Multi-scale simulations with complex automata: In-stent restenosis and suspension flow
Lorenz, E.

Citation for published version (APA):

General rights
It is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), other than for strictly personal, individual use, unless the work is under an open content license (like Creative Commons).

Disclaimer/Complaints regulations
If you believe that digital publication of certain material infringes any of your rights or (privacy) interests, please let the Library know, stating your reasons. In case of a legitimate complaint, the Library will make the material inaccessible and/or remove it from the website. Please Ask the Library: http://uba.uva.nl/en/contact, or a letter to: Library of the University of Amsterdam, Secretariat, Singel 425, 1012 WP Amsterdam, The Netherlands. You will be contacted as soon as possible.
Multi-scale Simulations with Complex Automata:
In-stent Restenosis and Suspension Flow

Thursday, 11 Nov 2010
10:00
Agnietenkapel
Oudezijds Voorburgwal 231
Amsterdam

There will be a reception after the ceremony

Eric Lorenz
e.lorenz@uva.nl
Multi-scale Simulations with Complex Automata: In-stent Restenosis and Suspension Flow

ACADEMISCH PROEFSCHRIFT

ter verkrijging van de graad van doctor
aan de Universiteit van Amsterdam
op gezag van de Rector Magnificus
prof. dr. D. C. van den Boom
ten overstaan van een door het college voor promoties ingestelde
commissie, in het openbaar te verdedigen in de Agnietenkapel
op donderdag 11 november 2010, te 10:00 uur

door

Eric Lorenz

geboren te Leipzig, Duitsland
The work described in this thesis has been carried out in the Computational Science research group of the University of Amsterdam. Financial support by the European Commission, through the COAST project EU-FP6-IST-FET, contract no. 033664, is gratefully acknowledged.

Author contact: e.lorenz@uva.nl
Contents

1 Introduction ............................................. 9 
  1.1 Distinct Entities .................................. 9 
  1.2 Distinct Scales .................................. 10 
  1.3 Interaction between Scales ......................... 11 
  1.4 Multiple Scales ................................ 12 
  1.5 Outline of this Thesis .............................. 14 

2 Complex Automata for multi-scale modeling ................. 17 
  2.1 Multi-scale Modeling ............................ 17 
     2.1.1 Introduction ................................. 17 
  2.2 Complex Automata ............................... 19 
     2.2.1 A definition ............................... 19 
     2.2.2 The Scale Separation Map ................. 20 
     2.2.3 The Sub-Model Execution Loop ............ 23 
     2.2.4 CxA Multi-Scale Coupling ................. 24 
     2.2.5 Multi-scale modeling strategies .......... 26 
     2.2.6 Execution Model ......................... 28 
        CxA components ............................ 28 
        CxA Communication ........................ 29 
        CxA Initialization and start ............. 30 
        CxA Synchronization ...................... 30 
        CxA Termination ........................... 30 
        Parallelization ........................... 33 
        MUSCLE Library ............................ 33 
     2.2.7 Formalism ................................ 33 
     2.2.8 Scale-splitting error .................... 37 
  2.3 Example: Reaction Diffusion ...................... 37 
  2.4 Conclusion ..................................... 42 

3 In-stent Restenosis .................................... 43 
  3.1 In-stent Restenosis ............................ 43 
  3.2 Multi-scale Model of In-stent Restenosis ........... 44 
     3.2.1 Single Scale Models and Coupling Templates .... 46 
        Bulk Flow Solver (BF) ....................... 46 
        Smooth Muscle Cells Dynamics (SMC) ......... 47 
        Drug Diffusion in Cellular Tissue ............ 48
### 3.2.2 The In-stent Restenosis CxA: Kernels, Connection Scheme and Conduits

49

### 3.3 Simulation Results

50

3.3.1 Benchmark Geometry and Initial Conditions 50

3.3.2 Qualitative assessment of simulation results 51

3.3.3 Sensitivity Analysis 52

### 3.4 3D CxA

53

3.4.1 Single-scale model for thrombus formation (TF) 53

Platelet Thrombus Formation 53

Thrombus Modeling 55

The "Blob" model 56

3.4.2 3D Results 60

### 3.5 Conclusions

61

### 4 Suspension Flow

63

4.1 Rheology of Hard-Sphere Suspensions 64

4.2 HMM Modeling of Suspension Flow 67

4.2.1 Description of the Problem 67

4.2.2 The Heterogeneous Multi-scale Method 68

4.2.3 Submodels and Coupling 69

4.2.4 Scales of the Problem and their Separation 72

4.2.5 More on Coupling 76

4.3 Macroscopic Non-Newtonian Flow ($MaF$) 79

4.3.1 Lattice-Boltzmann Method 79

4.3.2 Coupling 82

4.4 Fully Resolved Suspension Flow ($MiS$) 83

4.4.1 Direct Simulation of Suspended Particles 83

Fluid-particles forces 84

4.4.2 Lees-Edwards Boundary Conditions 88

Implementation 90

Numerical results 92

4.4.3 Corrected Momentum Exchange 96

Asymptotic Analysis 97

Numerical tests 98

Galilean invariant force computations 102

Initialization of new fluid nodes 105

Numerical Results 106

4.4.4 Initial Conditions and Observables 106

4.5 Shear-induced Diffusion 108

4.5.1 Diffusivity Measurements on the Micro-Scale 110

4.5.2 Anisotropic Advection-Diffusion ($MaS$/LPAD) 112

Mapping Macro Lattice Fields $\rightarrow$ LPAD: Interpolation 113

Mapping LPAD $\rightarrow$ Macro Lattice Fields: Kernel Estimator 116

Flow aligned Diffusivity Tensor 117

4.6 A Database coupled in 118

4.6.1 Serial vs. concurrent coupling 118

4.6.2 A Hybrid: Database filled on-the-fly 120

Extra- and Interpolation 120
Classification of the Role of the Database .......................... 123
Parallelization? .................................................. 124
4.7 Other Suspension Flow Models .................................. 125
4.7.1 Suspension balance model .................................. 126
4.7.2 Diffusive-Flux Model ....................................... 127
4.8 Simulations and Results ......................................... 128
4.8.1 Profile Development Dynamics ............................. 129
4.8.2 Equilibrium Profiles: Results and Discussion .......... 131
4.9 Outlook .......................................................... 139
4.10 Conclusion ...................................................... 141

5 Particle Clustering in Shear-Thickening Hard-Sphere Suspensions
5.1 Microstructure of Sheared Suspensions ....................... 143
5.1.1 Simulations .................................................. 144
5.1.2 Geometrical properties of Particle Clusters ............... 145
Cluster Pair Distribution Function ............................... 147
Fractal Dimension of Particle Clusters .......................... 147
Cluster Size Distribution and Typical Cluster Size ........... 150
5.2 2-Particle Collision Model ..................................... 153
5.2.1 Trajectories ................................................. 153
5.2.2 Two-Particle Sticking Probability ......................... 154
Cluster Condition, Sticking Time ................................. 154
Number Density of Collision Partners as Function of $d$ .... 156
Sticking Probability ............................................... 156
5.3 Statistical Clustering Model ..................................... 158
5.3.1 Elementary Processes ...................................... 158
Merging of smaller clusters ...................................... 158
Breaking of larger clusters ...................................... 158
Absorption into larger cluster ................................... 159
Dissolving into smaller clusters ................................. 159
5.3.2 Master Equation for $p_c(N)$ ............................... 159
Results and Discussion ............................................ 159
5.4 Discussion ........................................................ 161

6 Summary and Conclusions ......................................... 163
Bibliography ................................................................ 165
Samenvatting ................................................................ 180
Acknowledgements ...................................................... 182
Publications ................................................................ 183
Chapter 1

Introduction

How does Nature process information? Can we detect and describe the computational structure in natural processes and can we provide a quantitative characterization of essential aspects of this structure?

(Prof. Peter M.A. Sloot defining his research motivation and that of the whole Computational Science research group.)

These are the essential questions behind this thesis. They are the driving force behind Computational Science which does not only assist the solution of mathematical descriptions of natural systems by numerical methods. In fact, the aim of Computational Science is to develop means which can lead to a better study and understanding of natural systems by modeling the various interactions between individual components which give rise to endless signatures of order, disorder, self-organization and self-annihilation. The challenge in this field is that processes studied by natural scientists involve systems that are each described best in a different modeling language from continuous mathematical models to fully discretized cellular automata. Besides the challenges in modeling every single of such systems, the study of information processing between such systems in complex dynamical multi-scale problems is still in the early stages.

1.1 Distinct Entities

Our way to look at natural systems is based on entities appearing on distinct scales: to describe a part of nature we assume the entities of a particular subsystem not to change their properties nor the rules in which they interact with the other entities in such an extent that the concept of “object“ is not applicable anymore. One might argue that this concept is merely a very successful workaround developed by the impressive, but still very limited, cognitive capacity of humans in processing the input from the outer world by means of very compressed representations. However, assuming the most objective view of science, we cannot neglect that nature tends to do the same, or, at least, that we can successfully apply the same approach by mathematical means or that of other modeling languages. The matter of our universe organizes in structures we characterize as cosmic objects, also life on earth expresses itself in relatively independent creatures whose internal dynamics can be
neglected when we want to study their social behavior. When we focus on only one of such entities we find that it is made of other entities. Atomistic theories have been revised a few times in the history of science and the current candidate of an a-tom, the ultimate particle on the ultimately lowest scale, the quark, might be found to consist of smaller structures in the future. All those things are found to behave as entities (in a strange way, though) and from Quantum Field Theory we learn that also fields can be best described by a finite number of dynamical degrees and that these can only assume discrete states (so that even interaction between particles is mediated by particles - photons, vector bosons, gluons, gravitons (postulated but not found yet)). The discretization of space, time and state space has been successfully translated into a theory of Cellular Automata (CA) which successfully applies to most diverse systems (see for example [1] and the very recent [2]). Although it gives rise to a number of inconsistencies, particularly with the debatable probability interpretation of quantum mechanics, the theory of CA has been extended to extreme variants like the ideas of the Calculating Space (Zuse [3]) and Digital Philosophy (Fredkin [4], Wolfram [5]). Particularly Stephen Wolfram promotes the idea that below the Planck scale, thus “below” quantum mechanics, there exists an informational substrate that allows the build-up of time, space, and energy. Such ideas can even be traced back to Einstein [6] who stated: "One can give good reasons why reality cannot at all be represented by a continuous field. From the quantum phenomena it appears to follow with certainty that a finite system of finite energy can be completely described by a finite set of numbers (quantum numbers). This does not seem to be in accordance with a continuum theory, and must lead to attempts to find a purely algebraic theory for the description of reality.”

1.2 Distinct Scales

Typically, there is no continuous crossover between the definitions of an object on one scale and another on another scale being composed from the former. When we go up or down the scales we find different structures for which other descriptions of the system are more applicable. A drastic case is that of going from a microscopical quantum mechanical description of a physical system to a description according to classical mechanics. The dynamics of quantum mechanical entities cannot always be described as deterministic, a system containing a large number of such entities like a macroscopical particle however shows classical time-reversible dynamics thus actually contradicting the former observation. Another example is that of the Ising model where the interaction of a large number of entities of binary state gives rise to rather complex order-disorder phenomena on the material scale that is easier described by continuous variables but its macroscopical state might as well be binary again (magnetization at $T = 0$) comparable to the phenomenon of a diverging correlation length in a Bose-Einstein condensate. The phenomenon of condensation as an example of order-disorder transitions can be found throughout all sciences and represents only one of the various phenomenons that are displayed by complex systems. Another, but much more accessible example is that of an ant-colony where, aside from the possible differentiation of workers, soldiers and a egg-laying mother of all, despite the absence of a global control entity the limited responses on external stimuli and local pair-communication give rise to impressive
collective behavior that makes it valid to describe the whole colony as one entity when it comes to building home structures, growing food, hunting behavior, migration, defense, and the interaction with other ant colonies. There are endless other examples of complex systems showing a tremendous ability of the small-scale entities to form structures on the larger-scale by purely local interaction and a limited set of states they can assume. The so-called emergence describes exactly this phenomena: the occurrence of a new quality operating at a higher level than at which the system units are operating. Besides the mentioned examples of quantum mechanical entities that form as a classical particle and the ant-colony relation, numerous other examples for this can be found: flying birds group into V-shaped formations optimizing their aerodynamic conditions, the global economy may display a long-lasting crisis as response to short-term actions on sub-markets, in a Belousov-Zhabotinsky reaction (and other excitable media) locally interacting entities with a resting-time dynamics give rise to waves and spirals of activation on the macro-scale, and the human body representing a whole cascade of such emergent scales from amino acids, DNA, cells, tissues, organs, the whole body, and beyond that societies, ecosystems, etc.

1.3 Interaction between Scales

If physics, biology, sociology, and other sciences too, are so successful in applying the concept of distinguishable structures on different scales to describe the system under study, can we bring the introductory questions to a more specific level and ask: How do those structures on different scales interact?

In some of these examples mentioned so far we find counteracting tendencies with different correlation/interaction range between the small scale entities themselves leading to some form of pattern formation on the larger scale as a consequence of collective behavior. This emergent behavior and other forms of self-organization can also be a response to external stimuli. The range of interactions between different structures on different scales is wide as structures can also be induced by external influences as boundary conditions, for example waves on the ocean that appear on different spatial scales, all of them induced by streams of air that not necessarily shows structures on the same spatio-temporal scale. Another example, where also a feedback can be detected are periodic flow structures in cardiovascular subsystems induced by the cardiac cycle setting the pressure boundaries which than, in turn, act back to the heart which, as a regulatory system, adjusts the pressure to the flow resistance to maintain constant blood flux through the body. When Darwin published his theory of the evolution of the species on earth, for the first time a reasonable approach to understand the interrelation between biological creatures on the time scale of a life span and the dynamics of a whole species on a large time scale has been offered taking spatial boundaries into account. When Hodgkin and Huxley drafted their model of ion channels in neuronal dendrites, fast local processes on the molecular level could be related to the formation and dynamics of action potentials traveling on a much larger spatio-temporal scale. In turn, action potentials, in an abstract form, are the basic entities of information which can be related to a global computational capacity by artificial neuronal network models. For the example of a QM description for the structures on the micro-scale and the classical object on the macro-scale given earlier, laws of classical mechanics can be
derived from the laws of quantum mechanics at the limit of large systems or large quantum numbers (correspondence principle). We thus can apply a mathematical (statistical) model to couple these two structures in an analytical way. Considering the simple example of ideal gas particles, statistical physics gives us means to extract the properties of a macroscopic system of such particles. Another rare example for an analytical solution is the thermodynamics of the Ising model solved by Onsager. In most cases numerical methods have to be used, e.g. Molecular Dynamics (MD) to connect the particle level with the phenomena on the macroscopic level where the system can be described with the help of quantities that do not take individual particles into account. Information about individual particles is transformed into information on structures at an other scale. And, structures on the macro-scale act back on individual particles, e.g. in the form of external fields etc. If they are able to consider as many entities and as large time spans to reveal the emergent spatio-temporal structures, mathematically formulated theories and any other type of models like a CA help us in understanding the correlation between two scales: the entity scale and the scale of the emergent structures. Of course, solutions should be available in a reasonable computation time to allow models to make sense as a tool to study a phenomenon.

1.4 Multiple Scales

The multi-scale aspect of nature is a recognized fact reflected by the numerous publications under this topic in the last two decades. Especially in the field of physiology which covers a vast amount of scales from DNA to health the need for a better understanding of multi-scale interaction between multi-science sub-systems has been recognized. However, seldom systems consisting of more than two processes have been considered so far. Some phenomena, and in principle all of them if we want to study them on a higher level of realism, are the consequence of a number of processes that act on very different spatio-temporal scales and interact not only via a single micro-macro type of coupling in the same domain like a MD process giving local quantities on the Navier-Stokes level describing the same gas or fluid.

For example, to implement and test the theory of Darwin in a model that is able to compute the outcome of the evolution of the full range of species on earth to understand the fauna and flora as we know it now starting from an initial condition set billions of years ago, it would be necessary to incorporate not only the behavioral dynamics of individuals and the macroscopic behavior of a species. Also geological processes, the whole weather system from the evaporation of water from the surface of certain types of leaves to the global temperature changes, the change of the intensity of solar radiation, the dynamics of extraterrestrial bodies, the mutation process in relation to the presence of chemicals or radiation - the whole range of physics, chemistry and biology - would have to be taken into account. It is clear that we won’t be able to do that in the foreseeable future.

It is not only the computational power that would limit such huge kind of projects. The computational complexity of models could always be reduced by orders of magnitude by some smarter more advanced algorithm. It is therefore useful to investigate how processes on different scales interact to be able to reduce the information processed between the scales to a minimum that still carries all
relevant aspects of the interaction. In the way scientists have studied the interaction of entities on one scale so far and elaborated useful classifications of all possible interactions where much insight in universality of could be gained, the study of interactions between processes on different scales and a systematic classifications of them is promising in gaining insight in the multi-scale aspect of nature. Similarly to how we study the interaction of particles by their position $x_i$ at a certain time $t_i$ to model the interaction of a large number of them in a macroscopic system we can also identify whole processes occupying a distinct volume in this spatio-temporal space. Doing so, either an overlap (sharing the same domain), shared boundary (spatial interface, temporal successiveness) or separation of the two processes in either or both space and time can be found. Although this sounds trivial we can also apply this concept of classification in a space spanned by the spatial scales $\xi_i$ and temporal scales $\tau_i$ where the processes occupy a certain volume defined by the spatio-temporal scales of its entities and that of the emergent structures. The identification of overlap, shared boundary, or full separation can be repeated in such a space. Many types of single scale systems have been studied so far and the study of the coupling between two scales has begun (most of which are of the typical micro-macro coupling). However no systematic classification of coupling types has been done yet considering all possible relative positions in such a space with the aim to recognize similarities/universalities or fundamental differences in multi-scale multi-science interactions.

We are only at the beginning of understanding of how information is processed over the scales. We cannot yet treat information in the strict sense of the word as defined in information theory. We first need a common concept when it comes to multi-scale modeling. And we need many examples of implementations of multi-scale systems that are studied in the view of this concept. This thesis is meant to be a contribution to satisfy this need. Within the COAST project we had the chance to closely collaborate with bio-medical scientists to work on modeling of in-stent restenosis (ISR), a serious cardiovascular disease involving processes on multiple scales and that poses a great challenge to understand. Originating from initial plans to resolve the blood flow in an ISR model on the cellular level, we worked on modeling suspensions with the Lattice-Boltzmann method. However, to understand the multi-scale nature of blood rheology itself we realized that it was necessary to study the simplest suspension system, monodisperse suspensions of hard spheres in two dimensions, in more detail first. It turned out that suspension rheology and its dependence on the involved scales is very complex in itself requiring knowledge on the very details of the involved interactions. Additionally, it takes substantial work to develop numerical methods that allow such a study. Weinan E phrased it this way [7]:

"We should emphasize that HMM\textsuperscript{1} is not a specific method, it is a framework for designing methods. For any particular problem, there is usually a considerable amount of work, such as designing the constrained microscopic solvers, that is necessary in order to turn HMM into a specific numerical method."

Consequently, the work described in this thesis involves a large part on the method-

\textsuperscript{1}Heterogeneous Multi-scale Method, the multi-scale approach used to model the suspension flow in this work.
ological level, particularly in improving Lattice-Boltzmann method for suspension flow.

