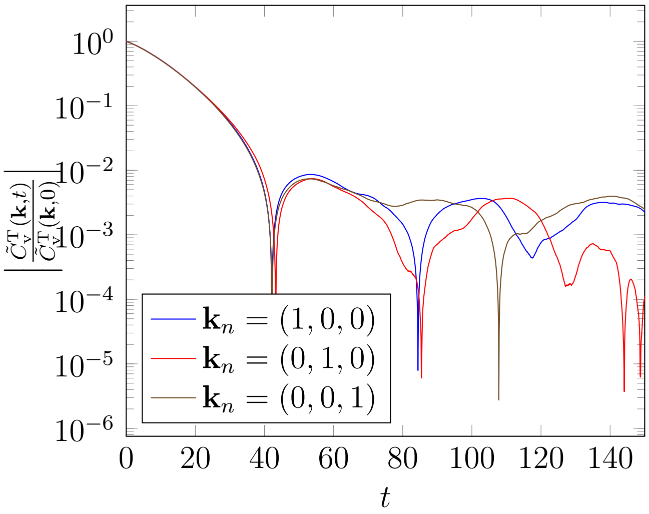

Figure 5.8: Simulation data for the same scenario as in figure 5.6, but with a spring constant $K = 10$.

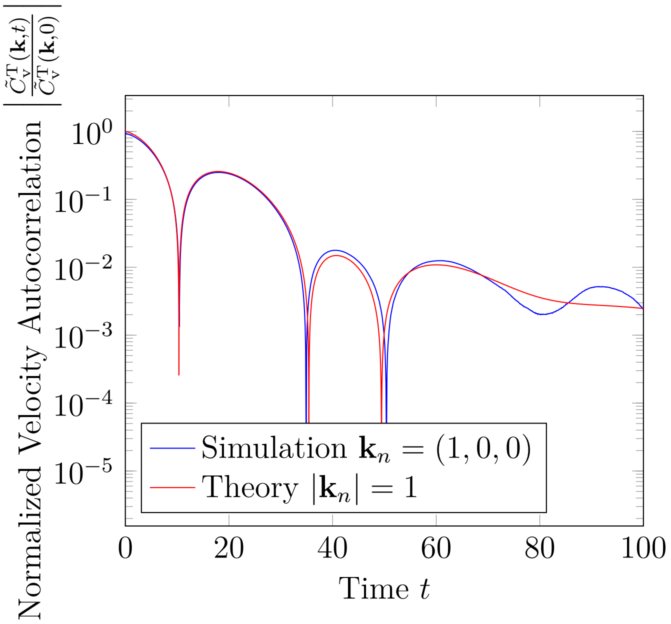

Figure 5.9: Simulation data for the same scenario as in figure 5.6, with $K = 1$, and the theoretical prediction (5.1.38).

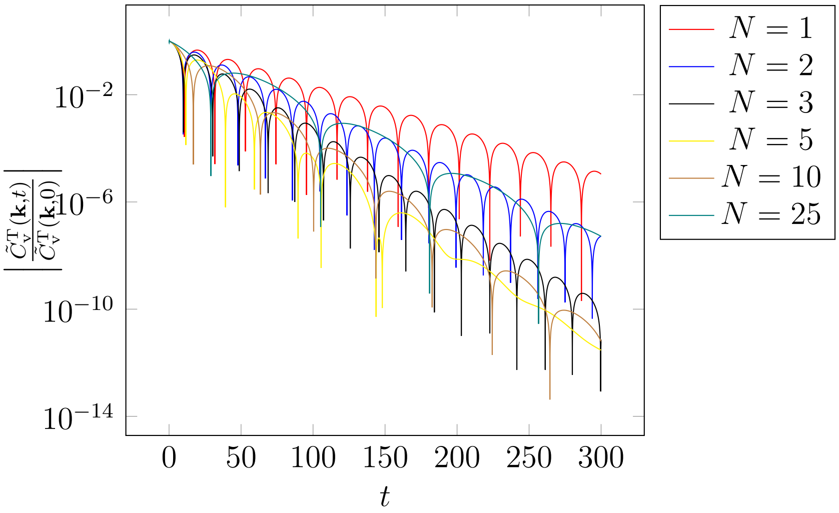

Figure 5.10: Numerical result for $\tilde{C}_{\mathrm{v}}^{\mathrm{T}}$ for various N with $\tau_1 = 26.8$ fixed.

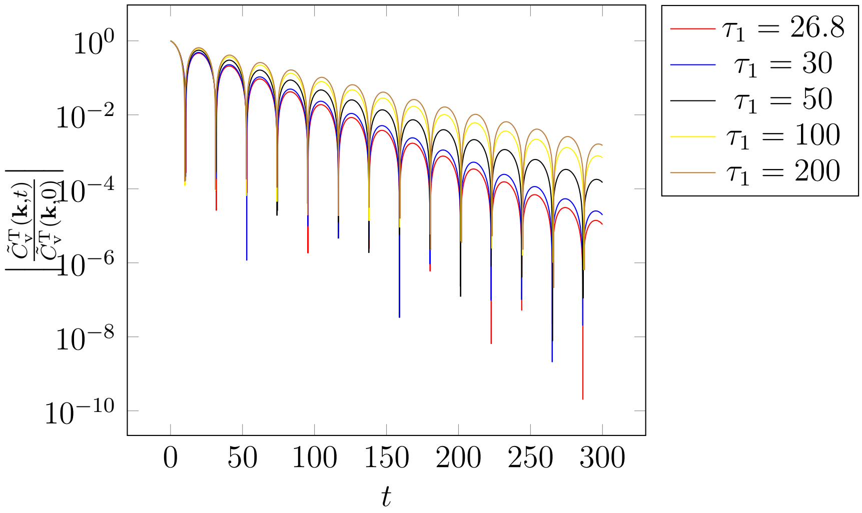

Figure 5.11: Numerical result for $\tilde{C}_{\mathrm{v}}^{\mathrm{T}}$ for various τ_1 with $N = 1$ fixed.

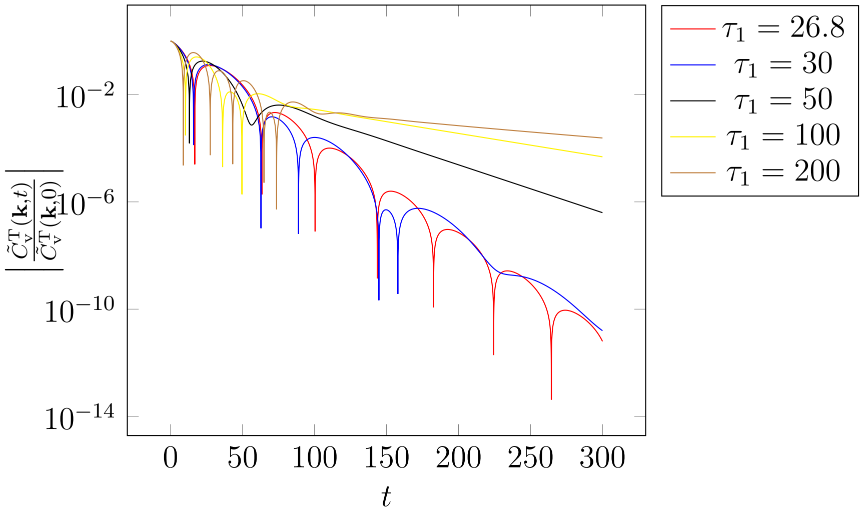

Figure 5.12: Numerical result for $\tilde{C}_{\mathrm{v}}^{\mathrm{T}}$ for various τ_1 with $N = 10$ fixed.

Chapter 6

Trimer MPC Fluid

While the behavior of dimers in MPC fluid has been studied already,[102] the next-harder problem, that of trimers, has not been treated yet.

With the goal of finding a relationship between the spring constant K and the root-mean-square bond length l, let $\mathbf{r}_i$, $i \in \{1, 2, 3\}$ denote the positions of MPC fluid particles that constitute a specific trimer, and let $\mathbf{R}_i := \mathbf{r}_{i+1} - \mathbf{r}_i$, $i \in \{1, 2\}$, be the bond vectors. The constituents are then interacting via the potential

$$U\left(\mathbf{R}_1, \mathbf{R}_2\right) := K\mathbf{R}_1^2 + K\mathbf{R}_2^2 \,. \tag{6.1}$$

In a system with hydrodynamic interactions, the equations of motion read[103]

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{r}_i = \sum_{j=1}^{3} \underline{\underline{H}}_{i,j}\left(\mathbf{r}_i - \mathbf{r}_j\right)\left(\mathbf{F}_i + \mathbf{\Gamma}_i\right) \,, \tag{6.2}$$

where $\underline{\underline{H}}_{i,j}\left(\mathbf{r}_i - \mathbf{r}_j\right)$ is the hydrodynamic tensor, $F_{i,\alpha} = -\frac{\partial U}{\partial R_{i,\alpha}}$ is the force exerted on particle i in the Cartesian direction $\alpha \in \{x, y, z\}$ due to the potential U, and $\Gamma_{i,\alpha}$ is a stochastic force acting on particle i along α, accounting for thermal fluctuations.