1.5 Outline of this Thesis

Addressing the need of a classification of multi-scale systems and their internal coupling, in chapter 2 we motivate and propose an appropriate methodology to classify multi-scale problems based on the concept of CA’s. In this theory of Complex Automata (CxA) we further elaborate the idea of a scale separation map sketched in the last paragraph and identify possible relative positions of two processes on it and discuss the consequences for their coupling. We also classify multi-scale coupling in regard to the relative position of two processes in space which leads to a full classification of multi-scale problems that can be reduced to two single-scale processes. Based on this full classification we discuss possible modeling strategies resulting from it. However, we do not leave it at describing the consequences of the classification in a general language. We also describe how this can be implemented in a simulation framework that allows researchers to couple single-scale models in a generic way by so-called smart conduits that are the software realizations of the coupling templates elaborated in this chapter. We also discuss the reduction of computational effort in applying the concept of a CxA that makes it very attractive to be applied to complex systems which could not be studied so far due to their computational complexity. Besides that, an estimation of the error that might be introduced by splitting a multi-scale system into sub-systems and replacing their coupling by one of the coupling templates is provided.

In chapter 3, as a first example of a multi-scale multi-science system, a model for the biomedical problem of in-stent restenosis (ISR) is described. Making use of the MUSCLE library the model was developed to study the influence of blood flow properties on the process. The involved subprocesses are identified on the scale map together with the type of their coupling. The implemented submodels will be described in detail, particularly that for the formation of thrombus in the early stages of ISR. Existing approaches to thrombus modeling will be discussed and a new simplified thrombus model will be proposed. At the end of this chapter results from runs of the 2D and 3D ISR CxA’s are discussed.

Chapter 4 describes a second multi-scale system: the macroscopic flow of non-Brownian hard-sphere suspensions. The macroscopically emergent rheology is dictated by details of fluid-particle, and particle-particle interactions. In systems where the typical spatial scale on the particle level is much smaller than that of macroscopic properties the scales can be split using a hierarchical multi-scale method (HMM) approach. In that, on the macro-scale the suspension is treated as a non-Newtonian fluid whereas local properties of macroscopic fields are input to a fully resolved suspension simulation. Down- and upward mapping of viscosity and diffusivity related quantities will be discussed. The coupling of the sub-models has not been implemented by means of MUSCLE. However, the coupling is discussed in the light of the CxA theory.

Developing concepts and models on a rather high level is one thing, realizing a model down to the very detail of the methods another. Consequently the research presented in this thesis addresses the improvement of existing methods as well as the development of new ones necessary to realize the intended models.
In the HMM model of suspension flow the lattice-Boltzmann method (LBM) is applied to model the fluid phase on both levels. In this thesis two major improvements in simulating fully resolved suspensions with the help of LBM are proposed. First, Lees-Edwards boundary conditions (LEbc) are developed to remove the side-effects that walls have in a Couette type of shear flow on the microstructure. It can be shown that using LEbc’s the shear-thickening behavior of hard-sphere suspensions is prolonged in comparison to Couette-flow results. Second, a correction to the momentum exchange algorithm (MEA) used in commonly applied LBM suspension methods is proposed that restores Galileian invariance of the particle dynamics. Furthermore, a database has been developed including inter- and extrapolation functionalities that effectively reduces the redundancy in executing the micro-model the way initially described in the HMM model. The reduction of the suspension’s local properties to a mean solid-fluid volume fraction, and a scalar shear rate under exploitation of rotational and Galileian invariance maps many different local states of the macroscopical suspension to one realization of boundary conditions on the micro-scale.

In chapter 5 the improved LBM methods are then used to investigate the behavior of the spatial scale in hard-sphere suspensions under increasing shear for different solid-fluid volume ratios. Typically at higher shear rates hydrodynamically interacting particles form clusters which give rise to shear-thickening. Cluster size distributions are measured and compared with experimental findings and theoretical models. A scaling of the apparent viscosity with the typical cluster size can be found supporting the assumption that clustering is the prominent process leading to shear-thickening. A statistical model for the development of cluster size distributions in dependence on \( \dot{\gamma}, \phi \), and solid-fluid mass density ratio is proposed. Based on the findings from a two-particle collision model the formation and breaking of particle clusters are modeled and relax to cluster size distributions that are in very good agreement with measurements in fully resolved LBM suspensions. Although this study has not been carried out in relation to the concepts of CxA’s, it returns to the motivation presented in the introduction chapter by presenting a study of a system in which new scales and objects on it emerge from the interaction of objects on a smaller scale. This study offered insight into the rheology of sheared suspensions absolutely necessary to realize the HMM suspension model in the previous chapter.
Chapter 2

Complex Automata for multi-scale modeling

2.1 Multi-scale Modeling

2.1.1 Introduction

Cellular Automata (CA) are generally acknowledged to be a powerful way to describe and model natural phenomena [1, 8, 9]. There are even tempting claims that nature itself is one big (quantum) information processing system, e.g. [10], and that CA may actually be nature’s way to do this processing [11, 3, 5]. We will not embark on this philosophical road, but ask ourselves a more mundane question. Can we use CA to model the inherently multi-scale processes in nature and use these models for efficient simulations on digital computers? The ever increasing availability of experimental data on every scale, from ‘atom to material’ or from ‘gene to health’, in combination with the likewise ever increasing computational power [12, 13], facilitate the modeling and simulation of natural phenomena taking into account all the required spatial and temporal scales (see e.g. [14]). Multi-scale modeling and simulation, as a paradigm in Computational Science, is becoming more and more important, as witnessed by e.g. dedicated special issues [15] and thematic journals [16, 17]. Consider for example the field of physiology. The sequence from the genome, proteome, metabolome, physiome to health comprises multi-scale, multi-science systems [18, 19]. Studying biological sub-systems, their organization, and their mutual interactions, through an interplay between laboratory experiments and modeling and simulation, should lead to an understanding of biological function and to a prediction of the effects of perturbations (e.g. genetic mutations or presence of drugs) [20]. The concept ‘from genes to health’ is the vision of the Physiome [21] and ViroLab [22] projects, where multi-scale modeling and simulation of aspects of human physiology is the ultimate goal. Modeling such systems is a challenging problem but has the potential to improve our understanding of key interactions. The inherent complexity of biomedical systems is

---

now beginning to be appreciated fully; they are multi-scale, multi-science systems, covering a range of phenomena from molecular and cellular biology, via physics and medicine, to engineering and crossing many orders of magnitude with regard to temporal and spatial scales [23]. Despite the widely acknowledged need for multi-scale modeling and simulation, there is a scarcity of underpinning literature on methodology and generic description of the process. There are many excellent papers that present multi-scale models, but few methodological papers on multi-scale modeling (such as [24, 7]) have appeared. When using Cellular Automata to model a natural process, the lattice spacing and time step have a clear meaning in relation to the corresponding physical space and time of the process. We denote by $A(\Delta x, \Delta t, L, T)$ the spatio-temporal domain of a CA, whose spatial domain is made of cells of size $\Delta x$ and it spans a region of size $L$, while the quantity $\Delta t$ is the time step and $T$ is the end of the simulated time interval. Therefore, processes with time scales between $\Delta t$ and $T$ can be represented and spatial scales ranging from $\Delta x$ to $L$ can be resolved. When executing such CA on a digital computer we note that the execution time $T_{\text{ex}}$ scales as

$$T_{\text{ex}} \sim \frac{T}{\Delta t} \left( \frac{L}{\Delta x} \right)^D,$$

where $D$ is the spatial dimension of the simulated domain. Trying to model a multi-scale system with a single CA would require to choose $\Delta x$ and $\Delta t$ in such a way that the smallest microscopic details and fastest dynamical response of the system are captured, yet the overall system size ($L$) and slowest dynamical time scale ($T$) need to be covered. For instance, in modeling human physiology the relevant range of spatial scales is from nanometer to meter (i.e. a factor $10^9$) whereas temporal scale is from microseconds to human lifetime (i.e a factor $10^{15}$). These numbers, in combination with (2.1) immediately show that one will probably never be able to simulate multi-scale systems with a single CA spanning such a wide range of scales.

The literature on using Cellular Automata to model multi-scale phenomena is relatively small, maybe with the exception of using CA to model land usage and geographical systems (e.g. [25]). Furthermore, many papers exist that use CA in multi-scale modeling, but there CA is typically coupled to other types of models (e.g. [26]). The bulk of CA multi-scale attempts are grid refinement methods, also termed multi-blocks. The idea is to adapt the local grid size to the local process scale, i.e. using a fine grid in regions where small scale processes occur and a coarse grid where larger scales are sufficient. A common approach is to couple grids of different scales with an overlap region [27].

Other ways of coupling multi-scale CA come from two theoretical frameworks. The first one is based on higher-order CA [28]. In this framework, the CA rules are not only able to change the cell state, but also the rules themselves, the neighborhood and the topology. Moreover, these models are also able to take into account hierarchical CA where higher level cells are connected to one or more lower level cells. The second one results from the work of Israeli and Goldenfeld [29] who have shown that it is possible to coarse-grain 1D nearest-neighbor CA, by defining a macroscopic CA whose behavior is similar to a microscopic CA. That is an important result because the authors have achieved the coarse-graining of CA known to be irreducible.
We developed a multi-scale, multi-science framework, coined Complex Automata (CxA), for modeling and simulation of multi-scale complex systems [30, 31]. The key idea is that a multi-scale system can be decomposed into N single-scale CA that mutually interact across the scales\(^2\). Decomposition is facilitated by building a Scale Separation Map (SSM) on which each single-scale system can be represented as an area according to its spatial and temporal scales. Processes having well-separated scales are easily identified as the components of the multi-scale model. We validate the CxA approach by building a large set of exemplary applications, and applying it to the challenging clinical problem of coronary artery in-stent restenosis (ISR) [32]. The CxA approach was developed within the context of the Coast project [33].

In this chapter we will review the current state of development of Complex Automata and explore the possibilities that are offered by Cellular Automata (CA) for multi-scale Modeling and Simulation.

### 2.2 Complex Automata

#### 2.2.1 A definition

Formally, we shall define a CA as a tuple

\[
C = \{ A(\Delta x, L, \Delta t, T), F, \Phi, f_{init} \in F, u, O \}. \tag{2.2}
\]

\(A\) is the domain, made of spatial cells of size \(\Delta x\) and spanning a region of size \(L\), while the quantity \(\Delta t\) is the time step and \(T/\Delta t\) is the number of iterations during which the CA will be run. Therefore, processes with time scales between \(\Delta t\) and \(T\) can be represented and spatial scales ranging from \(\Delta x\) to \(L\) can be resolved. The state of the CA is described by an element of \(F\) (space of states) and it evolves according to the update rule \(\Phi\) : \(F \rightarrow F\) (note that formally both \(F\) and \(\Phi\) depend on the discretizations \((\Delta x, \Delta t))\). Additionally, we constrain the update rule to be in the form of \(\text{collision} + \text{propagation}\), such that the operator \(\Phi\) can be decomposed as a

\[
\Phi = \text{PCB}, \tag{2.3}
\]

i.e. into a boundary condition, a propagation, and a collision operator, each depending, possibly, on the field \(u\) (see also section 2.2.3 for more details). The terminology collision-propagation is borrowed from the lattice gas automata framework (see e.g. [1]). This is equivalent to the more classical Gather-Update CA paradigm, as was formally demonstrated recently [34]. The initial condition \((f_{init})\) is a particular element of the space of states. At the spatial boundaries of \(A\), additional information is needed (boundary conditions).

In definition 2.2, we introduced additional elements. The field \(u\) collects the external information exchanged at each iteration between the CA and its environment. The functional \(O : F \rightarrow \mathbb{R}^d\), the observable, specifies the quantity we are interested in.

\(^2\)Note that our approach is not limited to CA but also includes extensions such as lattice Boltzmann models and agent based models, because they can all be described by a generic update rule discussed in section 2.2.3.
A CxA can be viewed as a collection of interacting CA. Definition 2.2 suggests that a CxA can be represented as a graph $\mathcal{X} = (V, E)$ where $V$ is the set of vertexes and $E$ the set of edges with the following properties

- Each vertex is a CA $C_i = \{ A_i(\Delta x_i, L_i, \Delta t_i, T_i), F_i, \Phi_i, f_{\text{init},i}, u_i, O_i \}$
- each edge $E_{ij}$ is a coupling procedure describing the interaction between $C_i$ and $C_j$. In practice, $E_{ij}$ will define how and when information is exchanged between the two subsystems.

During the initialization phase, this problem-dependent graph is built according to the modeler’s specifications.

### 2.2.2 The Scale Separation Map

A key idea behind CxA is that a multi-scale system can be decomposed into $N$ single-scale Cellular Automata that mutually interact across the scales. The decomposition is achieved by building a Scale Separation Map (SSM) on which each system can be represented as an area according to its spatial and temporal scales. Processes having well separated scales are easily identified as the components of the multi-scale model.

Fig. 2.1 shows a SSM, where the horizontal axis represents the temporal scales and the vertical axis the spatial scales. On the left a CA with spatio-temporal domain $A(\Delta x, \Delta t, L, T)$ is represented on the SSM. Assuming that the process to be simulated is really multi-scale in the sense that it contains relevant sub-processes on a wide range of scales, simulations based on the finest discretizations are not really feasible (recall (2.1)), the approach we propose in CxA modeling is to try to split the original CA into a number of single-scale CA and let these CA exchange information in such a way that the dynamical behavior of the multi-scale process is mimicked as accurately as possible. This is shown schematically in the right part in Fig. 2.1. The subsystem in the lower left part operates on small spatial scales, and short time scales, the one at the upper right part operates at large scales, and the other three at intermediate scales. This could e.g. be processes operating at the micro-, meso-, and macro-scale.
After identifying all subsystems and placing them on the scale map, coupling between subsystems is then represented by edges on the map. For instance, a process can be coupled with another through a lumped parameter or through detailed spatially and temporally resolved signals, in which case they would typically share a boundary and synchronously exchange information. The distance between subsystems on the map indicates which model embedding method to use to simulate the overall system. In the worst case, one is forced to use the smallest scales everywhere, probably resulting in intractable simulations. On the other hand, if the subsystems are well separated and the smallest scale subsystems are in quasi-equilibrium, then they can be solved separately, although infrequent (possibly event-driven) feedback between the subsystems will still be required.

Consider two processes A and B with their own specific spatial - and temporal scale, denoted by $\xi_i$ and $\tau_i$ respectively ($i \in \{A, B\}$). Assume that A has the largest spatial scale. In case the spatial scales are the same, A has the largest temporal scale. In other words, ($\xi_B < \xi_A$) OR ($\xi_B = \xi_A$ AND $\tau_B < \tau_A$). We can now place A on the scale map and then investigate the different possibilities of placing B on the map relative to A. This will lead to a classification of types of multi-scale coupling, as in Fig. 2.2.

Depending on where B is, we find the following regions:

- **B in region 0**: A and B overlap, so we do not have a scale separation, we are dealing here with a single-scale multi-science model.

- **B in region 1**: Here $\xi_B = \xi_A$ AND $\tau_B < \tau_A$, so we observe a separation of time scales at the same spatial scale.

- **B in region 2**: Here $\xi_B < \xi_A$ AND $\tau_B = \tau_A$, so we observe a separation in spatial scales, like coarse and fine structures on the same temporal scale.

- **B in region 3**: Separation in time - and spatial scales. Region 3.1 is the well-known micro ⇔ macro coupling, so fast processes on a small spatial scale coupled to slow processes on a large spatial scale. This type of multi-scale model has received most attention in the literature. In region 3.2 we have the reversed situation, a slow process on small spatial scales coupled to a fast process on large spatial scales. We believe that
this region is very relevant in for instance coupling of biological with physical processes, where the biological process is e.g. the slow response of cells to a faster physical process on a larger scale (e.g. blood flow in arteries).

Note that we do not have to consider other regions of the scale map, because then the role of \( A \) and \( B \) just reverses, and we fall back to one of the five cases identified above.

Next we address the question of the area that processes \( A \) and \( B \) occupy on the SSM. As discussed earlier, a 1D CA is characterized by a spatial discretization \( \Delta x \) and a system size \( L \). We assume that \( \Delta x \) and \( L \) have been chosen such that \( \Delta x < \xi < L \). We define \( N^{(x)} \) as the number of CA cells that extend the full domain, i.e. \( N^{(x)} = L/\Delta x \). Next assume that the discretization has been chosen such that the spatial scale is represented by \( 10^\delta^{(x)} \) cells (i.e. \( \Delta x = \xi/10^\delta^{(x)} \)) and the spatial extension of the CA is \( 10^{\eta^{(x)}} \) times the spatial scale, i.e. \( L = \xi10^{\eta^{(x)}} \), and therefore \( N^{(x)} = 10^{\eta^{(x)}+\delta^{(x)}} \). Likewise for the temporal domain, i.e. a single scale CA has a time step \( \Delta t \) and the CA is simulated over a time span \( T \), and we have \( \Delta t < \tau < T \). The number of time steps \( N^{(t)} = T/\Delta t \). The discretization has been chosen such that the temporal scale is represented by \( 10^\delta^{(t)} \) time steps (i.e. \( \Delta t = \tau/10^\delta^{(t)} \)) and that simulation time of the CA is \( 10^\eta^{(t)} \) times the temporal scale, i.e. \( T = \tau10^{\eta^{(t)}} \) and \( N^{(t)} = 10^{\eta^{(t)}+\delta^{(t)}} \).

A process’ position on the scale map is now fully determined by the tuple \( \{\xi, \delta^{(x)}, \eta^{(x)}; \tau, \delta^{(t)}, \eta^{(t)}\} \), and is drawn in Fig. 2.3, where the axes are now on a logarithmic scale. On such logarithmic SSM the process is rectangular with an area \( (\delta^{(t)} + \eta^{(t)}) \times (\delta^{(x)} + \eta^{(x)}) \) asymmetrically centered around the point \( (\log(\tau), \log(\xi)) \). In the special case that \( \delta^{(x)} = \eta^{(x)} = \delta^{(t)} = \eta^{(t)} = 1 \) (a reasonable first order assumption) we see that the process is symmetrically centered around \( (\log(\tau), \log(\xi)) \) and that the size of the box extends 2 decades in each dimension.

In Fig. 2.3 we show the extension of Fig. 2.2, where regions 1 – 3 now have well
defined positions and size. Depending on the location of process B, that is the point \((\log(\tau_B), \log(\xi_B))\) on the SSM, and with all information on the spatial and temporal extensions of processes A and B, we can unambiguously find in which region of the scale map they are located with respect to each other. The scale separation between two processes can now clearly be defined in terms of a distance on the SSM, and this can then become an important measure to determine errors that are induced by scale splitting procedures. This is further elaborated in Section 2.3.

Consider once more region 3, where there is a separation in time and length scales. In region 3.1 we find that \(L_B < \Delta x_A\) and \(T_B < \Delta t_A\). As said earlier, this is the classical micro ⇔ macro coupling, and in our language this means the full spatio-temporal extend \(T_B \times L_B\) of process B is smaller than one single spatio-temporal step \(\Delta t_A \times \Delta x_A\) of process A. A number of modeling and simulation paradigms have been developed for this type of multi-scale systems (see e.g. [7]).

Region 3.2 also exhibits separation of time and length scales, but now the situation is quite different. We find that, just like in region 3.1, \(L_B < \Delta x_A\).

So, the spatial extend of process B is smaller than the grid spacing of process A. However, now we find that \(T_A < \Delta t_B\). In other words, the full time scale of process A is smaller than the time step in process B. This will result in other modeling and simulation paradigms than in region 3.1. Typically, the coupling between A and B will involve time averages of the dynamics of the fast process A.

Let us now turn our attention to the regions where there is overlap on the temporal - or spatial scales, or both (regions 0, 1, and 2, in Fig. 2.3). In all these cases we can argue that we have partial or full overlap of the scales, giving rise to different types of (multi-scale) modeling and simulation. We say that the scales fully overlap if the point \((\log(\tau_B), \log(\xi_B))\) falls within (one of) the scales spanned by process A. On the other hand, there is partial overlap if \((\log(\tau_B), \log(\xi_B))\) falls outside (one of) the scales spanned by process A, but the rectangular area of process B still overlaps with (one of) the scales spanned by process A. The region of partial scale overlap can also be considered as a region of gradual scale separation, a boundary region between the scale separated regions 1, 2 and 3 and region 0. Simulations of this kind of multi-scale system would typically involve CxA’s with local grid refinements, or multiple time stepping approaches, or a combination of both.

### 2.2.3 The Sub-Model Execution Loop

A second important ingredient of the CxA formalism is the observation that each CA (i.e. vertex of the CxA) can be expressed with a common instruction flow. This gives a way to identify generic coupling templates and achieve a precise execution model (see also section 2.2.6). Using the specific collision+propagation form of the update rule, as introduced in section 2.2.1, we represent the workflow with a pseudo-code abstraction, termed the Sub-model Execution Loop (SEL), as shown below.
\[ D := D_{\text{init}} \quad /* \text{initialization of the domain} */ \]
\[ f := f_{\text{init}} \quad /* \text{initialization of state variables} */ \]
\[ t := 0 \quad /* \text{initialization of time} */ \]

While Not EC
\[ t += \Delta t \quad /* \text{increase time with one timestep} */ \]
\[ D := U(D) \quad /* \text{update the domain} */ \]
\[ f := B(f) \quad /* \text{apply boundary conditions} */ \]
\[ f := C(f) \quad /* \text{collision, update state of cells} */ \]
\[ f := P(f) \quad /* \text{propagation, sent information to neighbors} */ \]
\[ O_i(f) \quad /* \text{compute observables from new state} */ \]

End
\[ O_f(f) \quad /* \text{compute observables from final state} */ \]

Note that in the SEL, operators are written in bold and (state) variables as plain characters. The CA operates on a computing domain \( D \), being the lattice of cells and the boundaries. Each cell in a CA has a set of state variables \( f \). At the start of the SEL the domain and the state variables are initialized by the operators \( D_{\text{init}} \) and \( f_{\text{init}} \) respectively. The simulation time \( t \) is set to an initial value (0 in this case). After initialization the CA enters into an iteration loop, whose termination is controlled by an end condition computed by \( EC \). The end condition can simply be a fixed number of iterations, but could also be some convergence criterion depending upon the state variables. Within the main iteration loop, the time is first increased with a time step \( \Delta t \). Next the domain is updated by the operator \( U \). If the domain is static, this operator is just the identity operator \( I \). However, in many models the domain is dynamic. For instance, new cells can be created or existing cells removed (e.g. due to the movement of the boundary). In all these cases \( U \) will execute these domain updates. Next, the sequence \( PCB(f) \) is executed. First, the operator \( B \) applies the boundary conditions. This means that missing information is constructed that is needed for the actual state updates by \( C \) (see below) of the cells lying at the boundary of the domain \( D \). For instance, if the state variables represent a concentration of some species, the boundary condition could specify a flux of those species into the domain, and from that missing information on the domain boundary cells is computed. Next the actual state change of all cells is computed by the Collision operator \( C \). Finally, information is sent to neighboring cells or agents by the Propagation operator \( P \). The CA is now updated for the current time step, and the simulation can proceed to the next iteration. However, before doing so an intermediate observation operator \( O_i \) computes observables from the state variables \( f \). After termination of the main iteration loop a final observation is done of the state variables with the \( O_f \) operator.

### 2.2.4 CxA Multi-Scale Coupling

Despite the growing literature there is not a well accepted generic methodology, nor a well-defined nomenclature of multi-scale modeling. A few authors have proposed different typologies of multi-scale models. Weinan E et al. [7] have proposed 4 types of multi-scale problems and 4 general strategies. Despite the many examples given by them the relevance of their classification is not always clear, because they
Figure 2.4: Overlap or separation on spatial and temporal scales resulting from the relative positions of sub-processes leading on a SSM.

single out, in all their examples, one specific item from their classification, and do not further discuss the relevance or completeness of the other classes. Another proposition for a multi-scale modeling methodology is that of Ingram. Working on chemical engineering simulations, Ingram et al. [24] have defined five types of macro-micro scale coupling. Ingram et al. present simulation examples for three types of coupling, showing that different strategies may be used to solve the same problem. The choice of coupling has an influence on both computational efficiency and accuracy. The fact that it is not always easy or possible to make the correspondence between the approaches by Ingram et al. and E et al. indicates that the topic of multi-scale modeling lacks consensus. This lack of consensus on terminology and methodology can be attributed to the fact that actual coupling methodologies were mixed with classifications of the computational domain and/or with the type of scale separation (temporal, spatial, or both).

In the following discussion we try to clarify the situation, in the framework of the CxA formalism. However, we believe that this is also relevant to multi-scale modeling in general. Based on the discussion on the SSM in section 2.2.2, we identified 5 different types of scale separation. We call them Interaction Regions on the SSM, and they are shown in Fig. 2.4. Another important parameter to distinguish multi-scale models is the Domain type. We distinguish between single Domain (sD) and multi-Domain (mD) types. In case of sD processes A and B can access the whole simulated domain and communication can occur everywhere, whereas in case of mD each process is restricted to a different physical region and communication can only occur across an interface or small overlap region.

For each combination of interaction region and domain type we can now try to identify a multi-scale coupling. We will base our approach on the SEL discussed in section 2.2.3, and show which operators from the SEL are coupled to each other. We call this Coupling Templates. As an example consider Weinan E’s Heterogeneous Multi-scale Method [7]. On close inspection we must conclude that this is a Coupling Template for single Domain processes in interaction region 3.1. In terms of the SEL of the macroscopic process A and the microscopic process B we find as Coupling Template \( O_f^B \rightarrow C^A \); \( O_i^A \rightarrow f^B_{\text{init}} \) (see also Fig. 2.8). At each time step of the macroscopic process B a microscopic process A is initialized using macroscopic information. The microscopic model then runs to completion and sends final information to the collision operator of the macroscopic process.

We are currently investigating many examples of multi-scale models, their map-
Figure 2.5: Our classification of multi-scale problems, for systems that can be reduced of two single-scale processes. This classification is based on the five interaction regions given by the SSM, and the domain type (sD or mD). For each class, the generic coupling template is indicated, in terms of the CxA operators. Examples of specific applications belonging to the given categories are indicated in italic.

<table>
<thead>
<tr>
<th>Time Overlap</th>
<th>Time Separation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single Domain</td>
<td>Multi Domain</td>
</tr>
<tr>
<td>Coupling through collision operator.</td>
<td>Coupling through boundary condition.</td>
</tr>
<tr>
<td>Snow transport, diffusion/ advection, ...</td>
<td>Fluid structure, grid refinement, ...</td>
</tr>
<tr>
<td>Space Overlap</td>
<td>Space Separation</td>
</tr>
<tr>
<td>Single Domain</td>
<td>Multi Domain</td>
</tr>
<tr>
<td>Coupling through collision operator.</td>
<td>Coupling through boundary condition.</td>
</tr>
<tr>
<td>Wave propagation in two media, ...</td>
<td>&quot;Physics-Biology Coupling&quot;</td>
</tr>
<tr>
<td>Hierarchical Coupling</td>
<td>Coupling through collision operator and initialization.</td>
</tr>
<tr>
<td>''Oscillating blood flow and endothelial cells, ...&quot;</td>
<td>Wave propagation in two media, ...</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Single Domain</th>
<th>Multi Domain</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coupling through collision operator.</td>
<td>Coupling through boundary condition.</td>
</tr>
<tr>
<td>Forest-Savannah-Fire interactions</td>
<td>Coral Growth, ...</td>
</tr>
</tbody>
</table>

In the case of time scale overlap, the coupling will occur inside the inner iteration loop. In contrast, in the case of time scale separation, coupling is realized outside the inner loop through the initialization operators and the final observation operator.

- Single-domain models are coupled through the collision operator. Multi-domain models are coupled through the domain update or the boundary operators.

Based on our current set of examples, we hypothesize that for each type of multi-scale model, classified in terms of domain type and interaction region, only a very small set of coupling templates exists. If this is true, this would lead the way to a powerful CxA multi-scale modeling and simulation strategy, including a multi-scale modeling language, generic simulation software and a mathematical framework to analyze errors involved in CxA modeling. In what follows we will further elaborate on these ideas, sketching the contours of such a generic CxA based multi-scale modeling approach.

### 2.2.5 Multi-scale modeling strategies

A key question when dealing with a multi-scale system is how to decompose it in several coupled single-scale sub-processes. This decomposition is certainly not
unique and a good knowledge of the system may be required. Once the sub-processes are chosen, this specifies the relation between the computational domains and the interaction regions on the SSM. Then, our classification scheme indicates the expected coupling templates.

We have observed several strategies that can be used to deal with systems having a broad range of scales and to reduce their area on the scale separation map. They are briefly discussed below.

**Time splitting:** This approach is appropriate when two processes act at different time scales. Let us assume we have a sD problem described with a propagation operator $P$ and a collision operator $C$ that is the product of two operators

$$P_{\Delta t}C_{\Delta t} = P_{\Delta t}C^{(1)}_{\Delta t}C^{(2)}_{\Delta t} \quad (2.4)$$

where $\Delta t$ specifies the finer scale of the process. Then, if $C^{(1)}_{\Delta t}$ acts at a longer time scale than $C^{(2)}_{\Delta t}$ we can approximate $M$ iterations of the dynamics as

$$[P_{\Delta t}C_{\Delta t}]^M \approx P_{M\Delta t}C^{(1)}_{M\Delta t}[C^{(2)}_{\Delta t}]^M \quad (2.5)$$

We will illustrate this time-splitting strategy in detail in section 2.3.

**Coarse graining:** The goal of coarse graining is to express the dynamics of a given system at a larger temporal and/or spatial scale in some part of the computational domain where less accuracy is needed. After coarse graining we obtain a new process, specified by new collision and propagation operators and occupying a reduced area on the SSM. Within our formalism, a space-time coarse graining of a factor 2 can be expressed as

$$[P_{\Delta x}C_{\Delta x}]^n \approx \Gamma^{-1}[P_{2\Delta x}C_{2\Delta x}]^{n/2}\Gamma \quad (2.6)$$

where $\Gamma$ is a projection operator mapping the system state, originally defined on a fine scale $\Delta x$, to a scale $2\Delta x$. The exponent $n$ is the number of iterations needed to simulate the problem.

**Amplification:** This strategy can be used to reduce the larger time scale of a process. For instance, we can consider a process acting with low intensity but for a long time, in a time periodic environment, such as a growth process in a pulsatile flow.

Within our formalism, let us consider two coupled (mD) processes which are iterated $n \gg 1$ times

$$[P^{(1)}C^{(1)}]^n \quad \text{and} \quad [P^{(2)}C^{(2)}(k')]^n \quad (2.7)$$

where $k'$ expresses the intensity of the coupling of process 1 to process 2.

If the $C^{(1)}$ is periodic with period $m \ll n$, we can approximate the above evolution as

$$[P^{(1)}C^{(1)}]^m \quad \text{and} \quad [P^{(2)}C^{(2)}(k')]^m \quad (2.8)$$

with $k'$ the new effective intensity of the coupling. For a linear coupling we would have $k' = (n/m)k$. 
Figure 2.6: Coupling template for the so-called coral growth model. Numbers corresponds to the communication operation described in Fig. 2.7.

### 2.2.6 Execution Model

Coupling several sub-models, using coupling templates raises implementation issues. A typical situation is shown in Fig. 2.6 for the problem of coral growth. The growth of branching corals is modeled with the aim to understand the influence of abiotic factors (transport of nutrients by flow and diffusion) on the morphology. This is work performed under the supervision of Dr. Jaap Kaandorp, and for biological context and background we refer to [35], his recent book [36] and to [37, 38].

In short, this model works as follows: the fluid flow is transporting nutrients that are needed by the coral to grow. There is a clear time scale separation that can be exploited. Fluid flow establishes at a few seconds whereas the coral grows at a much slower pace. According to the coupling template shown in Fig. 2.6, the fluid solver is run until steady state and the resulting flow field is passed to the coral solver for calculating the growth rate. The new geometry of the coral is then used to build a new initial condition for the flow solver. The process stops when enough iterations of the coral solver have been performed.

Using this example we will explain the main concepts of our proposed execution model for CxA, which is compatible with the asynchronous channel actor-model framework [39]. A computer implementation for a CxA simulation environment, implementing this execution model, has been realized [40] and a public domain release is available.

### CxA components

For the sake of the present discussion, CxA can be described as directed bipartite graphs whose edges represent a single direction communication channel and the vertexes are either kernels or conduits. The kernels are the main computational units of a CxA. Generally, kernels are the single-scale sub-model solvers as described above. However, when needed, they can also execute other tasks such as measurements or complex data mappings. The conduits are ”smart” communication channels. Each conduit connects a pair of kernels together in an oriented fashion and, in principle, only one quantity is transported per conduit. These conduits are composed of three parts:

1. an incoming buffer (the entrance)

---

3[^1]: see http://www.complex-automata.org
(2) an outgoing buffer (the exit)
(3) (optional) one or several data filters between different scales (to perform interpolation, restriction, discretization, etc.)

Conduits work in a purely reactive way: when data is copied at the entrance, the conduit applies the filters and moves the resulting data into the outgoing buffer. Each conduit is connected to only two kernels, but kernels can be connected to an arbitrary number of conduits. Each component is either a full process or a thread depending on the implementation. They can reside in the same machine or be distributed across a network.

**CxA Communication**

In CxA, kernels communicate exclusively via conduits, using a message passing paradigm. Only two communication primitives are defined to interact with conduits:

1. `send(data)`: this primitive sends a data vector from a kernel to a conduit entrance. It is non-blocking, since it returns as soon as the data is sent to the conduit, whether or not the destination process has read the data. This corresponds to a push communication.

2. `receive()`: this primitive allows a kernel to receive data from a conduit exit. This primitive is blocking, it will return only when the desired data exist in the conduit. The receiving kernel will then simply wait until the data is available before resuming its computations. This corresponds to a pull communication.

Conduits entrances and exits are supposed to have large buffers, able to store several large data structures. These buffers act as FIFO ("first in, first out") where each entry is a reference to a data-structure. So, if the sending kernel is faster than the receiving one, several data vectors will be stored in the exit buffer, waiting for a `receive()` call from the destination kernel. The FIFO nature of the buffer ensures that the data are always read in the correct time order. The actual communication can be either a memory copy if the kernel and conduit reside in the same processor, or a network communication if both components reside on different machines. Note that the conduit could also be used to implement a mutex coordination primitive in case of shared memory execution.

Let us consider again the example of the coral growth. The coral SEL represented in Fig. 2.6 can be rewritten as follows, to include the two communication primitives explicitly:

```
While Not EC
  D := U(D)
  DomainConduit.send( D )
  f := B(f)
  velocityMap := VelocityConduit.receive()
  f := C(f, velocityMap)
  f := P(f)
End
```
**CxA Initialization and start**

CxA initialization occurs in a semi-decentralized way. First, each conduit and kernel is spawned (possibly on several machines). Then a special process, termed *plumber*, is responsible for connecting each kernel with the entrances and exits of the relevant conduits. The plumber terminates as soon as this basic task is finished. The rest of the initialization process is then fully decentralized:

1. As soon as a kernel is fully connected with the required conduits, it starts its computations. If it is sending data to a yet unconnected kernel, the data will be kept in the conduit until the receiver is active and reading. On the other hand, if a conduit tries to receive data originating from an unconnected kernel, it will hang on until the sending kernel connects and transmits data.

2. For conduits the situation is even simpler. Since they are purely reactive components, nothing will happen in an unconnected conduit. Similarly, if only the conduit exit is connected, the conduit will do nothing. In contrast, if only the conduit entrance is connected, the conduit will simply process incoming data which will be accumulated in the exit buffer. Therefore, the conduit is always in a valid state (assuming it has enough internal memory).

**CxA Synchronization**

CxA graphs are usually cyclic. Even the basic examples with just two single-scale models (see Fig. 2.6) will display a communication cycle if both models can influence one another. Moreover CxA are multi-scale systems and kernels can thus function at different time scales, maybe in an adaptive way. These properties make a central scheduler approach impractical. However, the fact that the receive primitive is blocking and the send is non-blocking, allows a data-driven synchronization to occur naturally. Indeed, kernels will just wait until information is available before continuing their computation. An example of such synchronization is shown in Fig. 2.7 for the coral model.

The main problem with this method are possible deadlock situations. However, such issues can be easily prevented with the CxA execution model. In the coral example, deadlock is avoided by having a model (the coral) which sends before receiving. This allows the flow model to continue its computations to produce the data that will unlock the coral, etc. In contrast, the situation presented in Fig. 2.8 will produce a deadlock because both models try to receive before sending anything. This problem is easily solved by moving the observation $O_i$ at the beginning to the inner loop, or adding initial send instructions before entering the submodel execution loop.

Furthermore, the fact that communication is pairwise and that the conduits use buffers, makes race conditions impossible. Data are meant to be read by only one process, data sent in a conduit entrance will be processed only by that conduit and data moved to conduit exits will concern only a single kernel.

**CxA Termination**

The termination of the whole CxA is also designed to be fully decentralized: when a kernel finishes its computations (because of e.g. a preset maximum time or a
Figure 2.7: UML sequence diagram of the CxA shown in Fig. 2.6. The vertical lines represent the "life-line" of the process: the kernels are represented by rectangles and the conduits by ovals. When a process is active, the gray life line is replaced by a vertical white rectangle. The arrows represent interaction. Solid arrows with triangular heads are blocking interactions and solid arrows with thin heads represent non-blocking interactions. The return values are indicated by dashed arrows. The circled numbers correspond to Fig. 2.6.
steady state condition), it first notifies all its conduits and then it terminates itself. Similarly, when a conduit receives termination notifications from all connected kernels, it can terminate itself. While the conduit termination rule is always safe (a conduit stops when no kernel is connected anymore), the kernel termination rule needs an extra mechanism. Otherwise, a problem occurs if a kernel is waiting for information from an already terminated kernel.

For instance, in the coral example (Fig. 2.6) the flow model will hang on for the domain update, even after the coral model termination. To solve this issue a stop signal is introduced which is able to release a kernel blocked in the receive primitive. This signal is propagated by a kernel through the existing conduits, using a third primitive: stop(): this primitive sends the stop signal through a conduit. The receive primitive is then modified slightly. It works exactly as seen above but can return either the expected data or the stop signal.

Therefore a kernel waiting for data can be released by a stop signal. Kernels are then responsible to send, process and propagate stop signals. Generally a kernel receiving a stop signal should:

1. Abort the submodel execution loop.
2. Send some final data, if required.
3. Propagate the stop signal to each connected conduit entrances.
4. Notify each connected entrance and exit.
5. Terminate itself gracefully.

With this termination scheme, all kernels which need data from the rest of the CxA will thus stop. The stop signal can originate from any kernel, and this approach also works if two (or more) kernels reach a stop condition at the same time.

As an illustration we can add a stop mechanism to the example of Fig. 2.6, as follows:

1. Coral submodel

   While Not EC
   
   D := U(D)
   DomainConduit.send( D )
   f := B(f)
   velocityMap :=
   VelocityConduit.receive()
   f := C(f,velocityMap)
   f := P(f)
   End
   DomainConduit.stop()
   myStop()

2. Flow submodel

   While True
   
   domain :=
   DomainConduit.receive()
   If domain == STOP_SIGNAL
   myStop()
   D := domain
   f := f_init
   While Not Steady_State
   [SEL]
   End
   End

where myStop() is a user-defined function which terminates the kernel. But, before, if needed, it: (i) saves results, (ii) propagates the stop signal, (iii) notifies the connected conduits.
Parallelization

With the execution model described above, our framework is compatible with a distributed or grid computing approach, in which each submodel could run on a different core or, alternatively, as a different threads on the same core. The actual support for parallelization depends on the chosen implementation of our framework. For instance, the MUSCLE library\(^4\) offers an easy but manual parallelization. On the other hand the CxA-lite library\(^5\) only allows a multi-thread execution in which all the submodels share the same memory space.