In order to simplify (6.2), Zimm's pre-averaging approximation[104, 105] is employed, where

[102] Kowalik and Winkler, "Multiparticle collision dynamics simulations of viscoelastic fluids: Shear-thinning Gaussian dumbbells".

[103] Kowalik and Winkler, "Multiparticle collision dynamics simulations of viscoelastic fluids: Shear-thinning Gaussian dumbbells", Equation (4).

[104] Doi and Edwards, *The Theory of Polymer Dynamics*, Equation (4.49).

[105] Kowalik and Winkler, "Multiparticle collision dynamics simulations of viscoelastic fluids: Shear-thinning Gaussian dumbbells", Equation (5).

$\underline{H}_{i,j}\left(\mathbf{r}_i-\mathbf{r}_j\right)$ is replaced with

$$\left\langle \underline{H}_{i,j}\left(\mathbf{r}_i-\mathbf{r}_j\right)\right\rangle = \underbrace{\left(\frac{\delta_{ij}}{6\pi\eta R_H}+\frac{1-\delta_{ij}}{\eta l\sqrt{6\pi^3\left|i-j\right|}}\right)}_{=:H_{ij}}\underline{I}\,, \tag{6.3}$$

with $\underline{I}$ being the unit tensor and δ_{ij} being the Kronecker symbol. l is the root-mean-square length of each bond, η is the dynamic viscosity, and R_H is the (Stokes-Einstein) hydrodynamic radius.

Defining

$$\begin{aligned} A &:= \frac{1}{6\pi\eta R_H} \\ B &:= \frac{1}{\eta l\sqrt{6\pi^3}}\,, \end{aligned} \tag{6.4}$$

the equations of motion (6.2) read

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t}\mathbf{r}_1 &= A\left(\mathbf{F}_1+\mathbf{\Gamma}_1\right)+B\left(\mathbf{F}_2+\mathbf{\Gamma}_2\right)+\frac{B}{\sqrt{2}}\left(\mathbf{F}_3+\mathbf{\Gamma}_3\right) \\ \frac{\mathrm{d}}{\mathrm{d}t}\mathbf{r}_2 &= B\left(\mathbf{F}_1+\mathbf{\Gamma}_1\right)+A\left(\mathbf{F}_2+\mathbf{\Gamma}_2\right)+B\left(\mathbf{F}_3+\mathbf{\Gamma}_3\right) \\ \frac{\mathrm{d}}{\mathrm{d}t}\mathbf{r}_3 &= \frac{B}{\sqrt{2}}\left(\mathbf{F}_1+\mathbf{\Gamma}_1\right)+B\left(\mathbf{F}_2+\mathbf{\Gamma}_2\right)+A\left(\mathbf{F}_3+\mathbf{\Gamma}_3\right)\,. \end{aligned} \tag{6.5}$$

Using

$$\begin{aligned} \mathbf{F}_1 &= 2K\mathbf{R}_1 \\ \mathbf{F}_1 &= -2K\mathbf{R}_1+2K\mathbf{R}_2 \\ \mathbf{F}_3 &= -2K\mathbf{R}_2\,, \end{aligned} \tag{6.6}$$

one can then obtain the equations of motion for the bond vectors $\mathbf{R}_i=\mathbf{r}_{i+1}-\mathbf{r}_i$:

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{r}_1 = a\mathbf{r}_1+b\mathbf{r}_2+\mathbf{E}_1 \tag{6.7}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{r}_2 = b\mathbf{r}_1+a\mathbf{r}_2+\mathbf{E}_2\,, \tag{6.8}$$

with

$$a := 4K\left(B - A\right) = -\frac{4K}{\zeta} \tag{6.9}$$

$$b := K\left(2A + \left(\sqrt{2} - 4\right)B\right) \tag{6.10}$$

$$\mathbf{E}_1 := \left(A - B\right)\left(\mathbf{\Gamma}_2 - \mathbf{\Gamma}_1\right) + B\left(1 - \frac{1}{\sqrt{2}}\right)\mathbf{\Gamma}_3 \tag{6.11}$$

$$\mathbf{E}_2 := \left(A - B\right)\left(\mathbf{\Gamma}_3 - \mathbf{\Gamma}_2\right) - B\left(1 - \frac{1}{\sqrt{2}}\right)\mathbf{\Gamma}_1\,, \tag{6.12}$$

where $\zeta = \left(A - B\right)^{-1}$ is the effective friction coefficient.[106] Since $K > 0$ and $\zeta > 0$, one can deduce that $a < 0$.

In order to incorporate shear flow into the system, the equations (6.7) and (6.8) have to be extended with a term $\underline{G}\mathbf{R}_i$, so that they read

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{R}_1 = a\mathbf{R}_1 + b\mathbf{R}_2 + \mathbf{E}_1 + \underline{G}\mathbf{R}_1 \tag{6.13}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{R}_2 = b\mathbf{R}_1 + a\mathbf{R}_2 + \mathbf{E}_2 + \underline{G}\mathbf{R}_2\,. \tag{6.14}$$

Aligning the coordinate system in such a way that the shear flow direction corresponds to the x axis and the shear gradient corresponds to the y axis, the linear map $\underline{G} : \mathbb{R}^3 \to \mathbb{R}^3$ is written

$$\underline{G} = \begin{pmatrix} 0 & \dot{\gamma} & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}\,, \tag{6.15}$$

so that

$$\left(\underline{G}\mathbf{R}_i\right)_\alpha = \dot{\gamma} R_{i,y}\delta_{\alpha x}\,. \tag{6.16}$$

Equations (6.13) and (6.14) can be combined into the linear system

$$\underbrace{\begin{pmatrix} \dot{\mathbf{R}}_1\left(t\right) \\ \dot{\mathbf{R}}_2\left(t\right) \end{pmatrix}}_{=:\dot{\rho}(t)} = \underbrace{\begin{pmatrix} a + \underline{G} & b \\ b & a + \underline{G} \end{pmatrix}}_{=:M} \underbrace{\begin{pmatrix} \mathbf{R}_1\left(t\right) \\ \mathbf{R}_2\left(t\right) \end{pmatrix}}_{=:\rho(t)} + \underbrace{\begin{pmatrix} \mathbf{E}_1\left(t\right) \\ \mathbf{E}_2\left(t\right) \end{pmatrix}}_{=:\xi(t)}\,, \tag{6.17}$$

so that ρ, $\dot{\rho}$ and ξ are elements of $\mathcal{V}^2$ and $M : \mathcal{V}^2 \to \mathcal{V}^2$ is a linear map, where $\mathcal{V} := \mathbb{R}^3$. Expressions like $a + \underline{G}$ are to be understood as $a\underline{I} + \underline{G}$, where again $\underline{I}$ is the unit tensor of the

[106] Kowalik and Winkler, “Multiparticle collision dynamics simulations of viscoelastic fluids: Shear-thinning Gaussian dumbbells”, Equation (7).

appropriate type; $\underline{I}$ will be suppressed in such cases.

Defining, as usual, the exponential of a linear map L to be

$$\exp\left(L\right) = \sum_{k=0}^{\infty} \frac{L^k}{k!}\,, \tag{6.18}$$

one can verify that, with $t \in \mathbb{R}$,

$$\dot{\rho}\left(t\right) - M\rho\left(t\right) = \exp\left(tM\right) \frac{\mathrm{d}}{\mathrm{d}t}\left(\exp\left(-tM\right)\rho\left(t\right)\right)\ . \tag{6.19}$$

So, (6.17) can be written as

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\exp\left(-tM\right)\rho\left(t\right)\right) = \exp\left(-tM\right)\xi\left(t\right)\ , \tag{6.20}$$

the solution of which is

$$\rho\left(t\right) = \exp\left(\left(t-t_0\right)M\right)\rho\left(t_0\right) + \exp\left(tM\right)\int\limits_{s=t_0}^{t} \exp\left(-sM\right)\xi\left(s\right)\,\mathrm{d}s\,, \tag{6.21}$$

with $t_0 \in \mathbb{R}$ arbitrary but fixed.