MUSCLE Library

Within the framework of the COAST project the conceptual ideas behind the CxA approach (decomposition into single scale models, restriction to a common instruction flow and specification of finite number of coupling templates) have been used to develop the Multi-scale Coupling Library and Environment (MUSCLE) \([40]\), a software environment in which a CxA can be implemented naturally.

Within the coupling library, both the kernels (i.e. the single scale models) and the conduits (i.e. the multi-scale coupling) are software agents of the underlying multi-agent platform JADE\(^6\). Kernels and conduits (conceptually central to the CxA modeling language) communicate using the two communication primitives of non-blocking \textbf{send} and blocking \textbf{receive}. The single scale models do not need to be aware of each other and the information on the coupling and the global setup are held by the framework. This allows the implementation of complex interfaces, where multi-scale couplings can be performed by the use of smart conduits.

Furthermore, the structure of the coupling library allows complete independence from native codes. These can be replaced with a different source, provided the interface with respect to the framework (i.e. the JAVA-wrapper agent) remains the same. In the particular example of in-stent restenosis, described in Sec. 3, three (in case of the 2D model) or four (3D model), respectively, single scale models have been implemented in different programming languages (FORTRAN90, C++, JAVA), wrapped as JAVA agents, and connected via the MUSCLE framework.

2.2.7 Formalism

The concept of a CxA as a set of coupled CA’s, where the coupling is expressed in terms of input-output relations between operators of the SEL of the coupled CA’s is not just a concept that allows to classify multi-scale models, as discussed in section 2.2.4, or a powerful concept to built CxA simulation software, see section 2.2.6, but it is also amenable to mathematical formalism and analysis. This section will introduce some of the formalism, which will be further used in one of the examples of section 2.3.

Recalling (2.2), the state of a CA at a certain time \(t\) is described by a \(f^t \in \mathbb{F}\), denoting the numerical solution at the time step \(t\), which evolves according to

\[
\begin{align*}
  f^0 &= f_{\text{init}}[u_0], \text{ initial condition} \\
  f^{t+\Delta t} &= \Phi[u; f^t]
\end{align*}
\] (2.9)

\(^4\)http://developer.berlios.de/projects/muscle
\(^5\)http://github.com/paradigmatic/CxALite/
\(^6\)www.jade.tilab.com
where $u_0$ is an external field connected to the initial condition. As previously discussed, we constrain the update rule $\Phi$ to the form

$$\Phi[u; f] = (B[u_B] \circ P \circ C[u_C]) [f],$$

(2.10)
i.e. written as a composition of three operators: collision $C[u_C]$, depending on external parameters $u_C$, propagation $P$, depending on the topology of the domain, and boundary condition $B[u_B]$, depending on external parameters denoted by $u_B$.

More precisely, the space of the states $F$ and the update rule $\Phi$ depend in general on the discretization parameters $\Delta x$ and $\Delta t$. For simplicity, in what follows, we let the definition of CA depend also on a (small) parameter $h$, related to spatial and temporal discretizations (for example $\Delta x_h = h$, $\Delta t_h = \alpha h$). Accordingly, considering the CA $C_h$, the evolution space and the update rule can be denoted as: $\Phi_h : F_h \rightarrow F_h$. Shortly, we will call $f_h$ the numerical outcome of the CA $C_h$.

To begin with, as in the left diagram in Fig. 2.1, we consider a multi-scale system represented as a single $C_h$ defined as in (2.2). Building a CxA, instead of describing the system with a single $f_h$, we lower the dimension of the problem and the computational complexity, introducing coarser temporal and/or spatial discretizations

$$H = (h_1, \ldots, h_M)$$

(2.11)
and building a corresponding Complex Automaton

$$\text{CxA}_H = (C_{h_1}, \ldots, C_{h_M}),$$

(2.12)
where each $C_{h_m}$ is an object as in (2.2).

Formally, the definition of a CxA can be summarized in two steps. First, a projection of the space of states $F$ on a product of spaces is considered

$$\Pi_{Hh} : F_h \rightarrow F_{h_1} \times \ldots \times F_{h_M},$$

(2.13)
each describing the evolution of a single scale model (on different discretizations). Second, a rescaling of the update rule is performed, according to the new discretizations $h_i$, on each space $F_{h_i}$, for $i = 1, \ldots, M$, depending on the multi-scale technique used. Due to the form of the execution model of each CA, the rescaling can be easily expressed in terms of operations on the operators $P$, $C$, $B$. Note that the spaces $F_{h_i}$ are not necessarily disjoint, i.e. part of a single scale evolution space could be shared by several CA, in case of space overlap and single domain coupling.

Let us denote with $f_{\text{CxA}}$ the numerical outcome of the complex automata simulation and with $f_{h_m}$ (or $f_m$) the state variable of the single CA’s. To be able to compare the results of the CxA versus the original multi-scale algorithm, we associate an observable $O_{\text{CxA}}$ to the Complex Automata, which projects the result $f_H$ on the space of $O(f_h)$. A sketch of the relevant spaces and operators is drawn below.
For the sake of simplicity, in what follows we describe the formalism restricting ourselves to the evolution of two coupled single scale models. From equations (2.9)-(2.10), we have the following general representation

\[
\begin{align*}
  f_{t_0}^1 &= f_{\text{init,1}}[f_2] \\
  f_{t_0}^{1+\Delta t_{h_1}} &= (B_{h_1}[f_2] \circ P_{h_1} \circ C_{h_1}[f_2])[f_1^1], \\
  f_{t_0}^2 &= f_{\text{init,2}}[f_1] \\
  f_{t_0}^{2+\Delta t_{h_2}} &= (B_{h_2}[f_1] \circ P_{h_2} \circ C_{h_2}[f_1])[f_2^2],
\end{align*}
\]  

(2.14)

where two CA’s are fully coupled in all the components. In detail,

- \( f_{\text{init,1}}[f_2] \) denotes a coupling through initial conditions (i.e. the initial condition of 1 depends on the results of 2)
- \( B_i[f_j] \) expresses coupling through boundary conditions,
- \( C_i[f_j] \) expresses the coupling through collision operator.

In general, for different situations (multi-domain/single-domain, time/space separation/overlap) we can restrict the set of possible couplings to a well-specified coupling template. Consider the example of a microscopic fast process coupled to a macroscopic slow process (micro-macro coupling), as introduced earlier in section 2.2.4. The macroscopic process takes input from explicit simulations of microscopic processes at each time step and on each lattice site of the macroscopic process. The microscopic processes run to completion, assuming that they are much faster than the macroscopic process and therefore are in quasi-equilibrium on the macroscopic time scales (this approach is known in the literature as the Heterogeneous Multi-scale Method, see [7]). The macroscopic process could e.g. be a fluid flow with takes its viscosity from an underlying microscopic process (e.g. explicit suspension model).

In Fig. 2.8 we show for this example of micro-macro coupling the SSM (left) and the coupling template (right). The later is defined in [30] and shows how the operators as defined in (2.10) are coupled to each other. A close inspection of this coupling template shows indeed that, upon each iteration of the macroscopic process, the microscopic process executes a complete simulation, taking input from the macroscopic process. In turn, the output from the microscopic process is fed into the collision operator of the macroscopic process.
We can formulate the CxA dynamics as follows (based on (2.14) )

\[
\begin{align*}
    f_{t_0}^1 &= f_{\text{init},1}[f_2] \\
    f_{t_1 + \Delta t_1}^1 &= (B_1 \circ P_1 \circ C_1)[f_{t_1}^1], \\
    f_{t_0}^2 &= f_{\text{init},2} \\
    f_{t_2 + \Delta t_2}^2 &= (B_2 \circ P_2 \circ C_2)[f_{t_2}^2],
\end{align*}
\]

(2.15)

where 1 refers to the micro-scale and 2 to the macro-scale. The micro-scale model 1 is run until completion (i.e. until the final time \(T_1\)), then a single time step \(\Delta t_{h_2}\) is performed for the macro-scale model.

We can now compare an estimation of the execution time of the CxA model of Fig. 2.8 with that of using a single CA for the same system, as in the left part of Fig. 2.1. For the single CA the execution time would be \(T_{CA} = k_{CA} T_2 \left( \frac{L_2}{\Delta x_2} \right)^D \), which is (2.1) using the subscripts as introduced in Fig. 2.8. For the CxA, the execution time becomes

\[
T_{CxA} = \frac{T_2}{\Delta t_2} \left( \frac{L_2}{\Delta x_2} \right)^D \left( k_2 + k_1 \frac{T_1}{\Delta t_1} \left( \frac{L_1}{\Delta x_1} \right)^D \right). \tag{2.16}
\]

where \(k_{CA}\), \(k_1\) and \(k_2\) are the CPU times to update one spatial cell for one time step, respectively for the full scale CA, the micro and the macro submodels.

Next one can compute a speedup, comparing the single scale CA formulation and the CxA formulation as \(S = T_{CA}/T_{CxA}\). After some algebra we find

\[
S = \left( k_{CA} \frac{\Delta t_2}{\Delta t_1} \left( \frac{\Delta x_2}{\Delta x_1} \right)^D \right) \left/ \left( k_2 + k_1 \frac{T_1}{\Delta t_1} \left( \frac{L_1}{\Delta x_1} \right)^D \right) \right. \tag{2.17}
\]

Under the reasonable assumption that the execution time for a full micro scale simulation needs much more time than a single iteration of the macro scale model, i.e. when \(k_1 \frac{T_1}{\Delta t_1} \left( \frac{L_1}{\Delta x_1} \right)^D \gg k_2\), (2.17) reduces to \(S = k_{CA} \frac{\Delta t_2}{\Delta t_1} \left( \frac{\Delta x_2}{L_2} \right)^D \). Note that \(\frac{\Delta t_2}{\Delta t_1}\) > 1 and \(\frac{\Delta x_2}{L_1}\) > 1, and can be interpreted as the distance on the SSM (Fig. 2.8). So, if the scale separation is large enough, the obtained speedups can be huge, principally rendering a CxA simulation feasible.
2.2.8 Scale-splitting error

The above arguments demonstrate the improvements in computational efficiency offered by the CxA formulation. On the other hand, replacing the original multi-scale model with many coupled single-scale algorithms, we face a partial loss of precision. A possible measure of this lowering in accuracy can be obtained considering the difference in the numerical results of the original $C_h$ and the Complex Automaton CxA$_{H}$, which we call *scale-splitting error*.

This error is measured according to the observables, i.e. the quantity of interest, formally resulting from the observable operators:

$$
E_{C_h ightarrow CxA} = \|O(f_h) - O_{CxA}(f_H)\|
$$

(2.18)

in an opportune norm. The scale-splitting error has a direct interpretation in terms of accuracy. In fact, calling $E_{CxA,EX}$ the absolute error of the CxA model with respect to an exact reference solution, and $E_{C_h,EX}$ the error of the model itself, we have

$$
\|E_{CxA,EX}\| \leq \|E_{C_h,EX}\| + \|E_{C_h \rightarrow CxA}\|.
$$

(2.19)

If we heuristically assume that the original fine-scale algorithm has a high accuracy, the scale splitting error is a measure of the error of the CxA model.

In general, a detailed and rigorous investigation of the scale-splitting error requires a good base knowledge of the single scale CA and of the full multi-scale algorithm. Case by case, error estimates can be derived using the properties of the algorithms, the operators involved in the update rule and in the coarse-graining procedure. An example of error investigation using the formalism for a simple CxA model can be found in section 2.3.

2.3 Example: Reaction Diffusion

Let us consider a reaction-diffusion process for a concentration field $\rho = \rho(t,x)$ described by the equation

$$
\begin{align*}
\partial_t \rho &= d \partial_{xx} \rho + \kappa (\rho_\lambda - \rho), \\
\rho(0,x) &= \rho_0(x)
\end{align*}
$$

(2.20)

with periodic boundary conditions in $x$, $\rho_0$ being the initial condition and $\rho_\lambda(x)$ a given function. To consider a multi-scale model, we assume the reaction to be characterized by a typical time scale faster than the diffusion.

Numerically, problem (2.20) can be solved employing a lattice Boltzmann method (LBM) (see for example [1, 41, 42] and the references therein), discretizing the space interval with a regular grid $G_h = \{0, \ldots, N_x - 1\}$ of step size $\Delta x_h = h$ and associating each node $j \in G_h$ with two variables, $f_1$ and $f_{-1}$ representing the density of probabilities of populations traveling with discrete velocities $c_i \in \{-1, 1\}$. The collision+propagation update has the form

$$
f^{t+n+\Delta t}_i(j + c_i) = f^{t+n}_i(j) + \frac{1}{\tau} \left( \hat{\rho}^{t+n}_i(j) - f^{t+n}_i(j) \right) + \Delta t_h \frac{1}{2} R(\hat{\rho}^{t+n}_i(j)).
$$

(2.21)
here $R(\hat{\rho}(j)) = \kappa(\rho_{\lambda}(j) - \hat{\rho}(j))$, and $\hat{\rho} = \rho(f) = f_1 + f_{-1}$ is the numerical solution for the concentration field. The time step is related to the grid size according to

$$\frac{\Delta t_h}{\Delta x_h^2} = \text{const. \ \forall h,} \quad (2.22)$$

and the parameter $\tau$ is chosen according to the diffusion constant in (2.20) (see [42, 1])

$$\tau = \frac{1}{2} + d \frac{\Delta t_h}{\Delta x_h^2}. \quad (2.23)$$

Equivalently, we can rewrite (2.21) in the form [43]

$$f^{t_{n+1}}_h = P_h(I_h + \Omega_{D_h}(\tau))(I_h + \Omega_{R_h})f^{t_n}_h = \Phi f^{t_n}_h, \quad (2.24)$$

highlighting the scale $h$ and omitting the subscript $i$. The update $\Phi = P_h(I_h + \Omega_{D_h}(\tau))(I_h + \Omega_{R_h})$, has been decomposed into a diffusion part and a reaction part.

The space of states is the set $F_h = \{ \phi: \mathcal{G}_h \rightarrow \mathbb{R}^{2N_x} \}$, of the real functions defined on the grid $\mathcal{G}_h$. The subscript $h$ for the operators denotes functions acting from $F_h$ to itself. In detail, $I_h$ is simply the identity on $F_h$, $P_h$ acts on a grid function shifting the value on the grid according to $c_i \rightarrow f_{i,h}^{j} = f_{i,h}^{j + c_i}$, $\Omega_D$ and $\Omega_R$ are the operations defined in the right hand side of (2.21):

$$\Omega_D f = \frac{1}{\tau} (f_{eq}(\rho(f_h)) - f_{i,h}), \quad (\Omega_R f_{i,j}) = h^2 \frac{1}{2} R(\rho(f_h))$$

The SSM for this example is shown in Fig. 2.9. To define the CxA, we set $\Delta t_R = \Delta t_h = h^2$ for the reaction and $\Delta t_D = M h^2$ for the diffusion. Focusing on the case shown in Fig. 2.9b, the reaction is run up to a time $T_R$, then re-initialized after a diffusion time step. If $T_R = \Delta t_D$, the two processes are not completely separated. Fig. 2.9c sketches the case when reaction leads very quickly to an equilibrium state in a typical time which is even smaller than the discrete time step of the diffusion.

We focus on the case of time-coarsening, i.e. choosing

$$\Delta x_D = \Delta x_R = h, \ \Delta t_D = M \Delta t_R = M h^2. \quad (2.25)$$

Introducing reaction and diffusion operators $R_s, D_s$, where $s = R, D$ specifies the dependence of the discrete operators on the space-time discretization of reaction and (resp.) diffusion, the evolution of the system can be described with the state variable $f_H = (f_R, f_D)$, whose components are updated according to

$$f_{R}^{t_{n+1}} = R(f_{R}^{t_n}), \quad f_{D}^{t_{n+1}} = D(f_{D}^{t_n}). \quad (2.26)$$

Equation (2.26) expresses that the algorithm $CA_R$, which is coupled to $CA_D$ through the initial condition (by setting at the initial time $t_0 = t_D$ (equal to a certain time of $CA_D$) the initial condition equal to the one obtained from $CA_D$,
Figure 2.9: SSM for the reaction-diffusion LBM. In (a) reaction (dashed line) and diffusion (solid line) are considered as a single multi-scale algorithm. In (b) we assume to use different schemes, where the diffusion time step $\Delta t_D$ is larger than the original $\Delta t_h$. (c) represents the situation where the two processes are time separated, with a very fast reaction yielding an equilibrium state in a time $T_R \ll \Delta t_D$.

and evolves for $M$ steps according to an update rule depending only on the reaction process. On the right, the diffusion part $C\Delta D$ is coupled to the reaction through the collision operator, since the new state of $f_D$ is locally computed starting from the output state of $C\Delta R$. With $f_{D,\text{init}}(\rho_0)$ we denoted the original initial condition, function of the initial concentration in (2.20).

In this case, the observable is represented by the concentration $\rho$, obtained from the numerical solution by a simple average over the particle distributions.

According to the definition introduced in 2.2.8, we define the scale-splitting error at time iteration $t_N$ as

$$E(t_N) = \|\rho(f_{h,N}^t) - \rho(f_{D,h,N}^t)\|$$

i.e. the difference between $\rho(f_h)$, the solution of the fine-grid algorithm (2.24) and the output of the CxA model (2.26) evaluated at the same physical time. Taking $f_{D,h}$ in the above formula corresponds to evaluating the error after both reaction and diffusion have been executed. Note that both $f_h$ and $f_{D,h}$ belong to the same evolution space, since the grid spacing are equal.

Following section 2.2.8 we now consider the scale-splitting error $E(M)$ resulting from using a diffusion time step $\Delta t_D$ $M$ times larger than the reaction time step $\Delta t_R$. The reference solution is here the solution obtained when both reaction and diffusion act at the smallest time scale, i.e. when $M = 1$. To estimate $E(M)$ we consider $M$ reaction steps at scale $h$ (defined by $\Delta t_R$) followed by one diffusion step at the coarser scale $h'$ (defined by $\Delta t_D = M\Delta t_R$) and we compare the results with $M$ reaction-diffusion steps both at the fine scale $h$. In terms of the reaction and diffusion operators, $E(M)$ can be expressed as

$$E(M) = \|(D_hR_h)^M - D_{h'}R_{h'}^M\|$$

$$\leq \|\neq (D_hR_h)^M - D_{h'}R_{h'}^M\| + \|[D_{h'}^M - D_h^M]R_h^M\|$$

$$= E_1(M) + E_2(M)$$

(2.28)

Contribution $E_1$ can be computed from the commutator $[D_hR_h - R_hD_h]$ and $E_2$ follows from the time coarse-graining of the original LB model. After some calcu-
lations we obtain (see [44])

\[ E(M) \leq \mathcal{O}(M^2 \kappa) + \mathcal{O}(M^2 D^3) \]  

(2.29)

\[ E(t_N) = \| \rho \left( (\mathcal{D}_h \mathcal{R}_h)^M f^{t_N - M \Delta t_h} - \mathcal{D}_{D,h} \mathcal{R}_{D,h}^M f^{t_N - M \Delta t_D} \right) \| \leq \| \rho \left( (\mathcal{D}_h \mathcal{R}_h)^M - \mathcal{D}_{D,h} \mathcal{R}_{D,h}^M \right) f^{t_N - M \Delta t_D} \| + C(h, \mathcal{D}, \mathcal{R}) E(t_N - \Delta t_D), \]  

(2.30)

\[ E(t_N) \leq \| \rho \left( (\mathcal{D}_h \mathcal{R}_h)^M - \mathcal{D}_{D,h} \mathcal{R}_{D,h}^M \right) \| \leq \| \rho \left( (\mathcal{D}_h \mathcal{R}_h)^M - \mathcal{D}_h^M \mathcal{R}_h^M \right) \| + \| \rho \left( \mathcal{D}_h^M \mathcal{R}_h^M - \mathcal{D}_{D,h} \mathcal{R}_{D,h}^M \right) \|. \]  

(2.31)

The first contribution depends on the difference \((\mathcal{D}_h \mathcal{R}_h)^M - \mathcal{D}_h^M \mathcal{R}_h^M\), which can be estimated as a function of \([\mathcal{D}_h, \mathcal{R}_h] = \mathcal{D}_h \mathcal{R}_h - \mathcal{R}_h \mathcal{D}_h\), i.e. the commutator of the operators \(\mathcal{R}_h\) and \(\mathcal{D}_h\). It can be shown that [44]

\[ [\mathcal{D}_h, \mathcal{R}_h](f_h) = O \left( h^3 \kappa \partial_x (\rho(f_h) - \rho_\lambda) \right). \]  

(2.32)

For the second contribution, deriving from the coarsening of the diffusion part of the original lattice Boltzmann algorithm, we obtain

\[ \rho \left( \mathcal{D}_h^M - \mathcal{D}_{D,h} \right) f_{D,h} = O(M^2 D^3 h^2). \]  

(2.33)

Furthermore, we observe that \(\mathcal{R}_h^M = \mathcal{R}_{D,h}^M\).