Let $q \in \mathbb{R}$. Then, one can use the simple structure of

$$qM = \begin{pmatrix} q\left(a+\underline{G}\right) & qb \\ qb & q\left(a+\underline{G}\right) \end{pmatrix} \tag{6.22}$$

to find its exponential

$$\exp\left(qM\right):\ \mathcal{V}^2 \to \mathcal{V}^2, \quad qM \mapsto \exp\left(qM\right) = \exp\left(q\left(a+\underline{G}\right)\right)\begin{pmatrix} \cosh\left(qb\right) & \sinh\left(qb\right) \\ \sinh\left(qb\right) & \cosh\left(qb\right) \end{pmatrix}\ . \tag{6.23}$$

Similarly, the exponential of the linear map $q\left(a+\underline{G}\right)$ is

$$\exp\left(q\left(a+\underline{G}\right)\right):\ \mathbb{R}^3 \to \mathbb{R}^3, \quad q\left(a+\underline{G}\right) \mapsto \exp\left(q\left(a+\underline{G}\right)\right) = \exp\left(aq\right)\begin{pmatrix} 1 & q\dot{\gamma} & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}\ . \tag{6.24}$$

As such, for all $v \in \mathcal{V}$,

$$\lim_{aq\to\infty}\left(\exp\left(qM\right)v\right) = 0\,. \tag{6.25}$$

Thus, and because $a < 0$, the choice $t_0 = -\infty$ reduces equation (6.21) to

$$\rho\left(t\right) = \exp\left(tM\right) \int\limits_{s=-\infty}^{t} \exp\left(-sM\right) \xi\left(s\right) \,\mathrm{d}s\,, \tag{6.26}$$

which, using the addition theorems for hyperbolic functions, can be written in terms of the $\mathbf{R}_i$ as

$$\begin{pmatrix} \mathbf{R}_1\left(t\right) \\ \mathbf{R}_2\left(t\right) \end{pmatrix} = \\ \int\limits_{s=-\infty}^{t} \exp\left(\left(t-s\right)\left(a+\underline{G}\right)\right) \begin{pmatrix} \cosh\left(b\left(t-s\right)\right) & \sinh\left(b\left(t-s\right)\right) \\ \sinh\left(b\left(t-s\right)\right) & \cosh\left(b\left(t-s\right)\right) \end{pmatrix} \begin{pmatrix} \mathbf{E}_1\left(t\right) \\ \mathbf{E}_2\left(t\right) \end{pmatrix} \mathrm{d}s\,. \tag{6.27}$$

Using (6.24), one can calculate the components of the bond vectors to be

$$\begin{aligned} R_{1,x}\left(t\right) = \int\limits_{s=-\infty}^{t} \Big(& \cosh\left(\left(t-s\right)b\right)\left(E_{1,x}\left(s\right) + \dot{\gamma}\left(t-s\right)E_{1,y}\left(s\right)\right) \\ & + \sinh\left(\left(t-s\right)b\right)\left(E_{2,x}\left(s\right) + \dot{\gamma}\left(t-s\right)E_{2,y}\left(s\right)\right) \\ \Big) & \cdot \exp\left(a\left(t-s\right)\right)\,\mathrm{d}s \end{aligned} \tag{6.28}$$

$$\begin{aligned} R_{2,x}\left(t\right) = \int\limits_{s=-\infty}^{t} \Big(& \sinh\left(\left(t-s\right)b\right)\left(E_{1,x}\left(s\right) + \dot{\gamma}\left(t-s\right)E_{1,y}\left(s\right)\right) \\ & + \cosh\left(\left(t-s\right)b\right)\left(E_{2,x}\left(s\right) + \dot{\gamma}\left(t-s\right)E_{2,y}\left(s\right)\right) \\ \Big) & \cdot \exp\left(a\left(t-s\right)\right)\,\mathrm{d}s\,, \end{aligned} \tag{6.29}$$

and, for $\alpha \in \{y, z\}$,

$$R_{i,\alpha}\left(t\right) = R_{i,x}\left(t; \dot{\gamma} = 0\right)\,. \tag{6.30}$$

In order to calculate correlation functions between the different $R_{i,\alpha}$, one has to know the correlation functions $\langle E_{i,\alpha}\left(t\right) E_{j,\beta}\left(t'\right)\rangle$, which in turn depend on $\langle \Gamma_{i,\alpha}\left(t\right)\Gamma_{j,\beta}\left(t'\right)\rangle$.

The expectation values for the random forces are zero, $\langle \Gamma_{i,\alpha}\rangle = 0$, and thus $\langle E_{i,\alpha}\rangle = 0$.

Interpreting H_{ij} from equation (6.3) as a $N \times N$ matrix, $N = 3$ being the number of interacting particles, and denoting the inverse of H_{ij} as $\left(H^{-1}\right)_{ij}$, it holds that[107]

$$\langle \Gamma_{i,\alpha}\left(t\right)\Gamma_{j,\beta}\left(t'\right)\rangle = 2k_BT\left(H^{-1}\right)_{ij}\delta_{\alpha\beta}\delta\left(t-t'\right), \tag{6.31}$$

[107] Doi and Edwards, *The Theory of Polymer Dynamics.*

where k_B is Boltzmann's constant and T is the temperature.

Having (see equation (6.3))

$$H = \begin{pmatrix} A & B & \frac{B}{\sqrt{2}} \\ B & A & B \\ \frac{B}{\sqrt{2}} & B & A \end{pmatrix} , \tag{6.32}$$

one can calculate the inverse matrix to be

$$H^{-1} = \frac{1}{\det(H)} \begin{pmatrix} A^2 - B^2 & \frac{B^2}{\sqrt{2}} - AB & B^2 - \frac{AB}{\sqrt{2}} \\ \frac{B^2}{\sqrt{2}} - AB & A^2 - \frac{B^2}{2} & \frac{B^2}{\sqrt{2}} - AB \\ B^2 - \frac{AB}{\sqrt{2}} & \frac{B^2}{\sqrt{2}} - AB & A^2 - B^2 \end{pmatrix} , \tag{6.33}$$

where

$$\det(H) = A^3 + \sqrt{2}B^3 - \frac{5}{2}AB^2 \tag{6.34}$$

is the determinant of H.

With this, equation (6.31), and the definitions of $\mathbf{E}_i$ in (6.11) and (6.12), one can calculate

$$\langle E_{i,\alpha}(t) E_{j,\beta}(t') \rangle = \frac{2k_BT}{\det(H)} \Xi_{ij} \delta_{\alpha\beta} \delta(t - t') , \tag{6.35}$$

where

$$\begin{aligned} \Xi_{11} = \Xi_{22} &= 2A^4 - 2A^3B - 5A^2B^2 + \left(5 + 2\sqrt{2}\right) AB^3 - 2\sqrt{2}B^4 \\ \Xi_{12} = \Xi_{21} &= -A^4 + \left(2 - \frac{1}{\sqrt{2}}\right) A^3B + \frac{5}{2}A^2B^2 + \frac{\sqrt{2} - 20}{4}AB^3 + \left(2\sqrt{2} - 1\right) B^4 . \end{aligned} \tag{6.36}$$