In conclusion, a qualitative estimate of the scale splitting error in terms of \(M\) is given by

\[ E_{C_h \rightarrow CxA} = O \left( M^2 \kappa \right) + O \left( M^2 D^3 \right). \]  

(2.34)

**Numerical Validation**

We consider the problem

\[ \partial_t \rho = d \partial_x x \rho - \kappa (\rho - \sin (\lambda x)), \quad t \in (0, T_{end}], \quad x \in (0, 1], \]  

\[ \rho(0, x) = \rho_0(x) \]  

(2.35)

with \(\frac{\lambda}{2\pi} \in \mathbb{Z}\), and periodic boundary conditions in \(x\)-direction.

By selecting different values of the parameters regulating (2.35) we can tune the relevance of different time scales. Additionally, we introduce the non dimensional parameter

\[ \sigma = \frac{\kappa}{\lambda^2 d} \]

to ”measure” the scale separation of the simulation. In the numerical tests, we run both the original fine scale LBM and the CxA model, measuring explicitly the scale-splitting error as the difference in the resulting concentrations. Fig. 2.10
Figure 2.10: Scale-splitting error as a function of $M$ for a time-coarsened CxA. The different curves represent different values of $\sigma$. Simulation parameters: $h = 0.02$, $\lambda = 4\pi$, $\kappa = 10$, $d \in \{0.05, 0.1, 0.25, 0.5\}$. (a): $1 < M < 100$. (b): $M > 100$. The size of the scale-splitting error becomes relatively large, except for the case $\sigma = 0.2$. (c): Order plot (fig. (a)-(b) in double logarithmic scale) of maximum scale-splitting error versus $M$. The dashed lines of slope 1 (bottom) and 2 (top) indicate that $E \sim M^\alpha$, with $1 < \alpha < 2$. For the largest $M$ a leveling off towards a maximum error can be observed.

shows the results of scale-splitting error for different values of $M$. The order plot in fig. 2.10c confirms estimate (2.34).

Results of a further test to link together scale separation and scale-splitting error are shown in Fig. 2.11a-b. Namely, for each simulation drawn in Fig. 2.10, we select the first $M$ such that the scale splitting error lies below a certain prefixed threshold error $\bar{E}(h,H)$. These values $M_{th}$ are plotted then as function of $\sigma$, validating the idea that better scale separation allows more efficient CxA formulations.

Detailed analysis and investigation of this example can be found in [44].

Figure 2.11: (a): Zoom of the previous fig. 2.10a, including a threshold error $\bar{E}(h,H) = 0.05$ (results with $\sigma = 0.04$ are also shown). (b): Values of $M_{th}$ such that the scale-splitting error equates a threshold error $\bar{E}(h,H)$, versus the measure of scale separation $\sigma = \kappa (\lambda^2 D)^{-1}$ (in double logarithmic scale).

We can also compute the speedup resulting from the above time-splitting. Let us call $a$ and $b$ the CPU times of one iteration of respectively the reaction and
the diffusion processes. If we run the full system at the finer scale $\Delta t_R$ for a time $T$, the total CPU time will be proportional to $(a + b)(T/\delta t_R)$. With the time-splitting method, the CPU time reduces to $(Ma + b)T/(M\Delta t_R)$ and the speedup is $(a + b)/(a + b/M)$. For large $M$, the speedup tends to $1 + b/a$. This might not be a very big gain, unless $a \ll b$. However, if we would have coarse grained the spatial scale for the diffusion processes, we would get a more interesting speedup value.

2.4 Conclusion

This chapter described a possible approach towards multi-scale modeling and simulation using Cellular Automata. The concept of Complex Automata should allow the modeling of a large range of multi-scale systems, and the related Complex Automata simulation software provides a framework to quickly develop Complex Automata simulations. The ideas behind Complex Automata have a broader significance than Cellular Automata modeling alone and, in the near future, we will explore the possibility to enlarge the CxA idea to other modeling paradigms. Moreover, we are developing a growing set of CxA models and simulations. This thesis will focus on two of them in the following chapters. Sec. 3 will present a model of a challenging multi-science problem in the field of cardiovascular diseases, namely that of in-stent restenosis. The subsequent chapter 4 will describe a multi-scale model for the simulation of macroscopic suspension flow based on microdynamics. Both of them will be described in the spirit of the concept of Complex Automata which proves very helpful in classifying and constructing single-scale models and their coupling.
Chapter 3

In-stent Restenosis

3.1 In-stent Restenosis

A stenosis is a narrowing of a blood vessel lumen due to the presence of an atherosclerotic plaque. This can be corrected by balloon angioplasty, after which a stent (metal mesh) is deployed to prevent the vessel from collapsing. The injury caused by the stent can lead to a maladaptive biological response of the cellular tissue (mainly due to smooth muscle cell proliferation). The abnormal growth can produce a new stenosis (re-stenosis).

Restenosis develops under conditions of pulsatile flow and there exists an interaction between the much studied biological pathways and those of a physical nature [45, 46]. The multi-science and multi-scale nature of in-stent restenosis has been discussed in detail previously by Evans et al. [32].

The design and geometry of the stent employed influences the biological events occurring in the vessel following deployment. Strut thickness, number, cross-sectional shape and arrangement, and stent length all influence the haemodynamics and degree of injury and stretch observed within the stented segment [47]. These in turn, are critical determinants of the severity of restenosis observed. Additionally, stents may be coated with active compounds targeted at the biological processes responsible for driving the progression of restenosis which, when eluted locally at the stented site, can prevent proliferation of smooth muscle cells and neointimal growth.

The development of a multi-scale in silico model capable of testing both the influence of stent geometry and that of drug elution is motivated by the desire for a better understanding of the dynamics regulating restenosis. Thus providing a potentially powerful tool for improved understanding of the biology, and to assist in the process of device/therapy development.

As in many other biological systems, the dynamics of in-stent restenosis span many orders of magnitude through the scales, from the smallest microscopic scales up to the largest macroscopic ones. The wealth of experimental data that is now

---

1 This chapter is based on A. Caiazzo, D. Evans, J.-L. Falcone, J. Hegewald, E. Lorenz, B. Stahl, D. Wang, J. Bernsdorf, B. Chopard, J. Gunn, R. Hose, M. Krafczyk, P. Lawford, R. Smallwood, D. Walker, and A. Hoekstra, A Complex Automata approach for In-stent Restenosis: two-dimensional multiscale modeling and simulations (submitted) and extended by the section on the thrombus formation model.
available has made in silico experimentation an attractive tool in systems biology, allowing hypothesis testing and formulation of predictions which can be further tested in vitro or in vivo [48]. In recent years the computational biology community has developed extremely powerful methods to model and simulate fundamental processes of a natural system on a multitude of separate scales. The next challenge is to study, not only fundamental processes, on all these separate scales, but also their mutual coupling across the scales and to determine the emergent structure and function of the resulting system [49]. Despite the large body of literature on multi-scale models, we have found that there is a surprising lack of (formal) methodology for multi-scale modeling (see Sec. 2 and [31]). Moreover, the key feature of multi-scale modeling, the actual coupling between scales is still at a very early stage of development [50]. In this context, in Sec. 2 introduced Complex Automata (CxA) have been introduced as a paradigm to simulate multi-scale systems as a collection of single scale models, interacting across the scales (see also [31, 51, 30]).

Based on the conceptual description of the relevant processes and their characteristic temporal and spatial scales which has been presented in [32], we describe a simplified CxA model of the multi-scale process, coupling a lattice Boltzmann bulk flow (BF) solver (for the blood flow), an agent based model for smooth muscle cell (SMC) dynamics (simulating cell growth, the cell cycle, physical and biological cell-cell interaction), and a Finite Difference scheme for the drug diffusion (DD) within the cellular tissue.

In section 3.2, following a short introduction on in-stent restenosis, we present the two dimensional multi-scale model for ISR. We describe the main characteristics of the single scale solvers, which have been developed independently from each other, and independently from the ultimate application. We also describe, in detail, the coupling of the single-scale solvers with relevance to this particular application. Preliminary simulation results are presented in section 3.3. In Sec. 3.4 an outlook towards a 3D CxA for ISR is given. As an additional single scale model the thrombus formation model is introduced in Sec. 3.4.1. Within this subsection an overview of the biological background is given and modeling approaches are discussed. Subsequently, a very simple thrombosis model is proposed and validated, and its implementation and coupling into the 3D ISR CxA is discussed. This chapter closes with a short conclusion in Sec. 3.5.

### 3.2 Multi-scale Model of In-stent Restenosis

Restenosis, can be loosely described as a 'loss of gain' - that is, a late return of the vessel lumen to a size similar to that seen before intervention (stent deployment; see figure 3.1). It has, historically, been considered as an overreaction of the general wound healing response within vascular tissue [52]. From a biological standpoint, injury caused by stent deployment (during balloon inflation) is thought to trigger a cascade of inflammatory events, that ultimately results in the development of new tissue (the neointima) [53, 54].

The majority of investigations into this phenomenon consider the biological and physical processes involved independently when, in fact, there is a complex interplay between the two. Blood flow, biological events (e.g. inflammation), stent geometry, drug elution and diffusion all influence the overall response of the artery wall to stent deployment. The aim of the CxA model is to improve our
understanding of this complex system by considering restenosis explicitly as a multi-scale multi-science system.

Following an in depth literature review, the processes key to the regulation of restenosis were identified, and their temporal and spatial scales determined. Coupling was considered in terms of the interactions between these processes. This allowed us to generate a comprehensive conceptual scale separation map [32], defining a CxA, containing the sub-models necessary to capture the behavior of the system, and depicting the coupling between them; i.e. the flow of information between models.

The first practical implementation of the CxA reported herein considers a simplified version of the model focusing on SMC behavior, and its interaction with blood flow and drug eluted from the stent. The simplified SSM is shown in figure 3.2.

Following deployment of the stent, which is modeled as a separate process to provide an initial condition (using the SMC model itself, see section 3.3.1), SMCs start to proliferate in response to the mechanical insult. The rate of smooth muscle cell proliferation is dependent on the blood flow (specifically wall shear stress (WSS) and oscillatory stress index (OSI)), the number of neighboring smooth muscle cells, and in the case of a drug eluting stent, the local concentration of drug. The blood flow, in turn, depends on the lumenal geometry (and thus changes with the proliferation of SMCs), and the concentration of drug depends on the SMC/tissue domain (and therefore also on SMC proliferation). In the current model we assume that scale separation between the single scale models is confined to the temporal scale, however it is worth noting that scale separation on a spatial scale exists within the SMC model itself. The SMC model can sub-divided into the processes which occur on the cellular level, and those occurring on the level of the tissue, resulting in a hierarchical CxA model. The SMC proliferation is the slowest process, dictated by the cell cycle, whereas flow is a fast process, dictated by the length of one cardiac cycle. Due to the specific value of the diffusion coefficients and the
3.2.1 Single Scale Models and Coupling Templates

In this section, the technical details of the CxA model of in-stent restenosis are presented in brief. We first describe the kernels of the CxA, i.e. the algorithms used to simulate the single scale models (Bulk Flow, SMC Behavior and Drug Diffusion). The native codes of these have been constructed independently from the multi-scale application. Then, we show how these elements are connected via smart conduits using a CxA dedicated coupling library [55].

**Bulk Flow Solver (BF)**

Blood flow is modeled as a Newtonian incompressible fluid governed by incompressible Navier-Stokes equations

\[
\begin{align*}
\rho_0 \partial_t \mathbf{u} + \rho \mathbf{u} \cdot \nabla \mathbf{u} + \nabla p &= \rho \nu \nabla^2 \mathbf{u}, \quad t > 0, \ x \in \Omega_{\text{flow}}(t) \\
\nabla \cdot \mathbf{u} &= 0, \quad t > 0, \ x \in \Omega_{\text{flow}}(t) \\
\mathbf{u}(t, \mathbf{x}) &= 0, \quad \mathbf{x} \in \Gamma_{\text{flow}}(t)
\end{align*}
\]

where \(\rho_0\) is the blood density, \(\nu\) is the viscosity, assumed constant in the Newtonian approximation (a commonly accepted hypothesis for large vessels). The set
Figure 3.3: Left: The 2D computational domain is divided into vessel lumen, tissue and stent struts. Right: the cell cycle model, based on three stages (G0, G1, S/G2/M) and a biological ruleset.

\( \Omega_{\text{flow}} \) represents the lumen domain, with \( \Gamma_{\text{flow}}(t) \) being its interface with the tissue domain\(^2\).

To obtain a numerical solution of (3.1), we employ a Lattice Boltzmann Method (LBM), which, unlike other CFD approaches, approximates the hydrodynamics starting from a pseudo-microscopic description of the fluid. In detail, the spatial domain is discretized using a regular lattice \( \mathcal{L}(h) \), of spacing \( h \), and a set of discrete velocity vectors \( \{hc_i, i = 1, \ldots, b\} \), connecting neighboring nodes of the lattice. At each node \( \mathbf{x} \in \mathcal{L}(h) \), and at each time step \( t \), the unknowns are the distributions \( f_i(t, \mathbf{x}) \), representing the density of particles traveling in direction of \( c_i \).

Given the time step \( \Delta t \), and with \( \Delta x = h \), the evolution in time of the variables \( f_i \) reads

\[
    f_i(t + \Delta t, \mathbf{x} + \Delta x c_i) = f_i(t, \mathbf{x}) + J_i(f(t, \mathbf{x}))
\]

where the right hand side defines the collision operator, and depends on the viscosity \( \nu \) in (3.1).

For detailed overviews of the LBM, we refer the reader to [56, 57, 1, 41].

The observable related to the BF single scale model is the wall shear stress on the vessel boundary (WSS), which is needed as input for the SMC model, after being properly mapped from the Cartesian lattice on the individual cells.

**Smooth Muscle Cells Dynamics (SMC)**

The dynamics of the smooth muscle cells are simulated using an Agent Based Model. Each single cell is represented by agent, which is identified by a set of state-variables: position, radius, biological state, drug concentration and structural stress.

Each SMC agent evolves in time according its own current state, and to the states of neighboring cells. Each time step involves a physical solver, simulating the structural dynamics of cells, and a biological solver, which simulates the cell cycle, according to a biological rule set.

\(^2\)Equation (3.1) has to be completed with appropriate inlet-outlet boundary conditions. As this is not directly related to the multi-scale model, details are omitted for simplicity.
Physical solver  From the structural point of view, 2D cells are represented by their centers, and a potential function, which determines non-linear repulsive and attractive inter-cell forces. In addition, boundary forces, viscous friction, radial elastic forces (modeling the primary fiber direction of SMCs in a physiologically relevant 3D environment) and motility forces (modeling cell migration) are taken into account.

Neglecting inertial terms, the model is described by the system of equations

\[ C \frac{dx}{dt} = F(t, x, r) = F_{\text{rep}}(t, x, r) + F_{\text{att}}(t, x, r) + F_{\text{el}}(t, x) + F_{\text{bound}}(t, x), \tag{3.3} \]

where \( x \) is the vector of cell displacements, \( r \) is the vector of cell radii, \( C \) is a matrix of friction coefficients and \( F \) is the vector of forces on cells (including cell-cell, boundary and external forces).

At each iteration step, new equilibrium positions of SMCs are computed by iterating a finite difference scheme\(^3\) for equation (3.3) until steady state is reached. A surrogate of structural stress is then calculated and provided as input to the biological solver.

Biological solver  The cell cycle model consists of a discrete set of states, a quiescent state \( G_0 \), a growth state \( G_1 \) and a mitotic state \( S/G2/M \) (see figure 3.3).

Progression through the cell cycle takes place at a fixed rate, depending on the time step, culminating in mitosis (cell division; a mother cell divides into two new daughter cells). Cells may enter or leave an inactive phase of the cell cycle (\( G_0 \)) depending on certain rules based on contact inhibition (calculated by neighbor count), structural stress, and local drug concentration (input from DD). Additionally, for SMCs in contact with the fluid, rules are based on thresholds of wall shear stress (WSS) and oscillatory shear index (OSI) received from BF also apply. Low WSS, high OSI or high structural strain are individually capable of inducing agent proliferation if drug concentration and contact inhibition criteria allow.

Drug Diffusion in Cellular Tissue

Drug eluting stents represent an effective way of inhibiting neointima formation after stent-deployment. This process is captured in the present model through implementation of the Drug Diffusion (DD) kernel. Drug is eluted from the stent and diffuses into the cellular tissue. Thus the spatial domain for the DD kernel is coincident with that of the SMC. Stent struts act as a source whilst boundaries between flow and cells are considered sinks (this assumes that drug eluted into the lumen is continuously flushed away by the faster blood flow). Biological tissues are heterogeneous in nature so we assume that this process can be described using a generic anisotropic diffusion law:

\[
\partial_t c(t, x) = \nabla \cdot (D_{\text{drug}} \nabla c(t, x))
\]

\[
c(t, x) = c_0, \; t > 0, \; x \in \Omega_{\text{stent}} \tag{3.4}
\]

\[
c(t, x) = 0, \; t > 0, \; x \in \Gamma_{\text{flow}}(t)
\]

\(^3\)A simple Euler method is currently used, but higher order Runge-Kutta schemes could also be employed. We found that results do not depend on the applied scheme if the numerical parameters are chosen carefully, and the Euler scheme is significantly more compatible, as it can be naturally written in the collision+propagation form.
where \( c(t, x) \) is the concentration of the drug in the position \( x \) at time \( t \), and \( D_{\text{drug}} \) is the diffusion tensor.

In equation (3.4), \( \Omega_{\text{stent}} \) denotes the part of a vessel occupied by the stent strut, \( \Omega_{\text{tissue}} \) represents the tissue subdomain, and \( \Gamma_{\text{flow}} \) corresponds to its interface with the flow domain. In the practical algorithm, after discretization of the whole model geometry, mesh points are classified as \textit{tissue}, \textit{source} or \textit{sink}. These are treated differently during the computation.

The diffusion tensor is chosen such that diffusion along the axis of the artery (or tangential to a cross section) is at least 10 times higher than diffusion in the radial direction \[58, 59\].

To solve equation (3.4) numerically, we employ a Finite Difference (FD) approach which is solved using a Propagation-Collision loop\(^4\), thus fitting with the CxA modeling language.

According to \[59\], the time scale to reach the steady state is of the order of minutes (comparable with the SSM in figure 3.2). Therefore, when coupling DD and SMC, we are mainly interested in the steady drug concentration (the time step for the SMC model, which uses the drug concentration as input, is of the order of 1 day). In the context of this CxA model, this allows direct consideration of the simplified equation:

\[
\nabla \cdot (D_{\text{drug}} \nabla c(t, x)) = 0 \tag{3.5}
\]

(with appropriate boundary conditions).