With equations (6.28) and (6.29), one can now proceed to calculate the correlation functions. Defining

$$g(u) := 1 - u(t - u)\dot{\gamma}^2 , \tag{6.37}$$

and assuming, without loss of generality, that $t > 0$,

$$\begin{aligned}
&\langle R_{1,x}(t)\, R_{1,x}(0)\rangle \\
&= \int\limits_{u=-\infty}^{t} \mathrm{d}u \int\limits_{s=-\infty}^{0} \mathrm{d}s \exp\left(a\left(t-u-s\right)\right) \cdot \\
&\quad \cdot \big(\cosh\left(\left(t-u\right)b\right)\cosh\left(-sb\right)\big(\langle E_{1,x}(u)\, E_{1,x}(s)\rangle - \dot{\gamma}^2 s\left(t-u\right)\langle E_{1,y}(u)\, E_{1,y}(s)\rangle\big) \\
&\quad + \cosh\left(\left(t-u\right)b\right)\sinh\left(-sb\right)\big(\langle E_{1,x}(u)\, E_{2,x}(s)\rangle - \dot{\gamma}^2 s\left(t-u\right)\langle E_{1,y}(u)\, E_{2,y}(s)\rangle\big) \\
&\quad + \sinh\left(\left(t-u\right)b\right)\cosh\left(-sb\right)\big(\langle E_{2,x}(u)\, E_{1,x}(s)\rangle - \dot{\gamma}^2 s\left(t-u\right)\langle E_{2,y}(u)\, E_{1,y}(s)\rangle\big) \\
&\quad + \sinh\left(\left(t-u\right)b\right)\sinh\left(-sb\right)\big(\langle E_{2,x}(u)\, E_{2,x}(s)\rangle - \dot{\gamma}^2 s\left(t-u\right)\langle E_{2,y}(u)\, E_{2,y}(s)\rangle\big)\big) \\
&= \frac{2k_BT}{\det\left(H\right)} \int\limits_{u=-\infty}^{t} \int\limits_{s=-\infty}^{0} \exp\left(a\left(t-u-s\right)\right)\delta\left(s-u\right) \cdot \\
&\quad \cdot \big(\Xi_{11}\cosh\left(\left(t-u\right)b\right)\cosh\left(-sb\right)\big(1-\dot{\gamma}^2 s\left(t-u\right)\big) \\
&\quad + \Xi_{12}\cosh\left(\left(t-u\right)b\right)\sinh\left(-sb\right)\big(1-\dot{\gamma}^2 s\left(t-u\right)\big) \\
&\quad + \Xi_{12}\sinh\left(\left(t-u\right)b\right)\cosh\left(-sb\right)\big(1-\dot{\gamma}^2 s\left(t-u\right)\big) \\
&\quad + \Xi_{11}\sinh\left(\left(t-u\right)b\right)\sinh\left(-sb\right)\big(1-\dot{\gamma}^2 s\left(t-u\right)\big)\big)\ \mathrm{d}s\,\mathrm{d}u \\
&= \frac{2k_BT}{\det\left(H\right)} \int\limits_{u=-\infty}^{t} \exp\left(a\left(t-2u\right)\right) g\left(u\right) \cdot \\
&\quad \cdot \left(\Xi_{11}\cosh\left(\left(t-u\right)b\right)\cosh\left(-ub\right) + \Xi_{11}\sinh\left(\left(t-u\right)b\right)\sinh\left(-ub\right)\right. \\
&\quad \left. + \Xi_{12}\cosh\left(\left(t-u\right)b\right)\sinh\left(-ub\right) + \Xi_{12}\sinh\left(\left(t-u\right)b\right)\cosh\left(-ub\right)\right)\ \mathrm{d}u \\
&= \frac{2k_BT}{\det\left(H\right)} \cdot \\
&\quad \cdot \int\limits_{u=-\infty}^{t} \exp\left(a\left(t-2u\right)\right) g\left(u\right)\left(\Xi_{11}\cosh\left(\left(t-2u\right)b\right) + \Xi_{12}\sinh\left(\left(t-2u\right)b\right)\right)\ \mathrm{d}u\,.
\end{aligned} \tag{6.38}$$

For symmetry reasons,

$$\langle R_{1,x}(t)\, R_{1,x}(0)\rangle = \langle R_{2,x}(t)\, R_{2,x}(0)\rangle \tag{6.39}$$

and $\langle R_{i,\alpha}(t)\, R_{i,\alpha}(0)\rangle$ for $\alpha \neq x$ is equivalent to $\langle R_{i,x}(t)\, R_{i,x}(0)\rangle$ except that $g\left(u\right)$ has to be replaced by 1.

For $t = 0$, one obtains (with $\alpha \in \{y, z\}$)

$$\begin{aligned}
\langle R_{1,x}(0)\, R_{1,x}(0)\rangle &= \frac{2k_BT}{\det(H)} \int\limits_{u=-\infty}^{t} \exp(-2au)\, g(u)\, (\Xi_{11}\cosh(-2ub) + \Xi_{12}\sinh(-2ub))\, \mathrm{d}u \\
&= \frac{k_BT}{\det(H)} \left(\frac{a\Xi_{11} - b\Xi_{12}}{b^2 - a^2} \right. \\
&\qquad \left. + \frac{\dot{\gamma}^2}{2\left(b^2 - a^2\right)^3} \left(\Xi_{11}\left(a^3 + 3ab^2\right) - \Xi_{12}\left(3a^2b + b^3\right)\right) \right) \\
\langle R_{1,\alpha}(0)\, R_{1,\alpha}(0)\rangle &= \frac{2k_BT}{\det(H)} \int\limits_{u=-\infty}^{t} \exp(-2au)\, (\Xi_{11}\cosh(-2ub) + \Xi_{12}\sinh(-2ub))\, \mathrm{d}u \\
&= \frac{k_BT}{\det(H)} \frac{a\Xi_{11} - b\Xi_{12}}{b^2 - a^2}\,.
\end{aligned} \tag{6.40}$$

As such, the expectation value for the bond length vector is calculated to be

$$\begin{aligned}
\left\langle \mathbf{R}_1^2 \right\rangle &= \langle R_{1,x}(0)\, R_{1,x}(0)\rangle + \langle R_{1,y}(0)\, R_{1,y}(0)\rangle + \langle R_{1,z}(0)\, R_{1,z}(0)\rangle \\
&= \frac{k_BT}{\det(H)} \left(3\frac{a\Xi_{11} - b\Xi_{12}}{b^2 - a^2} \right. \\
&\qquad \left. + \frac{\dot{\gamma}^2}{2\left(b^2 - a^2\right)^3} \left(\Xi_{11}\left(a^3 + 3ab^2\right) - \Xi_{12}\left(3a^2b + b^3\right)\right) \right)\,.
\end{aligned} \tag{6.41}$$

Substituting the definitions of $\det(H)$, a, b, Ξ_{11}, Ξ_{12}, A, and B into this expression, one obtains a complicated polynomial in k_BT, η, $\dot{\gamma}$, R_H, the spring constant K and the root-mean-square bond length l. Since, by definition, $\left\langle \mathbf{R}_1^2 \right\rangle = l^2$, one has obtained a polynomial equation that can be solved numerically. Figure 6.1 shows a graph of the solution.

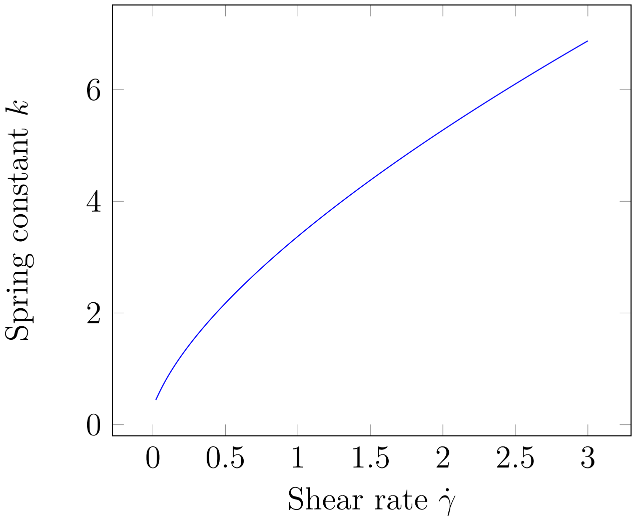

Figure 6.1: Solution of $\left\langle \mathbf{R}_i^2 \right\rangle = l^2$ (see equation (6.41)) for $R_H = 0.3$, $\eta = 8.7$, $l = 2$ and $k_B T = 1$. For large $\dot{\gamma}$, K scales with $\dot{\gamma}^{2/3}$.

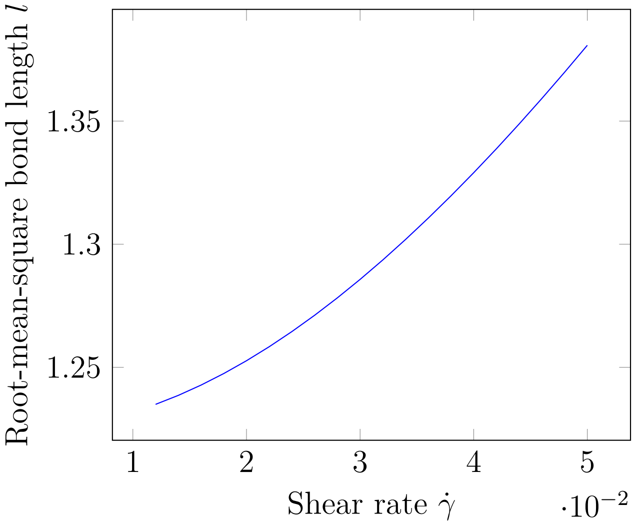

Figure 6.2: Dependence of $l = \sqrt{\left\langle \mathbf{R}_i^2 \right\rangle}$ on the shear rate $\dot{\gamma}$ (see equation (6.41)) for $K = 1$, $R_H = 0.3$, $\eta = 8.7$, and $k_B T = 1$.

Chapter 7

Conclusion and Outlook

Multi-Particle Collision Dynamics (MPC) simulations, such as implemented in the *MPCDSim* software packaged presented in this thesis, offer an important and efficient tool if one seeks to understand the role of hydrodynamic interactions in large-scale systems containing macroscopic objects, where “macroscopic” is relative to the size of the fluid’s constituents and may refer to passive solute particles, or even active entities such as self-propelling living organisms.

This thesis’ subject, an extension of MPC to incorporate elastic effects in fluids, allows for the treatment of a much broader class of problems than that of systems containing purely viscous fluids. Thus, one is now able to model more realistic scenarios not only in basic research into rheology, but also in a wide range of applications in engineering as well as biology and medicine. With this improved computer simulation technique at hand, one can now address, among others, the following questions:

- Which phenomena in fluidic systems arise due to the introduction of viscoelasticity?
- How do the emerging effects depend on rheological properties of the fluid?
- How can one tailor these properties, possibly through the engineering of microscopic details of the fluid constituents, to obtain a desired result?

Finding answers to these questions requires a certain understanding of the fundamental behavior of the (simulated) fluid itself, i.e. without interactions with solute objects. One way this thesis contributes to this knowledge is by deriving, starting from the low-Reynolds-number limit of the linear viscoelastic Navier-Stokes equation, the velocity autocorrelation function $\tilde{C}_{\mathrm{v}}^{\mathrm{T}}$ in Fourier space, and demonstrating that this solution indeed describes the simulation data for viscoelastic MPC fluids.

On the one hand, this information then can be used in future works to more profoundly understand how complex systems deviate from the simpler case of the pure viscoelastic fluid. On the other, the analytic nature of the solution found for both $\tilde{C}_{\mathrm{v}}^{\mathrm{T}}$ and for the relationship between the root-mean-square bond length in trimers and the corresponding spring constants, extends the freedom one has in designing fluids that exhibit certain features; it thus helps search systematically for fluid properties deemed optimal for a given problem.

These tasks – understanding a system's behavior, and manipulating that behavior to one's liking – are, of course, ultimately linked by the underlying theory. This thesis surely does not constitute an exhaustive treatment of said theory, but the author hopes that it serves as a useful foundation for future exploration of the topic and, in the spirit of the TU Wien's mission *technology for people*, that it finds applications that advance science and benefit society.

Appendix A

Notation and Mathematical Conventions

Definitions When defining a new quantity in terms of previously introduced quantities, the symbols := and =: are used. For example, if one wanted to define a new quantity X as the sum of the known objects g and h, one could synonymously write either

$$X := g + h \tag{A.1}$$

or

$$g + h =: X \tag{A.2}$$

Special Sets The set $\mathbb{N}_0$ is the set of natural numbers, including 0, and $\mathbb{N}_+$ is the set of natural numbers, excluding 0. $\mathbb{Z}$ is the set of integers. $\mathbb{R}$ is the set of real numbers, $\mathbb{R}_+$ is the set of real numbers greater than 0, and $\mathbb{R}_-$ is the set of real numbers smaller than 0. The set of complex numbers is denoted by $\mathbb{C}$.

The unit 2-sphere, that is the set of all points which lie a (Euclidean) distance 1 away from the center of the coordinate system, is denoted by

$$\mathbb{S}^2 := \left\{ \mathbf{r} \in \mathbb{R}^3 \mid |\mathbf{r}| = 1 \right\} . \tag{A.3}$$

Vectors Vectors are denoted by bold letters, such as $\mathbf{R}$ or $\mathbf{v}_i$, where i is part of the symbol's name, e.g. in the case in situations where the index i serves to designate a particular particle's velocity. The components of a vector, usually with respect to a Cartesian coordinate system with axes x, y, and z, or synonymously 1, 2, and 3, are referred to as, for example, R_x or R_1, where no bold face is used. If the vector itself had an index, as in the case $\mathbf{v}_i$, the y-component of $\mathbf{v}_i$ is denoted by $v_{i,y}$ or equivalently $v_{i,2}$. The norm $|\mathbf{R}|$ of a vector is, unless noted otherwise, to be understood as the Euclidean norm:

$$|\mathbf{R}| := \sqrt{R_1^2 + R_2^2 + R_3^2} \tag{A.4}$$

The scalar product, or dot product, between two vectors $\mathbf{a}$ and $\mathbf{b}$, is denoted by $\mathbf{a} \cdot \mathbf{b}$, while the cross product is written $\mathbf{a} \times \mathbf{b}$.

Index notation Throughout this document, the *index notation* is used, along with the *Einstein summation convention* – that is, repeated indices are summed over, although the summation sign is suppressed. Partial derivatives with respect to time are denoted by ∂_t, while partial derivatives with respect to the spatial coordinate i are written as ∂_i.

Fourier transformation Unless noted otherwise, the Fourier transform of a function f is synonymously denoted by either $\tilde{f}$ or $\mathcal{F}\left\{f\right\}$. Similarly, the inverse Fourier transform of $\tilde{f}$ is written f or $\mathcal{F}^{-1}\left\{\tilde{f}\right\}$.

If it is desired to explicitly assign a symbol for the argument of the transformation's resultant function, the notation chosen is $\tilde{f}\left(\omega\right)$ or $\mathcal{F}\left\{f\right\}\left(\omega\right)$ for the Fourier transform, and $f\left(t\right)$ or $\mathcal{F}^{-1}\left\{\tilde{f}\right\}\left(t\right)$ for its inversion. It may be necessary to be specific about the argument that is transformed, e.g. to avoid ambiguity. In such a case, the notation $\tilde{f}\left(\mathbf{r},\omega\right) = \mathcal{F}_t\left\{f\left(\mathbf{r},t\right)\right\}\left(\mathbf{r},\omega\right)$ or $f\left(\mathbf{r},t\right) = \mathcal{F}_\omega^{-1}\left\{\tilde{f}\left(\mathbf{r},\omega\right)\right\}\left(\mathbf{r},t\right)$ is chosen.

While the arguments can, of course, be named arbitrarily, it is customary to call them $\mathbf{k}$ if the transformed argument of f is a (vector of) spatial coordinates $\mathbf{r}$, and ω if the transformed variable is the time t. In the definition of the Fourier transform and its inverse, the following convention, called *unitary angular frequency convention*, is used:

$$\mathcal{F}\left\{f\right\}\left(\mathbf{k}\right) = \tilde{f}\left(\mathbf{k}\right) := \left(2\pi\right)^{-\frac{D}{2}} \int\limits_{\mathbb{R}^D} f\left(\mathbf{r}\right) \exp\left(-\mathrm{i}\mathbf{k}\cdot\mathbf{r}\right)\, \mathrm{d}r^D\,, \tag{A.5}$$

$$\mathcal{F}^{-1}\left\{\tilde{f}\right\}\left(\mathbf{r}\right) = f\left(\mathbf{r}\right) = \left(2\pi\right)^{-\frac{D}{2}} \int\limits_{\mathbb{R}^D} \tilde{f}\left(\mathbf{k}\right) \exp\left(\mathrm{i}\mathbf{k}\cdot\mathbf{r}\right)\, \mathrm{d}k^D\,. \tag{A.6}$$

Here, D is the dimension of the vector space the argument affected by the (inverse) Fourier transformation is an element of. Since the integration range is the entire $\mathbb{R}^D$, said function has to be defined at least in the cases where the transformed argument takes on values in $\mathbb{R}^D$.

Laplace transformation Similarly, the (unilateral) Laplace transform of a function f is denoted by $\hat{f}$ or $\mathcal{L}\left\{f\right\}$, while the inversion of the Laplace transform is written $\mathcal{L}^{-1}\left\{\hat{f}\right\}$. The arguments of the original and the transformed functions may be specified, if required, as in the case of Fourier transforms.