### 3.2.2 The In-stent Restenosis CxA: Kernels, Connection Scheme and Conduits

In order to combine the single scale \textit{kernels} described above using MUSCLE \[55\], we need to define a communication graph, the \textit{Connection Scheme} (CS), which specifies in detail the communication topology of the CxA, defining which pairs of kernels communicate. The Connection Scheme for the CxA model of in-stent restenosis is shown in figure 3.4.

In addition to BF, DD and SMC kernels, the current CxA setup includes a kernel which generates the initial conditions (IC) by simulating stent deployment into the cellular tissue (see section 3.3.1).

Multi-scale coupling is implemented using special agents called smart conduits. Often, these perform \textit{filtering} operations, converting output data from one single scale model to appropriate input for another. This is the case for geometrical couplings (through changes in the domains), when new SMC configurations (continuum based) are transformed into lattice based computational domains for BF and DD:

**Conduit: SMC to BF.** This conduit converts the array of positions and radii of cell agents, into a computational mesh for the flow solver which is decomposed into fluid and solid nodes.

**Conduit: SMC to DD.** Similarly, this conduit converts the array of positions and radii of the cells, into a computational mesh for the drug diffusion solver.

\(^4\)LB approaches for the diffusion equation could also be used. The choice of a FD scheme was dictated by two main arguments: (i) FD schemes in general need less memory than the LBM; (ii) the choice of a FD helped us to investigate and demonstrate the coupling of the different modeling approaches within the same CxA.
marking the nodes as tissue, source, or sink.

In some instances, the interaction between kernels is slightly more complex, and multiple inputs are required to compute one output. In these cases we introduce mapper agents (see figure 3.4) which, in the present CxA, are required whenever an input to the SMC model is generated:

**Mapper: BF to SMC.** The values of fluid shear stress at the boundary affect the biological evolution of the cells. Given the output of the bulk flow solver, and the current cell configuration, a mapper agent computes the shear stress on each cell. Depending on the discretization used for the flow solver, different approximation approaches can be used. If the flow grid is coarser than the spatial scale of the SMC model (the radius of the cells), an algorithm must be used in order to determine which cells are in contact with the flow, then the shear stress is extrapolated from the closest boundary fluid nodes for each cell position. On the other hand, if the flow discretization is sufficiently fine more fluid boundary nodes interact with a single cell and the shear stress on the cell surface can be calculated by averaging the values of the closest nodes.

**Mapper: DD to SMC** In this case, the drug concentration calculated in the DD has to be mapped to the SMC agents. Given the current drug concentrations and the SMC configuration, the mapper agent approximates the concentration on each cell. As for the shear stress approximation, the algorithm used depends on the grid size of the DD model. If the grid is fine enough (with many lattice nodes per SMC), the concentration on a cell can be integrated. If a coarse DD grid is used, the concentration for each cell is extrapolated using data from the closest nodes.

### 3.3 Simulation Results

#### 3.3.1 Benchmark Geometry and Initial Conditions

As a benchmark geometry for the 2D CxA model, we consider a vessel, of length 1.5 mm and width 1.24 mm, where two square struts of side length 90 \( \mu \text{m} \) have been deployed. The vessel wall has a thickness of 120 \( \mu \text{m} \). Smooth muscle cells are generated with an average radius of 15 \( \mu \text{m} \) and densely packed inside the
wall. To obtain the initial condition based on the above geometry, an initial stress configuration compatible with the initial geometry must be provided. To do this, stent deployment is simulated, iterating the structural SMC solver until a stationary state is reached. The initial cell configuration resulting from this procedure is shown on the left in figure 3.5. The struts are clearly visible, embedded in the upper and lower vessel wall. SMC agents (blue) are lined by smaller internal elastic lamina (IEL) agents which are absent from the vessel wall region where the strut has penetrated.

3.3.2 Qualitative assessment of simulation results

We have run the simulation for an equivalent of 72 days (1700 time steps with $\Delta t = 1h$ for the SMC model) for both a bare metal stent ($n=6$) and a drug eluting stent ($n=6$). In the current 2D implementation of the model, stent deployment results in laceration of the internal elastic lamina (as is observed in vivo) allowing proliferation of smooth muscle cells into the vessel lumen. These preliminary results demonstrate neointimal growth (proliferation of smooth muscle cells) in response to stent-induced injury. If we compare the output from immediately after stent deployment with that of 28 days later (Figure 3.5) it is apparent that the developing neointima causes a reduction in lumen diameter and an increase in wall shear stress. Because the SMC ruleset dictates that SMC agent proliferation is inhibited by high shear, once the neointimal growth causes shear stress to increase past a threshold, an equilibrium is reached and no more proliferation occurs. This fits nicely with biological theory which asserts that a vessel remodels in response to changes in haemodynamic forces, until those forces are normalized [46].

The proliferative response is reduced in the presence of drug; at the simulation endpoint (72 days), average neointimal thickness at the strut site in the absence of drug was $0.206 \pm 0.005$ mm versus $0.192 \pm 0.001$ mm in the presence of drug (Figure 3.6).
This trend was confirmed by examining the 'Normalized Peak Absolute Growth Fraction (NPAGF)'. This is defined as:

\[
\text{NPAGF}(t) = \frac{r_{\text{M-phase}}(t)}{N_{\text{cells}}(t)} \max_{s \geq 0} r_{\text{M-phase}}(s),
\]

i.e. the product of growth fraction \(r_{\text{M-phase}}\) (Percentage of cells in M Phase/100) and total cell number \(N_{\text{cells}}\), divided by the maximum value of the growth fraction across the series (Figure 3.6). The present data suggests that peak proliferation occurs at approximately 22 days in the presence of drug, and 20 days in the absence of drug. As our cell cycle dynamics and bulk flow parameters are based on porcine data, this second value agrees well with the findings of Schwartz et al. [60] who derived the NPAGF for the rat, pig and human based on their experimental data and found the peak for the porcine series to be approximately 20 days.

### 3.3.3 Sensitivity Analysis

The single kernels have been singularly validated and their sensitivity with respect to model dependent parameters has been investigated. We remark that, given the structure of the MUSCLE framework, it is inherently simple to perform further sensitivity studies, for the global CxA setup with respect to key parameters of the single kernels. As an example, we investigated the threshold of drug concentration at which SMC agents change from a proliferative to quiescent phenotype, tunable by changing a single parameter in the global CxA setup. Figure 3.7 shows the relationship between this threshold and the amount of neointima present at seventy two days. Sensitivity analyses using different spatial resolutions for individual kernels can be also easily implemented.
3.4 3D CxA

The two dimensional CxA provides us with a tool for testing hypotheses regarding the relationship between stent geometry, the cellular response to injury and the influence of haemodynamic forces. In order to evaluate realistic stent designs, however, it is necessary to run three dimensional simulations. MUSCLE was used to couple three dimensional versions of the bulk flow, SMC and drug diffusion kernels, and additionally, a thrombus kernel.

3.4.1 Single-scale model for thrombus formation (TF)

Platelet Thrombus Formation

The process of thrombus formation is one of the early responses of the biological system to the deployment of the stent. Endothelial cell loss due to stent deployment results in exposure of thrombogenic sub-endothelial molecules which activate platelets, causing them to aggregate. As a consequence of the change in flow geometry, activation of platelets also occurs due to non-laminar flow and increased shear stresses. Platelet activation can also be induced when blood is exposed to a reactive surface, e.g. the synthetic surface of a cardiovascular device [61]. Activated platelets can interact with the vessel wall to form mural thrombi. Activated platelets serve to achieve primary haemostasis and release a plethora of molecules that are able to influence the subsequent healing response. Once adhered to the vessel wall they secret chemicals that support the invasion of fibroblasts from surrounding connective tissue into the wounded area. Leukocytes are able to interact transiently (i.e. roll) with platelets before becoming stationary and adhering firmly. Both leukocyte rolling and firm adhesion are mediated by the interaction of adhesion molecules and their respective receptors on the leukocyte and platelet surfaces. Transmigration of the leukocyte into the vessel wall occurs in response to chemotactic agents. The platelet clot is slowly dissolved by the fibrinolytic enzyme.
Figure 3.8: The time course of the processes involved in human in-stent restenosis. As an immediate response to the initial injury caused by the deployment of the stent thrombus formation is triggered and lasts in decaying strength for up to 30 days. It slightly overlaps the onset of SMC hyperplasia. In the current work thrombus formation is modeled as an initial condition to SMC hyperplasia. Leukocyte accumulation is not modeled explicitly.

(PLASMIN) and the platelets are cleared by phagocytosis.

The presence of adequate red blood cell numbers supports the formation of a haemostatic plug in vivo. This observation can be attributed to a red blood cells' physico-chemical effect on platelet transport and platelet-surface interactions. For example, red blood cell release of adenosine-diphosphate is likely to account for the increase in platelet aggregation that is observed as the hematocrit increases from 10% to 40% [62]. It was suggested [63] that the small size of the platelet relative to a red blood cell means that platelets may be concentrated in a peripheral layer at the blood-vessel interface, therefore increasing interaction with sub-endothelial molecules in the instance of an injured artery. The composition of a thrombus (in terms of fibrin to red blood cell ratio) is critically dependent on flow [63].

**Scales**  Human platelets are between 1.5-3 \( \mu m \) in diameter and have a range of cross sections shaped from a circle to a square (for details see references at [http://www.imm.org/Reports/Rep018.html](http://www.imm.org/Reports/Rep018.html)). Porcine platelets are similar in size and appearance to human platelets. Softeland *et al.* [64] highlight certain differences between porcine and human platelets. These largely relate to differences in response to various platelet activators such as adrenaline. Regarding the temporal scales, experimental evidence suggests human platelet activation by shear requires less than 10 seconds at all shear rates above threshold [65]. Activation in response to ADP or other platelet agonists is likely to occur on a similar timescale. The earliest events involved in the formation of platelet aggregates in response to adenosine diphosphate occur in the range of 3 to 10 seconds. Initial discoid shape is lost with a decay time of 2-3 seconds, followed by the extension of pseudopods with a time constant of 7-8 seconds. This exposes GPIIb/IIIa receptors allowing the formation of fibrinogen bridges between adjacent platelets, and therefore, aggregation.

A red blood cell rich thrombus can constitute anything from the aggregation of around 50 platelets up to a mass encompassing the diameter of the vessel lumen, i.e. 100\( \mu m \) to 4mm. A red blood cell rich thrombus represents the aggregation of at least 50 platelets. Given that aggregation of two platelets to form a doublet may...
take between 3 and 10 seconds from exposure to a given stimulus, it seems sensible to postulate that red blood cell thrombus formation takes somewhere in the region of minutes to form.

**Thrombus Modeling**

Platelet aggregation and adhesion are strongly affected by various mechanical factors amongst which the shear stress plays a dominant role. Numerous studies have shown that the shear stress enhances the thrombosis process and can directly cause platelet activation (see for example [66, 67, 68, 69]). In a flow field with high shear rate, the interaction between GPIb receptors and vWF multimers can initiate the tethering of circulating platelets to the vessel wall and to already adherent platelets [70]. In general, in the process of thrombus formation platelets get activated by abnormal physical stress. Whether this is high shear stress or low shear stress is still an open question in the community and different modeling approaches exist [71, 72, 68, 73]. With the activation platelets become "sticky", expressing surface proteins that bind to surfaces and other platelets. In regions with low stress the probability of sticking without being torn away by the blood flow is increased. Eventually, this leads to a filling and smoothing of rough surfaces as they are present in case of fresh wounds. However, also the activation dynamics (activation itself, and its decay over time) plays an important role as platelets loose their stickiness after being not stimulated for a while.

The most commonly used way to model thrombus formation is to treat blood as a continuous homogeneous medium defined by the continuity equation and the Navier-Stokes equations. The distributions of platelets and proteins relevant to thrombosis are described by the diffusion-convection reaction equations [74, 75]. In those works simplifications were made such as diffusion-limited rates of platelet-surface adhesion and constant platelet-surface reactivity allowing solutions even to be obtained in an analytical closed form [76]. An LBM approach was used in [72] to model flow and platelet-induced clotting, and in [73] flow-related clotting was modeled based on the residence time of concentrations of platelets using the same method for the advection-diffusion of the concentration fields.

Although tracking additional information like, e.g. “age” of the fluid, all of these models treat blood as a homogeneous medium and do not describe the behavior of individual blood constituents. However, blood is a highly complex suspension of chemically and electrostatically active cells suspended in an electrolytic fluid with active proteins and organic substances [77]. To gain insight into the nature of platelet activation, aggregation and adhesion, it would be beneficial to track individual platelets in the process of advection and activation. In [77] such a Lagrangian approach was demonstrated. Also a dissipative particle dynamics (DPD) approach to simulate platelet-mediated thrombosis can be found [78].

Recognizing the dynamics of activation of single platelets and the resulting sticking behavior as the most important feature of the complex thrombus formation system besides the blood flow, we have set up a model of massless non-interacting point particles that are convected in a flow field. The increase of the activity of these "platelets" is modeled as a threshold function of the shear stress the platelet is exposed to, from which the activation resulted as an integrated quantity. Competing with this is the decay of the activity with time. From the activation we then derived a function for the probability to stick in case it is next to a solid
surface which models both counteracting processes, stickiness and friction with the flow, respectively. Assuming a lattice based flow solver, lattice nodes are then solidified whose volume contains a number of sticking platelets larger than a certain threshold similar to solidification modeled in [79, 72].

Although such an activation dynamics model would offer the ability to simulate thrombus formation on the right time scale in a coupled simulation together with the other models, we, however, did not finalize this model to be part of the ISR model, also resulting from the non-triviality of tuning the parameters of such a model. Too less physiological data could be found in the literature to extract parameters and quantify the model, e.g. in respect to relaxation times of platelet activation.

The "Blob" model

From the above estimation of the temporal scales and the time-course of the processes involved in ISR (see Fig. 3.8) we see that the separation of thrombus formation and all successive processes involved in in-stent restenosis on the time-course is, strictly speaking, not absolutely given. However, in the models for in-stent restenosis we consider the final thrombus as an initial condition for the actual dynamic simulation of hyperplasia. This decision is not only motivated by the attempt to keep the complexity of the model controllable. It is also justified by the fact that in microscopy pictures of porcine samples at early stages of ISR the thrombus had already reached its final extent while the onset of neointimal growth was not yet clearly visible. When SMCs proliferate they migrate into a by then final thrombus tissue which transforms into fibrin-rich matrix. This fact allows for a different, much more pragmatic, very simple, but still convincing approach we called the Blob.

With Blob we ignore the above discussion on the need to resolve the dynamics of single platelets. Instead, Blob makes use of the observation that typically thrombus is found in areas which, before the thrombus formed, were characterized by circulating back flow of the blood. Regions of recirculation prevent a too quick escape of reaction products [80]. Also in [81] it was concluded that platelet aggregation occurred in annular vortices because of favorable combination of long residence time and frequent inter-particle collisions. In [71] the reattachment point was identified as first place for deposition in a stenosis model. However, experimental results show that maximum deposition occurs between the tubular expansion and the reattachment point [82] which can be explained by the fact that at the reattachment point the velocity is very low and the frequency of platelet-wall collisions is small whereas slightly upstream this situation has changed. This observation and explanation is supported by theoretical [83] and experimental [84] works. Recirculation zones can be described as sites of auto-catalytic augmentation of previously activated enzymes and sites of high residence time, both constituting ideal conditions for the deposition of platelets [71].

Given a steady flow field for a given geometry and Reynolds number, the simple Blob algorithm is defined so that in one shot all fluid nodes are marked as 'thrombus node' at which the radial velocity is opposite to the main flow direction. The rule reads therefore

\[
\text{type}(x_i) = \begin{cases} 
\text{thrombus} & \text{if } u(x_i) \cdot u_0 < 0 \\
\text{fluid} & \text{else} \end{cases} \forall x_i | \text{type}(x_i) = \text{fluid} \tag{3.6}
\]
Figure 3.9: Thrombii as defined by the Blob algorithm in comparison to results of other clotting models for a steady flow through a two-dimensional channel with a rectangular stenosis. a) Blob result for \( \text{Re} = 100 \), b) shear stress model with high shear stress threshold, also \( \text{Re} = 100 \) (from [85], colors improved), c) Blob result for \( \text{Re} = 550 \), and d) result of a shear stress model with medium threshold. b) and d) are from [85] where for other parameter settings also thrombus can be found in front and on top of the strut as seen in a) and c).

where \( \mathbf{u}(\mathbf{x}) \) is the flow velocity at node \( \mathbf{x} \) and \( \mathbf{u}_0 \) is the general flow direction, e.g. \( \mathbf{u}_0 = (u_x, 0, 0) \) in case of flow from left to right in a tube aligned with the x-axis.

**Validation** In Fig. 3.9 a number of results are shown for the application of (3.6) to flow fields through simple 2D test geometries. Although this method can be described as “quick and dirty” in comparison to more realistic modeling its results surprisingly compare very well in overall shape and extension with results obtained by Harrison et al. [85] for a modified solidification model based on a residence time approach where nodes of the LBM lattice were solidified whenever the residence time of the fluid is of sufficient magnitude. In the thesis by Harrison it is shown that these in turn are very similar to experimental results with clotting milk.

**Flow Solver** In order to obtain a flow field \( \mathbf{u}(\mathbf{x}_i) \) on which (3.6) can be applied a lattice-Boltzmann method (LBM) has been implemented. In section Sec. 4.3.1 the LBM will be introduced for 2D. However, the extension to 3D is straightforward. We used a D3Q19 model which means the velocity distribution function \( f_{i=0..N}(\mathbf{x}_i, t) \) is represented by \( N = 18 \) components each standing for the probability to find a fluid particle with velocity \( \mathbf{e}_i/\Delta t \) in one of the 18 lattice directions.
connecting neighbors and diagonal neighbors in the planes $xy$, $xz$, and $yz$.

As it is the computationally most costly part of the TF model we aimed at a very efficient implementation of the flow solver. Since the initial surface roughness due to the struts of the employed stent is small in comparison to the lumen width and the vessel is assumed to be approximately circular, the bulk part of the flow field is very much comparable to that of a Poiseuille flow in a perfect tube. We made use of this fact by omitting the inner bulk flow and only allocate memory and compute the evolution of the velocity distributions function for those fluid nodes that are further than $r_{\text{bulk}}$ away from the center of the tube. The thickness of the remaining cylinder hull is determined dependent on the roughness of its outer surface by

$$R_{\text{bulk}} = R_{\text{max}} - a \cdot R_{\text{min}}$$

(3.7)

parallelized and memory-optimized for sparse geometries similar to the implementation presented in [86]. The inner volume of the lumen was omitted and replaced by proper velocity-boundary conditions at the inner surface of the hollow cylinder whose wall contains the actually simulated fluid volume. The thickness of the cylinder wall is determined from the roughness of the inner vessel surface. The flow field is initialized with a Poiseuille profile for a pipe radius determined from the smallest lumen and a given Reynolds number. Fluid in the hollows starts from zero velocity. All parameters of the LBM simulation are automatically determined from the geometry and Reynolds number in a way that simulation time and memory allocation is minimized. After reaching a convergence criterion the Blob algorithm then determines thrombus nodes from the flow field and these are sent to the CxA.

**Leaky Interface Boundary Conditions for Hollow Cylinder Boundary**

Since nodes inside the bulk, i.e. nodes $n$ for which $|x_n - x_{\text{center}}| < R_{\text{bulk}}$, are only virtual, an appropriate boundary condition has to be applied at the adjacent fluid boundary nodes. One way to realize this would be to consider the boundary as an impermeable moving wall in which case a bounce-back condition for moving boundaries would be appropriate like that applied at the boundaries of the LBM fluid to the suspended particles described in Sec. 4.4.1. In this case, the velocity of the solid wall $u_b$ could be set assuming a parabolic flow in the maximum free lumen.