The arguments of $\hat{f}$ are usually named s if the transformed argument of f was the time t, and vice versa. The Laplace transform of f is defined by

$$\mathcal{L}\left\{f\right\}\left(s\right) = \hat{f}\left(s\right) := \int\limits_{t=0}^{\infty} f\left(t\right) \exp\left(-st\right)\,\mathrm{d}t\,. \tag{A.7}$$

Special functions Given a complex number $z \in \mathbb{C}$ with $z = x + \mathrm{i}y$, $x \in \mathbb{R}$, $y \in \mathbb{R}$, the *real part* of z is denoted by $\mathrm{Re}\left(z\right) := x$, while the *imaginary part* is written as $\mathrm{Im}\left(z\right) := y$.

The convention and notation chosen for the *Heaviside step function* is

$$\theta\left(x\right) := \begin{cases} 0, & x < 0 \\ \frac{1}{2}, & x = 0 \\ 1, & x > 0 \end{cases}. \tag{A.8}$$

The *floor function* is defined as

$$\lfloor x \rfloor := \max\left(\left\{z \in \mathbb{Z} \mid z \leq x\right\}\right)\,. \tag{A.9}$$

The *gamma function*[108] is defined on $\mathbb{R}_+$ by

$$\Gamma\left(x\right) := \int\limits_{y=0}^{\infty} y^{x-1} \exp\left(-y\right)\,\mathrm{d}y\,. \tag{A.10}$$

Probability Distributions and Sampling of Random Numbers The sampling of a random variable X from a given probability distribution function F is denoted by $X \sim F$.

Given an interval $I \subset \mathbb{R}$, the *uniform probability distribution* over the interval I is called $U\left(I\right)$.

[108] R. V. Hogg, J. W. McKean, and A. T. Craig. *Introduction to Mathematical Statistics.* 7th ed. Pearson, 2012, Chapter 3.3.

The *gamma distribution*[109] has two parameters $a \in \mathbb{R}_+$, $b \in \mathbb{R}_+$ and is defined for $x \in \mathbb{R}$ by

$$f_\Gamma\left(x; a, b\right) := \frac{1}{b^a \Gamma\left(a\right)} x^{a-1} \exp\left(-\frac{x}{b}\right) \theta\left(x\right) \ . \tag{A.11}$$

[109] Hogg, McKean, and Craig, *Introduction to Mathematical Statistics*, Equation (3.3.1).

Appendix B

Properties of the Fourier Transformation

B.1 Fourier Transformation of Derivatives

Given a function $f(\mathbf{r})$ and its Fourier transform $\tilde{f}(\mathbf{k})$, the inverse Fourier transform of $\tilde{f}(\mathbf{k})$is given by equation (A.6):

$$f(\mathbf{r}) = (2\pi)^{-\frac{D}{2}} \int\limits_{\mathbb{R}^D} \tilde{f}(\mathbf{k}) \exp(\mathrm{i}\mathbf{k}\cdot\mathbf{r})\ \mathrm{d}k^D\,. \tag{B.1.1}$$

Taking the derivative ∂_j of this equation with respect to the j-component of $\mathbf{r}$, one can write

$$\begin{aligned} \partial_j f(\mathbf{r}) &= (2\pi)^{-\frac{D}{2}} \int\limits_{\mathbb{R}^D} \tilde{f}(\mathbf{k})\, \partial_j \exp(\mathrm{i}\mathbf{k}\cdot\mathbf{r})\ \mathrm{d}k^D \\ &= (2\pi)^{-\frac{D}{2}} \int\limits_{\mathbb{R}^D} \tilde{f}(\mathbf{k})\, \mathrm{i}k_j \exp(\mathrm{i}\mathbf{k}\cdot\mathbf{r})\ \mathrm{d}k^D\,. \end{aligned} \tag{B.1.2}$$

Defining $g(\mathbf{r}) := \partial_j f(\mathbf{k})$ and comparing the equation above with (A.6), i.e.

$$g(\mathbf{r}) = (2\pi)^{-\frac{D}{2}} \int\limits_{\mathbb{R}^D} \tilde{g}(\mathbf{k}) \exp(\mathrm{i}\mathbf{k}\cdot\mathbf{r})\ \mathrm{d}k^D\,, \tag{B.1.3}$$

one sees that

$$\tilde{g}(\mathbf{k}) = \mathrm{i}k_j \tilde{f}(\mathbf{k})\ . \tag{B.1.4}$$

B.2 Fourier Transform of 1 and the Dirac Delta Function

The Dirac delta function δ can be represented as[110]

$$\delta\left(t-t'\right)=\frac{1}{2\pi}\int\limits_{\omega=-\infty}^{\infty}\exp\left(\mathrm{i}\left(t-t'\right)\omega\right)\,\mathrm{d}\omega\,, \tag{B.2.1}$$

which, conversely, allows one to define the inverse Fourier transform of 1 via

$$\int\limits_{\omega=-\infty}^{\infty}1\cdot\exp\left(\mathrm{i}t\omega\right)\,\mathrm{d}\omega=2\pi\delta\left(t\right). \tag{B.2.2}$$

Since the Dirac delta function is even in its argument, also the relation

$$\int\limits_{\omega=-\infty}^{\infty}\exp\left(-\mathrm{i}t\omega\right)\,\mathrm{d}\omega=2\pi\delta\left(t\right) \tag{B.2.3}$$

holds.

[110] F. W. J. Olver, D. W. Lozier, R. F. Boisvert, and C. W. Clark, eds. *NIST Handbook of Mathematical Functions.* Cambridge University Press, 2010, Equation (1.17.12).

Appendix C

Properties of the Laplace Transformation

C.1 Laplace Transformation of Derivatives

For a given function $f(t)$, let $g(t) := \partial_t f(t)$. Then, inserting $g(t)$ into equation (A.7), one can write for the Laplace transform of $g(t)$

$$\begin{aligned} \hat{g}(s) &:= \int\limits_{t=0}^{\infty} g(t) \exp(-st)\, \mathrm{d}t \\ &= \int\limits_{t=0}^{\infty} (\partial_t f(t)) \exp(-st)\, \mathrm{d}t \\ &= [f(t)\exp(-st)]_{t=0}^{\infty} - \int\limits_{t=0}^{\infty} f(t)\,(\partial_t \exp(-st))\, \mathrm{d}t\,. \end{aligned} \tag{C.1.1}$$

Assuming that the upper boundary term tends to 0, i.e.

$$\lim_{t\to\infty} (f(t)\exp(-st)) = 0\,, \tag{C.1.2}$$

one arrives at the following expression for the Laplace transform of a derivative $g(t) := \partial_t f(t)$:

$$\begin{aligned} \hat{g}(s) &= -f(0) + s\int\limits_{t=0}^{\infty} f(t)\exp(-st)\, \mathrm{d}t \\ &= s\hat{f}(s) - f(0)\ . \end{aligned} \tag{C.1.3}$$

C.2 Convolution Theorem

Given two functions $f\left(t\right)$ and $g\left(t\right)$ and defining their convolution to be

$$h\left(t\right) := \int\limits_{\tau=0}^{t} f\left(\tau\right) g\left(t-\tau\right)\,\mathrm{d}\tau\,, \tag{C.2.1}$$

one can calculate the Laplace transform $\hat{h}\left(s\right)$ of $h\left(t\right)$ by insertion of equation (C.2.1) into the definition (A.7):

$$\begin{aligned}\hat{h}\left(s\right) &:= \int\limits_{t=0}^{\infty} h\left(t\right)\exp\left(-st\right)\,\mathrm{d}t \\ &= \int\limits_{t=0}^{\infty}\int\limits_{\tau=0}^{t} f\left(\tau\right) g\left(t-\tau\right)\exp\left(-st\right)\,\mathrm{d}\tau\,\mathrm{d}t \\ &= \int\limits_{t=0}^{\infty}\int\limits_{\tau=0}^{\infty} f\left(\tau\right) g\left(t-\tau\right)\exp\left(-st\right)\theta\left(t-\tau\right)\,\mathrm{d}\tau\,\mathrm{d}t\,.\end{aligned} \tag{C.2.2}$$

Interchanging the order of integration and defining $u\left(t\right) := t-\tau$ for a fixed τ, one can perform the change of integration variables $t \to u$,

$$\begin{aligned}\mathrm{d}t &= \mathrm{d}u\,,\\ u\left(t=0\right) &= -\tau\,,\\ u\left(t=\infty\right) &= \infty\,,\end{aligned} \tag{C.2.3}$$

so that

$$\begin{aligned}\hat{h}\left(s\right) &= \int\limits_{\tau=0}^{\infty}\int\limits_{u=-\tau}^{\infty} f\left(\tau\right) g\left(u\right)\exp\left(-s\left(u+\tau\right)\right)\theta\left(u\right)\,\mathrm{d}u\,\mathrm{d}\tau \\ &= \int\limits_{\tau=0}^{\infty}\int\limits_{u=0}^{\infty} f\left(\tau\right) g\left(u\right)\exp\left(-s\left(u+\tau\right)\right)\,\mathrm{d}u\,\mathrm{d}\tau \\ &= \int\limits_{\tau=0}^{\infty} f\left(\tau\right)\exp\left(-s\tau\right)\,\mathrm{d}\tau \int\limits_{u=0}^{\infty} g\left(u\right)\exp\left(-su\right)\,\mathrm{d}u \\ &= \hat{f}\left(s\right)\hat{g}\left(s\right)\,.\end{aligned} \tag{C.2.4}$$

Appendix D

Tensors

D.1 The Kronecker Tensor δ_{ij}

The Kronecker tensor δ_{ij} is defined such that

$$\delta_{ij} = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases} . \tag{D.1.1}$$

Evidently, the Kronecker tensor is symmetric: $\delta_{ij} = \delta_{ji}$.

D.2 The Levi-Civita Tensor ε_{ijk}

The Levi-Civita tensor ε_{ijk} is defined as

$$\varepsilon_{ijk} = \begin{cases} 1, & (i,j,k) \text{ is an even permutation of } (1,2,3) \\ -1, & (i,j,k) \text{ is an odd permutation of } (1,2,3) \\ 0, & \text{else} \end{cases} . \tag{D.2.1}$$

Therefore, ε_{ijk} is anti-symmetric in all pairs of its indices, i.e. $\varepsilon_{ijk} = -\varepsilon_{jik}$, $\varepsilon_{ijk} = -\varepsilon_{kji}$, and $\varepsilon_{ijk} = -\varepsilon_{ikj}$.

D.3 Symmetric and Anti-Symmetric Parts of a Tensor

Any tensor T_{ij} of order 2 can be decomposed into a symmetric part $S_{ij} = S_{ji}$ and an anti-symmetric part $A_{ij} = -A_{ji}$:

$$T_{ij} = \underbrace{\frac{1}{2}\left(T_{ij} + T_{ji}\right)}_{=:S_{ij}} + \underbrace{\frac{1}{2}\left(T_{ij} - T_{ji}\right)}_{=:A_{ij}} \,. \tag{D.3.1}$$

Pairs of indices of higher-order tensors can be symmetrized (denoted by parentheses) and anti-symmetrized (denoted by square brackets) in an analogous manner; for example,

$$\begin{aligned} T_{i(jk)l} &:= \frac{1}{2}\left(T_{ijkl} + T_{ikjl}\right) \\ T_{i[jk]l} &:= \frac{1}{2}\left(T_{ijkl} - T_{ikjl}\right) \,. \end{aligned} \tag{D.3.2}$$

The contraction of the Levi-Civita tensor with a symmetric tensor S_{jk} of order 2 is 0. To prove this, let $a \in \{1,2,3\}$ be an arbitrary index. Furthermore, let $b \in \{1,2,3\} \setminus \{a\}$ and finally $c \in \{1,2,3\} \setminus \{a,b\}$. Then,

$$\begin{aligned} \varepsilon_{ajk} S_{jk} &= \varepsilon_{abc} S_{bc} + \varepsilon_{acb} S_{cb} \\ &= \varepsilon_{abc} S_{bc} - \varepsilon_{abc} S_{bc} = 0 \,. \end{aligned} \tag{D.3.3}$$

Contraction of the Levi-Civita tensor ε_{ajk} with an anti-symmetric tensor A_{jk}, on the other hand, gives

$$\begin{aligned} \varepsilon_{ajk} A_{jk} &= \varepsilon_{abc} A_{bc} + \varepsilon_{acb} A_{cb} \\ &= 2\varepsilon_{abc} A_{bc} = 2\varepsilon_{acb} A_{cb} \,. \end{aligned} \tag{D.3.4}$$

Therefore, for a general tensor T of order n, one obtains

$$\varepsilon_{j i_x i_y} T_{i_1 i_2 \ldots i_x i_y \ldots i_n} = \varepsilon_{j i_x i_y} T_{i_1 i_2 \ldots [i_x i_y] \ldots i_n} \,. \tag{D.3.5}$$

Appendix E

Uniform Sampling from the 2-Sphere

There are various methods[111,112,113,114,115,116,117] that can be used to uniformly sample from the n-sphere $\mathbb{S}^n$ in general, and the 2-sphere $\mathbb{S}^2$ in particular. For this thesis, only the latter was relevant, and the following algorithm was used:

Let $X_1 \sim U\left([0,1]\right)$ and independently $X_2 \sim U\left([0,1)\right)$. Then, let $z := 2X_1 - 1$ and $\varphi := 2\pi X_2$, so that z is uniformly distributed over $[-1,1]$ and φ is uniformly distributed over $[0,2\pi)$. Finally, the Cartesian coordinates R_1, R_2, and R_3 of the uniformly sampled, random unit vector $\mathbf{R} \in \mathbb{S}^2$ are

$$\begin{aligned} R_1 &:= \sqrt{1-z^2}\cos\left(\varphi\right) \\ R_2 &:= \sqrt{1-z^2}\sin\left(\varphi\right) \\ R_3 &:= z\,. \end{aligned} \tag{E.1}$$

The implementation in *MPCDSim* deviates from the algorithm just described in that no guarantee is given on whether the end points of the distribution function intervals have non-zero probability of being sampled. The reason for this is that the primitive random number gener-

[111] Y. Tashiro. "On methods for generating uniform random points on the surface of a sphere". *Annals of the Institute of Statistical Mathematics* 29 (1977), 295.

[112] G. Marsaglia. "Choosing a Point from the Surface of a Sphere". *The Annals of Mathematical Statistics* 43 (1972), 645.

[113] E. W. Weisstein. *Sphere Point Picking*. URL: `http://mathworld.wolfram.com/SpherePointPicking.html`.

[114] E. Allahyarov and G. Gompper. "Mesoscopic solvent simulations: Multiparticle-collision dynamics of three-dimensional flows". *Physical Review E* 66 (2002), 036702. Appendix.

[115] M. E. Muller. "A note on a method for generating points uniformly on n-dimensional spheres". *Communications of the ACM* 2 (1959), 19.

[116] J. M. Cook. "Rational formulae for the production of a spherically symmetric probability distribution". *Mathematics of Computation* 11 (1957), 81.

[117] J. S. Hicks and R. F. Wheeling. "An efficient method for generating uniformly distributed points on the surface of an n-dimensional sphere". *Communications of the ACM* 2 (1959), 17.

ators provided by various operating systems and programming libraries do not agree whether either end point of the interval is part of the set of possible results. While it is mathematically possible to construct a function that has the same range of return values for any operating system or library used, it comes at a cost in computational efficiency.

However, one can convince oneself that the question of whether the interval end points can be sampled is of little importance in practice: The computations were performed with IEEE-754[118, 119] double-precision (i.e. 64-bit) arithmetic. One can roughly estimate the number of distinct floating point values in the interval $[0, 1]$: since there are about approximately 2^{64} distinct and finite floating point values, about half of which are positive, and since about half of the exponents are smaller than 0, there are of the order of $2^{62} \approx 4.6 \cdot 10^{8}$ representable numbers in $[0, 1]$, such that the addition or omission of the boundary points only has a negligible impact.

[118] *IEEE Standard for Floating-Point Arithmetic.*

[119] Goldberg, “What every computer scientist should know about floating-point arithmetic”.

Bibliography

Adinetz, A. V. “GPUs and Other Non-Standard Hardware”. *Computing Solids: Models, ab-initio methods and supercomputing.* Ed. by S. Blügel, N. Helbig, V. Meden, and D. Wortmann. Vol. 74. Key Technologies. Forschungszentrum Jülich, 2014. Chap. D5.

Allahyarov, E. and Gompper, G. “Mesoscopic solvent simulations: Multiparticle-collision dynamics of three-dimensional flows”. *Physical Review E* 66 (2002), 036702.

Allen, M. P. and Tildesley, D. J. *Computer Simulation of Liquids.* Reprint by The Ipswich Book Co Ltd in 1991. Clarendon Press, 1987.

Batchelor, G. K. *An Introduction to Fluid Dynamics.* Cambridge Mathematical Library. Cambridge University Press, 2000.