Aiming at flow fields in the fluid ring that are as close as possible to the case flow would be simulated throughout the whole lumen, another type of boundary condition was applied, however. We applied a boundary condition that weakens the impermeability of the interface between ring and bulk domain. Whenever probability densities $f_i(x_v)$ have to be propagated from a virtual $v$ inside the bulk to a fluid boundary node at $x_i$ these are obtained using

$$f_i(x_v) = (1 - c_{\text{leak}}) f_i^{eq}(u(x_v), \rho_0) + c_{\text{leak}} f_i^{eq}(u(x_i), \rho(x_i))$$

(3.8)

constituting a linear combination of the equilibrium values for the Poiseuille velocity $u(x_v)$ and density $\rho_0$ in the bulk and the equilibrium value for $u$ and $\rho$ at the fluid boundary node at $x_i$. With that, quantities depending on orders larger 0 of the deviation from the equilibrium, e.g. stresses, will not be resolved correctly, of course. However, we know that only a few nodes further away from the boundary these are restored in sufficient accuracy [87].
Figure 3.10: Flow, indicated by flow lines with same random seeds, and detected thrombus area (red) in a 2D test geometry (black=solid) of size 92x107. Flow is from left to right. Different boundary conditions were applied at the cylinder-bulk boundary: a) none (full domain flow), b) leaky bc, and c) bb for moving walls. When the bulk is removed, leaky bc’s result in a flow field (and hence thrombus) that only slightly deviates from the full domain simulation. The application of the impermeable bounce-back bc’s for moving walls leads to recognizable deviations from the full flow field. In d) the \( y \)-velocity of the fluid is shown at a distance of one lattice node along the upper bulk boundary.

In Fig. 3.10 flow through a 2D test geometry is shown together with the thrombus area as from application of (3.6) for the cases that a) the full domain was simulated, b) leaky boundaries were applied at \( R_{\text{bulk}} = 18 \) with \( c_{\text{leak}} = 0.95 \), and c) bounce-back for moving walls. The velocity at the boundary was set so that the \( x \)-component corresponds to the mean \( x \)-component of the velocity along the same line in the full domain case. The \( y \)-component was set to 0. In Fig. 3.10d) the \( y \)-component is shown as measured one lattice node into the fluid. For bounce-back, modeling a impermeable wall, it is almost zero, whereas for leaky bc we obtain a profile that is similar to that of the full domain flow. This results in a flow field that is similar to the full domain field. Therefore also the thrombii are very similar. Whereas the thrombus detected from the bounce-back flow field does deviate from the full domain version in recognizable extend.

In its 3D version, Blob was extended by a prune and enrich algorithm that, run after the thrombus node detection according to (3.6), removes single thrombus nodes which result from recirculation in features of the voxel domain that are only one lattice node wide. Mapping the volumes of the SMCs and the struts
of the stent to the voxel mesh these are unwanted consequences of the lattice representation of the flow domain. In the pruning step all thrombus nodes are removed, i.e. turned back into fluid type, that have less than 2 neighbor nodes of thrombus type. This could also be interpreted as washing out of clots that are not solid enough. In a subsequent step, agglomerates that are left are enriched by marking all fluid nodes as thrombus that have more than 5 thrombus nodes in their D3Q19 neighborhood. This effectively fills possible holes in thrombus compounds and results in compact and smooth thrombii. In Fig. 3.11 the result of applying the Blob algorithm including the prune and enrich procedure are shown for the geometry also used for a 3D ISR CxA for which preliminary results are shown in Sec. 3.4.2. To realize the TF kernel, the Fortran code for the sparse and parallel implementation of the flow solver, and parallelized codes for the Blob algorithm and the prune and enrich smoothing procedure was completed with C++ and Java wrappers where entrances and exits to the framework were realized. The TF kernel functions as part of the generation of the initial conditions, more precisely, as a stage 3 after generation of the cells and the deployment of the stent. Thrombus nodes constitute solid obstacles for the bulk flow whereas SMCs are allowed to migrate into the thrombus. For the drug diffusion model the thrombus domain is indistinguishable from the SMC tissue and receives both as a combined domain.

3.4.2 3D Results

For the 3D ISR CxA, again, MUSCLE was employed to pass information between kernels using conduits and a modified generic mapper agent (in instances when
multiple inputs were required to calculate one output). A simplified geometry of the BiodivYsio stent was deployed into a three dimensional representation of the vessel wall, causing laceration of the internal elastic lamina, thus permitting underlying SMC agent proliferation. Figure 3.12 depicts preliminary output from the 3D CxA in which neointimal formation is clearly visible in areas adjacent to stent struts. The wall shear stress distribution can be visualized at the vessel surface.

3.5 Conclusions

In this work we have shown how Complex Automata methodology can be applied in a challenging multi-scale model of in-stent restenosis. In particular, we describe implementation of the coupling of three different subprocesses which operate on different time scales. The model has been realized employing a CxA-dedicated coupling library (MUSCLE), and the results presented in this chapter demonstrate that the CxA model can be successfully implemented within this framework in both two and three dimensions.

Although the individual models are at a relatively early stage and the current CxA is simple in nature, certain emergent behaviors are already apparent. For example, proliferation begins in response to injury, peaking at approximately 20 days following deployment in the absence of drugs (Figure 3.6). We are currently in the process of running additional simulation series, to validate the CxA against a biological data-set obtained from in vivo and in vitro experimentation using stented porcine arteries. In particular, we aim to characterize restenosis behavior as a function of injury index [88] and to investigate the positive correlation between injury and restenosis.

This first realization of the coupled CxA is an important milestone on the journey towards a full multi-scale model of in-stent restenosis. Future developments will require development of the single scale kernels. Implementation of more complex rulesets will allow inter-cellular signalling pathways and the effects of deep injury to be modeled. We also aim to achieve more realistic local hydrodynamics by integrating a full pulsatile flow model. Moreover, the current CxA can be improved further by including extra kernels to model processes such as endothelial loss and regrowth.
Figure 3.12: The 3D ISR model in which a simplified representation of the BiodivYsio stent is deployed. (a) Neointima is observed forming around the stent struts. Also shown are flakes of thrombus (red). (b) Drug distribution in the SMC tissue (blue - low, red -high). (c) Flow lines near the wall colored according to flow velocity. SMCs are colored according to the wall shear stress (blue - low, red - high).
Chapter 4

Suspension Flow

In this chapter a multi-scale system will be described in which we apply a coupling scheme which is different to those used in the ISR model in the previous chapter. Focusing on simulations of macroscopic flow behavior of non-Brownian monodisperse hard-sphere suspensions we will investigate the application of the Heterogeneous Multi-scale Method (HMM) \[7\] to this type of systems.

The original aim of the work on suspension modeling has been to provide a submodel for detailed simulations of blood flow on the scale of the blood cells in a layer along the boundary to the SMC domain which would then have been coupled to the coarser bulk flow. However, instead of appropriate coupling of two models in a multi-domain approach (as for example in \[89\]) we consider the flow domain to be a single domain where a macro-scale model for the non-Newtonian hydrodynamics is coupled everywhere to a micro-model for the suspension’s micro-dynamics (see Chapter 2 and the classification of multi-scale problems in Fig. 2.5). Each of the model’s spatio-temporal domains \(A_{m,M}(\Delta x_{m,M}, \Delta t_{m,M}, L_{m,M}, T_{m,M})\) is chosen such that typical temporal, \(\tau_{m,M}\), and spatial scales, \(\xi_{m,M}\), of the phenomena can be resolved on the micro scale (denoted by subscript \(m\)), or macro scale (denoted by subscript \(M\)), respectively.

The remainder of this chapter will be as follows. In Sec. 4.1 an overview of the rheology of hard-sphere suspensions is given in the aspects that are important to our simulation approach. In Sec. 4.2 the idea of how suspension flow can be simulated using the HMM approach will be presented. Subsections discuss the scales and their separation in this type of systems (Sec. 4.2.4) and consequences of the choice of simulation methods. Also, the proposed multi-scale model will be discussed with the help of the scale separation map and sub-execution loops, helpful tools offered by the CxA formalism. The subsequent section will discuss the modeling of the fully resolved micro-scale of the suspension with the Lattice-Boltzmann method (LBM). In that two major improvements regarding this method are proposed. First, Lees-Edwards boundary conditions (LEbc) are presented which remove the side-effects that walls have in a commonly used Couette type of shear flow on the microstructure. Second, a correction to the momentum exchange algorithm (MEA) commonly used in LBM suspension methods is proposed that restores Galileian invariance of the particle dynamics. Section 4.5 is dedicated to the phenomenon of shear-induced diffusion in non-Brownian suspensions including a description of diffusivity measurements on the micro-scale as well as the appropriate modeling of
advection-diffusion with a flow-aligned anisotropic diffusivity tensor on the macroscale. Section 4.6.2 discusses the possibility of using a database coupled in between micro and macro-models which significantly reduces the number of micro simulations to be launched. In the last two sections simulation results for macroscopical flow through a straight channel are presented and discussed in a comparison to other models and experiments addressing the same flow problem.

4.1 Rheology of Hard-Sphere Suspensions

The dynamics of dense liquid-particle suspensions is of great importance for many physical, biological and industrial processes. Suspension behavior is rich in rheological aspects triggered by various properties such as particle-fluid volume density, particle shape, size distribution and properties of the suspending fluid [90, 91, 92, 93]. Depending on the situation, the apparent viscosity \( \nu_{\text{app}} \), the crucial property describing a (complex) fluid at the macroscopic scale in general, is a function of almost all the parameters characterizing these properties. The effect of the volume ratio \( \phi \) is best described by the Krieger-Dougherty relation

\[
\nu_{\text{app}} = \nu_t \left(1 - \frac{\phi}{\phi_{\text{max}}} \right)^{-[\eta] \phi_{\text{max}}}
\]  

(4.1)

which describes the divergence of \( \nu_{\text{app}} \) when \( \phi \) gets closer to the maximum density \( \phi_{\text{max}} \) at which the suspension jams. For a system of disks in 2D \( \phi_{\text{max}} = \frac{\pi}{4} \approx 0.785 \) which is the density of the densest square lattice packing and which lower than the maximum packing density of \( \approx 0.95 \) for hexagonal packing of monodisperse particles. The square lattice packing gives the maximum packing density at which layers of particles still can move in a sheared flow. When particles get stuck in formations that percolate the whole system no further movement is possible. This leads to a divergence of the apparent viscosity scaling like a power law \( \nu_{\text{app}} \sim (\phi - \phi_{\text{max}})^{-\alpha} \) with a critical exponent \( \alpha \approx 2 \) in 2D. In case of non-spherical particles this effect sets in at even lower \( \phi \) whereas suspensions with a non-Dirac distribution of particle sizes have higher \( \phi_{\text{max}} \) [94]. The factor \([\eta]\) in (4.1) is the intrinsic viscosity which depends on shape and additional interactions between the particles. For spherical non-colloid particles \([\eta] = 2.5 \). In the limit of small \( \phi \) (4.1) reproduces the linear increase \( \nu_{\text{app}} = \nu_t (1 + 2.5\phi) \) as derived by Einstein [95, 96] under the assumption that particles do not interact hydrodynamically and Brownian motion can be neglected.

Central role in this work plays the influence of the shear rate \( \dot{\gamma} \) on \( \nu_{\text{app}} \). Different types of viscosity behavior can be observed in an unit-less parameter space described by the Reynolds number \( \text{Re}_p = 4\dot{\gamma} R^2 / \nu_t \), expressing the ratio of inertial forces to viscous forces, and the Peclet number \( \text{Pe}_p = 4\dot{\gamma} R^2 / D \), the ratio of advection time scale to that of diffusion (where \( \dot{\gamma} R \) denotes a typical velocity, \( R \) the particle radius, \( \nu_t \) the kinematic viscosity of the fluid, and \( D \) the Stokes-Einstein diffusivity of a single particle).

In general, from low to high \( \text{Pe}_p \) a hard-sphere suspension shows the following behavior [97] (see Fig. 4.1): for small \( \text{Pe}_p < 1 \) thermal fluctuations govern the system dynamics and lead to an increased viscosity due to the increased effective volume of the particles. In this regime \( \nu_{\text{app}}(\text{Pe}_p) \) shows a high plateau. Increasing the shear rate \( \dot{\gamma} \) to mediate \( 1 < \text{Pe}_p < 50 \) causes the viscosity to exhibit
shear-thinning due to the decreasing influence of Brownian motion, followed by a Newtonian plateau where particle dynamics is governed by Stokesian drag forces and particles are believed to form layers that are easily sheared [98, 94]. Increasing \( \dot{\gamma} \) further causes the suspension to thicken as consequence of a breaking of the layers in a order-disorder transition (ODT) [98, 99] where particles now collide with smaller impact parameters and therefore higher momentum exchange. Going to even higher \( Pe_p \) on the microstructural level this is accompanied by ever closer approaches of particles and the formation of non-compact elongated (hydro-)clusters whose sizes in case of dense suspensions eventually diverges in the process of jamming [100, 101, 102, 103].

The question about the origin of shear-thickening is still not answered consistently. Considerable experimental effort has been applied through the study of mono-disperse steric stabilized particles and charged latices. However, these are contradictory in some aspects. Theoretical approaches are limited by the need to include both many-body non-equilibrium distribution functions and hydrodynamic interactions.

Both processes might contribute to the increase in \( \nu_{app} \), the breaking of particle layers that might have formed as a consequence of leveled Brownian and hydrodynamical forces and the formation of clusters at higher shear rates as a consequence of particle inertia and the dissipative nature of the hydrodynamical interaction between particles. The two components of the viscosity, Brownian thermodynamical (elastic-like) and hydrodynamic (viscous-like), respectively, differ substantially between clustering theory and the order-disorder transition theory. The thermodynamical component decays as \( Pe_p^{-1} \) [105] (constant elastic stress) whereas the hydrodynamical component is clearly associated to the onset and course of shear-thickening. Based on our simulations of sheared non-Brownian suspensions presented in the next sections and the last chapter of this thesis we see a clear correlation between shear-thickening and the formation of particle clusters. Brownian forces, however, were not considered and, related to that or not, we could not observe layering at small or medium \( Pe_p \). Also the dimensionality might play a role here as the additional degrees of freedom in three dimensions will have impact.

Figure 4.1: General course of the apparent viscosity of a sheared suspension against the shear rate which can be sub-divided into five regions of typical characteristics, most notably the regimes of shear-thinning (II) for Brownian suspensions and the shear-thickening (IV). Different curves correspond to different particle volume fractions \( \phi \) with larger absolute and change in viscosity for higher \( \phi \). Figure reproduced from [104, 97].
on the dynamics in the shear plane compared to the pure two dimensional system where particle movement is much more restricted. It might be that particles in a 3 dimensional suspension at small shear rates use the additional dimension to arrange in two dimensional layers.

For small perturbations from equilibrium Brady suggested a master curve for Brownian hard-sphere suspensions at any volume fraction can be produced by plotting reduced viscosity against $\text{Pe}_s = R^2\dot{\gamma}/D_s$ [106], the Peclet number calculated using the short-time self-diffusivity, $D_s$. The short-time self-diffusivity quantifies the instantaneous mobility of a particle in the suspension. Its value is reduced below that for a single particle in the solvent because hydrodynamic interactions with adjacent particles in the suspension slow particle motions. Therefore, only at infinite dilution is the short-time self-diffusivity equal to the Stokes-Einstein diffusivity $D$. As we do not consider Brownian fluctuations in this work, and are more interested in the regime far from equilibrium, the scaling can not be extended.

Depending on the volume fraction $\phi$ two distinct shear thickening regimes have been observed. A critical value $\phi_c$ exist below which only continuous shear-thickening occurs, characterized by a smooth increase with $\dot{\gamma}$. For $\phi > \phi_c$ a sudden increase in $\nu_{\text{app}}(\dot{\gamma})$ can be observed. In the literature values for $\phi_c$ vary with suspension type and particle size distribution between $0.52 \leq \phi_c \leq 0.575$ [107, 108, 109]. Any colloidal interaction can drastically effect $\phi_c$ and weaken or amplify the effect of discontinuous viscosity increase. In the present work we simulated two-dimensional suspensions at $\phi$ which apparently in all cases was below the critical $\phi_c^{(2D)}$. This was probably due to the fact that the flow did not reach high enough $\dot{\gamma}$ for $\phi > \phi_c^{(2D)}$ in the diffusion coupled simulations presented to see a crossover to discontinuous shear-thickening. Another reason for not seeing discontinuity in $\nu_{\text{app}}$ might be because Brownian motion was neglected. For volume fractions that clearly showed discontinuous shear-thickening the increase of $\nu_{\text{app}}$ was accompanied by an increase of the viscous-like viscosity component and a slow decay of the stress after shear cessation [105] giving rise to the assumption that when particles are densely packed Brownian motion again might play an important role.

In search for the origins of clustering inertia of the particles can be identified as playing a central role as we observe Stokes number scaling of the shear-thickening curves, i.e. $\nu_{\text{app}} = f((\rho_s/\rho_f)\dot{\gamma})$, shown in Fig. 5.5. The influence of particle inertia on microstructure and shear-thickening will be subject of Sec. 5.

Another question remaining open is whether ideal non-colloidal hard-sphere suspensions jam at very high shear rates at all. Some experiments on real suspensions give rise to the assumption that the shear-thickening regime again is followed by a shear-thinning region [90, 108]. Against that can be argued that while experimenting with real suspensions the influence of the experimental setup and additional forces between the particles can not be absolutely excluded in most cases [110, 111, 112, 109]. Also, in our simulations we see a continued increase of $\nu_{\text{app}}$ up to the highest shear rates allowed by the methods we used [113] in contrast to comparable simulations which resemble more the experimental setup of real viscometers [114, 115, 116].

Simulation approaches to address suspension rheology range from, in decreasing order of abstraction, (i) modeling the suspension as non-Newtonian fluid, (ii) continuum models of two-phase flows and (iii) fully resolved particulate suspen-
sions where the fluid dynamics is governed by the Navier-Stokes equations and the motion of the solid particles is governed by Newton’s laws. In many practical problems, the more detailed level of description turns out to be necessary and Brownian motion has to be incorporated [117] for small particle sizes at small Reynolds numbers.

4.2 HMM Modeling of Suspension Flow

4.2.1 Description of the Problem

The emergent rheology on the macro-scale is dictated by the details of fluid-particle, and particle-particle interactions on the micro-scale. In systems where the typical spatial scale on the particle level is much smaller than that of macroscopic properties the scales can be split using a heterogeneous multi-scale method (HMM) approach [7]. In a simple test case for this multi-scale approach we consider the pressure driven flow of a 2-dimensional suspension of hard-discs in a straight channel with a wall-to-wall distance $L_y = 5 \cdot 10^{-3} \text{m}$ and a no-slip condition at the boundaries. In $x$-dimension the channel is infinitely long implemented by means of periodic boundary conditions. As the actual computational domain we consider only a length of $L_x = \Delta x_M$ of the channel where $\Delta x_M$ is the spatial resolution of the macro model we will use. With that we impose a symmetry on the system so that the macroscopic fluid velocity has a zero $y$-component under the assumption of non-compressible flow. With this we rule out any effects related to non-laminar flow. We will discuss the possible extension to this class of flow problems in an outlook at the end of this chapter. As any mapping of additional effects between the scales might be another source of scale splitting error in the current work we enforced laminar flow to minimize the complexity already present in the coupling of the models described here.