Bird, R. B., Armstrong, R. C., and Hassager, O. *Fluid Mechanics.* 2nd ed. Vol. 1. Dynamics of Polymeric Liquids. John Wiley & Sons, 1987.

Bird, R. B., Curtiss, C. F., Armstrong, R. C., and Hassager, O. *Kinetic Theory.* 2nd ed. Vol. 2. Dynamics of Polymeric Liquids. John Wiley & Sons, 1987.

Cook, J. M. “Rational formulae for the production of a spherically symmetric probability distribution”. *Mathematics of Computation* 11 (1957), 81.

Dhont, J. K. G., Gompper, G., Lang, P. R., Richter, D., Ripoll, M., Willbold, D., and Zorn, R., eds. *Macromolecular Systems in Soft and Living Matter.* Vol. 20. Key Technologies. Forschungszentrum Jülich, 2011.

Dhont, J. K. G., Gompper, G., and Richter, D., eds. *Soft Matter: Complex Materials on Mesoscopic Scales.* Vol. 10. Matter and Materials. Forschungszentrum Jülich, 2002.

Doi, M. and Edwards, S. F. *The Theory of Polymer Dynamics.* Oxford University Press, 1994.

Dyke, P. *An Introduction to Laplace Transforms and Fourier Series.* Ed. by M. A. J. Chaplain, K. Erdmann, A. MacIntyre, E. Süli, M. R. Tehranchi, and J. F. Toland. 2nd ed. Springer Undergraduate Mathematics Series. Springer, 2014.

Farago, J., Meyer, H., Baschnagel, J., and Semenov, A. N. “Mode-coupling approach to polymer diffusion in an unentangled melt. II. The effect of viscoelastic hydrodynamic interactions”. *Physical Review E* 85 (2012), 051807.

Ferry, J. D. *Viscoelastic Properties of Polymers.* 3rd ed. John Wiley & Sons, 1980.

Frenkel, D. and Smit, B. *Understanding Molecular Simulation. From Algorithms to Applications.* 2nd ed. Academic Press, 2002.

Frisch, U., Hasslacher, B., and Pomeau, Y. “Lattice-Gas Automata for the Navier-Stokes Equation”. *Physical Review Letters* 56 (1986), 1505.

Goldberg, D. “What every computer scientist should know about floating-point arithmetic”. *ACM Computing Surveys* 23 (1991), 5.

Gompper, G., Ihle, T., Kroll, D. M., and Winkler, R. G. *Multi-Particle Collision Dynamics: A Particle-Based Mesoscale Simulation Approach to the Hydrodynamics of Complex Fluids.* Advances in Polymer Science. Springer, 2008.

Hicks, J. S. and Wheeling, R. F. “An efficient method for generating uniformly distributed points on the surface of an n-dimensional sphere”. *Communications of the ACM* 2 (1959), 17.

Hogg, R. V., McKean, J. W., and Craig, A. T. *Introduction to Mathematical Statistics.* 7th ed. Pearson, 2012.

Huang, C., Chatterji, A., Sutmann, G., Gompper, G., and Winkler, R. G. “Cell-level canonical sampling by velocity scaling for multiparticle collision dynamics simulations”. *Journal of Computational Physics* 229 (2010), 168.

Huang, C.-C., Gompper, G., and Winkler, R. G. “Hydrodynamic correlations in multiparticle collision dynamics fluids”. *Physical Review E* 86 (2012), 056711.

Huang, Y., Zhang, X., Ma, Z., Li, W., Zhou, Y., Zhou, J., Zheng, W., and Sun, C. Q. “Size, separation, structural order, and mass density of molecules packing in water and ice”. *Scientific Reports* 3 (2013), 3005.

IEEE Standard for Floating-Point Arithmetic. Institute of Electrical and Electronics Engineers.

Ihle, T. and Kroll, D. M. “Stochastic rotation dynamics: A Galilean-invariant mesoscopic model for fluid flow”. *Physical Review E* 63 (2001), 020201.

— “Stochastic rotation dynamics. I. Formalism, Galilean invariance, and Green-Kubo relations”. *Physical Review E* 67 (2003), 066705.

Jeffreys, H. *Cartesian Tensors.* Cambridge University Press, 1931.

Koks, D. *Explorations in Mathematical Physics.* 1st ed. Springer, 2006.

Kowalik, B. and Winkler, R. G. “Multiparticle collision dynamics simulations of viscoelastic fluids: Shear-thinning Gaussian dumbbells”. *Journal of Chemical Physics* 138 (2013), 104903.

Lamura, A. and Gompper, G. “Numerical study of the flow around a cylinder using multi-particle collision dynamics”. *The European Physical Journal E - Soft Matter* 9 (2002), 477.

Lamura, A., Gompper, G., Ihle, T., and Kroll, D. M. “Multi-particle collision dynamics: Flow around a circular and a square cylinder”. *Europhysics Letters* 56 (2001), 319.

Landau, L. D. and Lifshitz, E. M. *Fluid Mechanics*. 2nd ed. Vol. 6. Course of Theoretical Physics. Pergamon Press, 1987.

Lauga, E. and Goldstein, R. E. “Dance of the microswimmers”. *Physics Today* 65 (2012), 30.

Lauga, E. and Powers, T. R. “The hydrodynamics of swimming microorganisms”. *Reports on Progress in Physics* 72 (2009), 096601.

Lees, A. W. and Edwards, S. F. “The computer study of transport processes under extreme conditions”. *Journal of Physics C: Solid State Physics* 5 (1972), 1921.

Malevanets, A. and Kapral, R. “Mesoscopic model for solvent dynamics”. *The Journal of Chemical Physics* 110 (1999), 8605.

— “Solute molecular dynamics in a mesoscale solvent”. *The Journal of Chemical Physics* 112 (2000), 7260.

Marsaglia, G. “Choosing a Point from the Surface of a Sphere”. *The Annals of Mathematical Statistics* 43 (1972), 645.

Muller, M. E. “A note on a method for generating points uniformly on n-dimensional spheres”. *Communications of the ACM* 2 (1959), 19.

Noguchi, H., Kikuchi, N., and Gompper, G. “Particle-based mesoscale hydrodynamic techniques”. *Europhysics Letters* (2007).

Oberhettinger, F. *Tables of Fourier Transforms and Fourier Transforms of Distributions*. Springer, 1990.

Oberhettinger, F. and Badii, L. *Tables of Laplace Transforms*. Springer, 1973.

Olver, F. W. J., Lozier, D. W., Boisvert, R. F., and Clark, C. W., eds. *NIST Handbook of Mathematical Functions*. Cambridge University Press, 2010.

Purcell, E. M. “Life at low Reynolds number”. *American Journal of Physics* 45 (1977), 3.

Rouse, P. E. “A Theory of the Linear Viscoelastic Properties of Dilute Solutions of Coiling Polymers”. *The Journal of Chemical Physics* 21 (1953), 1272.

Tao, Y.-G., Götze, I. O., and Gompper, G. “Multiparticle collision dynamics modeling of viscoelastic fluids”. *The Journal of Chemical Physics* 128 (2008), 144902. arXiv: `0802.2200 [cond-mat.soft]`.

Tashiro, Y. "On methods for generating uniform random points on the surface of a sphere". *Annals of the Institute of Statistical Mathematics* 29 (1977), 295.

Teraoka, I. *Polymer Solutions. An Introduction to Physical Properties.* John Wiley & Sons, 2002.

Theers, M. and Winkler, R. G. "Effects of thermal fluctuations and fluid compressibility on hydrodynamic synchronization of microrotors at finite oscillatory Reynolds number: a multiparticle collision dynamics simulation study". *Soft Matter* 10 (2014), 5894.

Torvalds, L. et al. *git (Version 1.9.1).* URL: `https://git-scm.com`.

van Heesch, D. *Doxygen.* URL: `http://www.stack.nl/~dimitri/doxygen/manual/index.html`.

Vinberg, E. B. *A Course in Algebra.* Ed. by W. Craig, N. Ivanov, S. G. Krantz, and D. Saltman. Vol. 56. Graduate Studies in Mathematics. American Mathematical Society, 2003.

Weisstein, E. W. *Sphere Point Picking.* URL: `http://mathworld.wolfram.com/SpherePointPicking.html`.

Westphal, E., Singh, S., Huang, C.-C., Gompper, G., and Winkler, R. G. "Multiparticle collision dynamics: GPU accelerated particle-based mesoscale hydrodynamic simulations". *Computer Physics Communications* 185 (2014), 495.