**Momentum Transport: Viscosity** Complex fluids, in which constituents on the micro-scale interact in various ways leading to a shear-dependent microstructure, mostly show strong non-Newtonian behavior. That is, in local shear stresses

$$\tau(x_M) = \frac{\nu_{\text{app}}(x_M)}{\rho(x_M)} \frac{\partial u}{\partial x}\bigg|_{x=x_M} = \frac{\nu_{\text{app}}(x_M)}{\rho(x_M)} \dot{\gamma}(x_M)$$

(4.2)

the local apparent viscosity $\nu_{\text{app}}$ of the suspension strongly depends on the local shear rate $\dot{\gamma}(x_M)$. Furthermore $\nu_{\text{app}}$ is a function of the local volume fraction of the solid phase so that we have

$$\nu_{\text{app}}(x_M, t) = F_{\nu_{\text{app}}}^{\nu_1, \rho_f, \rho_s}(\dot{\gamma}(x_M, t), \phi(x_M, t)).$$

(4.3)

Additionally, $\nu_{\text{app}}$ strongly depends on parameters like the kinematic fluid viscosity $\nu_1$, the fluid mass density $\rho_f$, and the solid phase mass density $\rho_s$. Although there is tremendous progress in understanding the rheology of hard-sphere suspensions, so far scaling and other more complete relations could only be derived under conditions where the hydrodynamic interaction between suspended particles was simplified. Most of them are only valid in the limits of small $\dot{\gamma}$ and $\phi$. A full functional form for the whole variable space of interest does not exist and the higher $\dot{\gamma}$ and $\phi$ the
more open are the discussions on how \( \nu_{\text{app}} \) behaves in this regime. In Sec. 4.1 we discuss the rheology of hard-sphere suspensions regarding these aspects in relevant detail. In case of suspensions that deviate from the ideal monodisperse hard-spheres even less consensus on the form of \( \nu_{\text{app}}(\dot{\gamma}, \phi) \) exists. If we want, and this is the further aim of this work, to develop a tool to study the rheology of various types of suspensions involving numerous parameters \( \{P\} \) characterizing properties such as shape and size distributions, additional inter-particle forces and maybe even dynamic processes like the shear-stress activation of inter-cellular forces in blood, we simply have to conclude that full knowledge about the function \( \nu_{\text{app}} = F_{\nu_{\text{app}}}(\dot{\gamma}, \phi) \) does not exist and the constitutive equation as (4.2) cannot be given in complete form (see e.g. [97] for a review on the rheology of dense suspensions).

**Particle Transport: Diffusivity** In the presence of variations in the solid volume fraction \( \phi(x_M) \), e.g. particle concentration, a net flux of particles will try to relax the concentration gradient according to Fick’s law of diffusion

\[
J = -D \frac{\partial \phi(x)}{\partial x} \bigg|_{x=x_M}
\]  

(4.4)

However, the diffusivity tensor \( D \) is, similar to the momentum transport coefficient \( \nu_{\text{app}} \), dependent on the shear rate \( \dot{\gamma} \) and the volume fraction \( \phi \). In the limit of vanishing Brownian fluctuations in the momentum exchange with the fluid, the case we consider in this work, there is no force that could drive the diffusive motion of the particles through the suspension. With the resulting zero diffusivity a concentration gradient would not be enough to cause a flux of particles. However, when the suspension is sheared particles will collide with each other and, like random walkers, will be progressively displaced from their origin. This results in a dynamics that is found to be diffusive [118]. This so-called shear-induced diffusivity is a strong function of the shear rate and the volume fraction, e.g.

\[
D(x_M, t) = F_{D}^{\{P\}}(\dot{\gamma}, \phi)(\dot{\gamma}(x_M, t), \phi(x_M, t))
\]  

(4.5)

where \( \{P\} \) is again the set of parameters that characterizes the exact type of suspension. Here, the situation is similar to that of \( F_{\nu_{\text{app}}}^{\{P\}}(\dot{\gamma}, \phi) \): we know its behavior only in some limits and then mostly only for the case of monodisperse hard-spheres [119] (see also Sec. 4.5 for a brief discussion on that). However, because a non-trivial feedback loop exists between shear rate \( \rightarrow \) diffusivity \( \rightarrow \) volume fraction \( \rightarrow \) viscosity \( \rightarrow \) shear rate, we cannot just neglect diffusive effects. We rather expect non-trivial particle distributions dynamics \( \phi(x_M, t) \).

### 4.2.2 The Heterogeneous Multi-scale Method

The Heterogeneous Multi-scale Method [7] is a general methodology grouping approaches to the efficient numerical computation of multi-scale and multi-physics problems (see Sec. 2 for a classification of multi-scale problems). The fact that the involved submodels might be of very different nature gave rise to the characterization of being “heterogeneous”. They all have in common that the macro-scale model, whose observables and dynamics we are interested in, lacks completeness
in one or more aspects. That could be in case of the non-existence of constitutive equations or if the macro-scale is not valid due to localized singularities, e.g. boundaries. To provide the missing information, micro-models are employed which typically are descriptions of the underlying process on a much smaller scale. HMM is thus a strategy for designing multi-scale algorithms that are driven by data. A very typical and descriptive example for a HMM coupling could be a molecular dynamics (MD) model with which the state space of a gas system of specifically interacting atoms is sampled for a given subset of the local state variables of a macroscopical flow field. The subset is used as boundary conditions to the MD micro-model which in turn completes the set of state variables acting as a replacement (or other form) of a constitutive equation that might not be available but is needed on the level of the Navier-Stokes description of the fluid. Although the micro-scale model contains all the information, its level of detail makes it too costly to be applied to the whole macroscopic domain. Based on the multi-scale system we investigated in the current work we will also see that this argument can even be extended. Due to the possible limited validity range of the methods used in a micro-model, the concept of independent micro realizations that communicate in a compressed way (exploiting the symmetries of the physical system) via a macro-scale makes simulations spanning the complete range of scales feasible in the first place.

Several other or related existing multi-scale methods fall in the HMM category such as ab initio molecular dynamics [120], quasi-continuum methods [121] and projective methods for multi-scale systems [122]. The approach is different to the more traditional multi-scale methods such as multi-grid, fast multipole method, adaptive mesh refinement, or wavelet representation that make explicit use of multi-scale decomposition of functions and signals, and thus, in contrast to HMM, resolve the details of the solutions of the micro-scale model as general purpose micro-scale solvers.

A trade-off can always be found between completeness of the information gained, accuracy of the solution, computational efficiency and straightforwardness of the implementation. Using a HMM strategy we have to cut back in the following two main aspects. First, we won’t gain information on the micro-scale level everywhere but only on parts of the physical domain. For the rest of the macro-scale domain we have to be satisfied with the solutions of the macro-scale solver and/or interpolated solutions of the micro-scale model. Also, due to the coupling of different types of models, an estimation of the error we make on the macro-scale is not always simple. Second, we must explore possible special features of the micro-scale problem, such as scale separation and self-similarity. The HMM strategy only gives us a starting point, applications of HMM to specific problems can be a highly non-trivial task. A great understanding of the multi-physics system on all involved scales is needed, formulation issues have to be resolved and technical tools have to be developed along the way.

4.2.3 Submodels and Coupling

With the ideas of HMM in mind we will now construct an abstract computational model for flow of a non-Brownian monodisperse suspension of hard spheres through a macroscopic channel. As elaborated in Sec. 4.2.4 the scales of the micro-
and macrodynamics are separated and lead to a classical micro-macro coupling. In Sec. 2.2.2 we have identified this type of scale separation as interaction region 3.1 in scale separation map in Fig. 2.3. This corresponds to the case that the full spatio-temporal extend of MiS is smaller than a single spatio-temporal step of MaF, i.e. $T_m \times L_m < \Delta t_M \times \Delta x_M$. We also learned that when we face the problem of incomplete constitutive equations we need to obtain the missing information by means of explicit simulations of the microdynamics. As another property, we see that all the submodels are defined on the same single domain (sD). Using the classification of multi-scale systems we have motivated in chapter 2 the present multi-scale system falls into the group of hierarchical coupling problems (see Fig. 2.5). The HMM approach will be an applicable guideline to construct an appropriate multi-scale model.

Besides the macro-model for the fluid dynamics MaF a model MaS is introduced on the macro-scale in order to handle the evolution of the local volume fraction $\phi(x_M)$ consisting of advection with the velocity field $u(x_M)$ and diffusion induced by a local shear rate which is a function of the velocity gradient field, e.g. $\dot{\gamma}(u(x))$.

In Fig. 4.2 an HMM suspension is depicted in a way as to show the coupling between the macro-scale level consisting of a non-Newtonian fluid model and an advection-diffusion model, and a fully resolved suspension model on a micro-scale level. At every point on the macro-scale lattice and time step the macro-models need information on collision parameters which can not be provided due to the lack of constitutive equations. Instead, a “micro pocket” is opened up, running a fully resolved simulation of suspended particles in a shear flow. The initial and boundary conditions of the micro simulation are determined from the local macroscopic shear rate $\dot{\gamma}(x_M, t_M)$ and volume ratio $\phi(x_M, t_M)$, i.e. the microscopic shear rate and the number of particles in the micro domain given by

$$\dot{\gamma}_m = s^{-1}\dot{\gamma}(x_M, t_M)$$  \hspace{1cm} (4.6)
$$N_p = N_p(\phi(x_M, t_M)).$$  \hspace{1cm} (4.7)

We will discuss this mapping in some more detail in the section on the preparation of the initial conditions of the micro simulations.

From the simulation run of the micro-model the apparent viscosity $\nu_{\text{app}}(\dot{\gamma}_m, N_p)$ and anisotropic diffusivity tensor $D_m$ are determined. They are converted into the macro-scale unit system according to

$$\nu_M(x_M, t_M) = s_{\nu} \cdot \nu_{\text{app}}(\dot{\gamma}_m, \phi)$$  \hspace{1cm} (4.8)
$$D_M = s_D \cdot D_m$$  \hspace{1cm} (4.9)

and then passed as collision parameters to the macro-scale model. Also, $u(x_M, t_M)$ is passed from MaF to MaS within the macro-scale spatio-temporal domain, therefore without conversion. In this formulation we introduced a scale factor $s_Q$ which is a scalar constant translating a quantity $Q$ between the units of the micro and macro model. It is defined as

$$s_Q = \frac{Q_M}{Q_m}$$  \hspace{1cm} (4.10)

where $Q_m$ and $Q_M$ are the numerical representations of the quantity $Q$ in the unit base formed by $\Delta x_m, \Delta t_m$ and $\Delta x_M, \Delta t_M$, respectively. The scale factor $s_Q$ therefore can be expressed in terms of $\Delta x_{m,M}, \Delta t_{m,M}$, and $s_x$ and $s_t$. In Table 4.4
Figure 4.2: HMM type of coupling for the multi-scale simulation of the flow of a suspension. On the macro-scale a non-Newtonian fluid dynamics model (MaF) computes the evolution of the suspensions pressure and velocity fields. Besides that, an advection-diffusion model treats the evolution of the particle concentration field, i.e. the local volume ratio, on the same macroscopical scale. As no constitutive equations are present to compute the collision parameters $\nu$ and $D$ needed at every spatio-temporal point $(x_M, t_M)$, a “micro pocket” is opened where a fully resolved suspension simulation is run. The initial and boundary conditions, i.e. shear rate $\dot{\gamma}_m$ and $N_p$, of the micro simulation are determined from the local macroscopic $\dot{\gamma}_M(x_M, t_M)$ and volume fraction $\phi(x_M, t_M)$ (a). From the simulation run apparent viscosity $\nu_{\text{app}}$ and diffusivity tensor $D$ are determined and passed to the macro-scale (b). Also the local macroscopic velocity is passed from MaF to MaS.
this is done for the quantities describing the suspension system together with the actual numerical values for the scales given in Table 4.3.

As useful tools to construct an algorithmic description of a complex multi-science multi-scale system, the CxA formalism offers the concepts of the scale separation map (SSM), a generic sub-execution loop (SEL) and coupling templates. The SSM will prove its usability in the next subsection where we discuss the scales that have to be resolved and their coupling. The coupling will be specified at hand of coupled SEL’s in the subsection thereafter.

4.2.4 Scales of the Problem and their Separation

The scales to be resolved are not only dictated by the processes in the system that should be modeled. In all cases of multi-scale modeling the limitations of the chosen models and methods itself have to be taken into account carefully because their application might be limited to a certain range of $\Delta x$ or $\Delta t$, or the ratio $\Delta x/\Delta t$. Typically, outside the validity range of a model its results become meaningless, either because the assumptions it is based on are not true anymore, or a numerical method involved becomes unstable.

On the spatial scale, the two central properties that define the upper and lower limits of the scales to be resolved are the size of the particles and the dimension of the domain. The particle diameter in the experiments we will compare the results of the simulations to is in the $\mu m$ range \[123, 124\]. We have chosen a particle radius $R_p = 3.15 \mu m$. To resolve the fluid in sufficient detail with LBM a resolution of $1 \mu m$ is appropriate \[125\], we thus define $\Delta x_m = 1 \mu m$. Clusters of particles that form at higher $\dot{\gamma}$ are spatial structures giving rise to a correlation length $\xi_m$ (see Chapter 5). They are believed to be the origin of changes in mesoscopic properties such as the diffusivity $D_{xy}$ of particles or the apparent viscosity $\nu_{app}$ (see discussions in Sec. 4.1 and Chapter 5). Choosing a mass-density ratio of $M = \rho_s/\rho_f = 10.0$, for $\phi_0 = 0.40$ we find a typical cluster size of $N \approx 10$ for the highest shear rates feasible with LBM (see Chapter 5). With an observed fractal dimension of $D = 1.15$ and a typical orientation angle of $\approx 45^\circ$ we can estimate a typical size of the clusters in the direction of the velocity gradient to $2^{-1/2}N^{1/1.15} \cdot 2R_p \approx 36\Delta x_m$. To lower the probability that clusters percolate the system we set the spatial extend of the micro system to $L_m = 72\Delta x_m > \xi_m$. Under the Mach number limitation of the LB method, i.e. $u < 0.8c_s \approx 0.46$, and practicable fluid viscosity $\nu_t = 0.1$ this allows us to reach a maximum Reynolds number of $\text{Re}_p = 4R_p^2\dot{\gamma}/\nu_t \approx 2.535$. This is approximately an order of magnitude higher than the Reynolds number at which shear thickening sets on. These settings can thus be used to sample the state space of the suspensions in a sufficient $\dot{\gamma}$-range to find interesting behavior.

On the macro-scale, the width of the channel is $L_M = 50mm$ and we have chosen a spatial resolution $\Delta x_M = 1.25mm$ which should be fine enough to resolve the properties of the macroscopic flow which will be laminar due to very low flow Reynolds numbers and due to the symmetry enforced by setting $L_{M,x} = \Delta x_M$. For the spatial scale factor,

$$s_x = \frac{\Delta x_M}{\Delta x_m}, \quad (4.11)$$

we get $s_x = 1.25 \cdot 10^3$. 
The advection-diffusion model \texttt{MaS} relies on the velocity field $u(x_M, t_M)$ provided by \texttt{MaF} and its spatio-temporal resolution. It will also output information on the local volume fraction $\phi(x_M, t_M)$ in the same spatio-temporal resolution. Consequently, it is reasonable to write $A_{\text{MaS}} = A_{\text{MaF}} = A(x_M, L_M, t_M, T_M)$ although the dynamics of $\phi(x_M, t_M)$ will be solved using a Lagrangian point particle representation (LPAD) for $\phi(x_M, t_M)$ whose actual spatial resolution and timestep are smaller.

In contrast to the freedom in setting $\Delta x$ according to the scales of the physical system, the temporal scale is dictated by the range of fluid viscosities $\nu_f$ we can simulate with LBM. Using a LBGK relaxation collision, due to the limited stability of the LBM the relaxation parameter is limited to $0.5 < \tau < 2$. We have chosen $\frac{1}{100} < \nu_f < \frac{2}{5}$ as a “save” range in units of the macro-system, $\Delta x_M^2 \Delta t_M^{-1}$. Expecting the apparent viscosity of the micro-system to be in the range of $1 \cdot \nu_f < \nu_{\text{app}} < 50 \cdot \nu_f$, we chose a viscosity scale factor,

$$s_{\nu} = \frac{\Delta \nu_M}{\Delta \nu_m} = \frac{\Delta x_M^2 \Delta t_M^{-1}}{\Delta x_m^2 \Delta t_m^{-1}}, \quad (4.12)$$

of $s_{\nu} = 1 \cdot 10^1$ to map viscosities from the micro-scale to the macro-scale by $\nu_M = s_{\nu}^{-1} \nu_m$. This results in a time scale factor,

$$s_t = \frac{\Delta t_M}{\Delta t_m} = s_{\nu}^{-1} s_x, \quad (4.13)$$

of $s_t = 1.5625 \cdot 10^4$.

The typical time scale on the micro level $\tau_m$ is given by the shear rate set. From autocorrelation measurements of the apparent viscosity we find that the correlation time scales linearly with $\dot{\gamma}$ and can be estimated as 1 in normalized strain units. A maximum shear rate $\dot{\gamma}_{\text{max}} = Ma_{\text{max}} c_s L_m^{-1}$ results from the Mach number constraint and the system size $L_m = 72$ used. For a range of shear rates $10^{-8} \Delta t_m^{-1} < \dot{\gamma}_m < 6.42 \cdot 10^{-3}$ we estimate

$$10^8 > \tau_m > 1.56 \cdot 10^2. \quad (4.14)$$

Averages of mesoscopic properties will be taken over a normalized strain of 100. Assuming equivalence of averages in space and time we don’t extend $T_m$ for the considerations here. We still can assume that averaging out fluctuations at this instance does not influence the macro-dynamics. To picture the average case at later stages of the channel flow simulations we consider a typical micro simulation time $T_m = 100 \cdot (10^{-4})^{-1} = 10^6$ for the estimation of the computational benefit using the HMM approach for such simulations.

The maximum time scale of the system $T_M$ depends on what feature of the system we want to investigate. In this work we focus on the macroscopic equilibrium of a suspension flowing through a channel. The relaxation dynamics of the fluid is relatively fast and the dynamic equilibrium is reached after a few $10^3 \Delta x_M$. Base on that, we assume $T_M = 1 \cdot 10^4 \Delta t_M$. In contrast, the diffusive motion of suspended particles is much slower and particle concentration profiles develop on a larger time-scale only. The fluid adapts rather quickly to the change in the distribution of particles. This type of scale separation will be discussed in section Sec. 4.8.1 where we motivate a legitimate rescaling of $D_M$ to accelerate the diffusive dynamics
without violating the scale separation. As long as the separation is intact, this will change only the time scale the system reaches the macroscopic equilibrium. The equilibrium itself will not be biased. For the maximum simulated time we thus still consider $1 \cdot 10^4 \Delta t_m$.

In Table 4.3 all dimensions are summarized. In Table 4.4 the scale factors are listed defining the conversion of quantities from one level to the other.

The scale-separation map in Fig. 4.5 shows the micro and the macro-models placed according to the scales we have discussed. On the lower left (small spatial and temporal scale region) we find the fully resolved suspension model $\text{MiS}$ which itself consists of a LBM solver and the Newtonian dynamics solver for Lagrangian particles (see Sec. 4.4). The dynamics of the particles in the micro-model are described on a relatively fine spatial scale. Also, their dynamics is resolved on a finer temporal scale than the fluid to increase the numerical stability of the integration scheme (see Sec. 4.4.1). To reflect the dynamically adapted time-step the particle box was symbolically extended to lower temporal and spatial scales. However, because interaction between fluid and solid phase is realized via a grid representation of the particles we will assume the spatio-temporal domain of $\text{MiS}$ to be $A_{\text{MiS}} = A(\Delta x_m, L_m, \Delta t_m, T_m)$.

Looking at the relations expressing the scale separation, $L_m < \Delta x_m$ and $T_m < \Delta t_m$, we find the typical micro-macro coupling, identified as interaction region 3 in the classification of CxA in chapter 2.

Based on the points on the scale map defining lower-left and upper-right corners of the models on both scales we can make a theoretical estimation of the reduction of the computational cost due the application of the scale separation. Assuming that the time needed for the computation of one iteration of each of the models is the same (the extra effort due to the particles in the resolved suspension model and that due to the advection-diffusion solver on the macro scale is indeed comparable)
Figure 4.5: Scale separation map for the suspension system showing the spatio-temporal scale domains of the micro and macro models and thereby illustrating the scale separation between them. On the macro-scale both models, MaS and MaF, respectively, are defined on the same scales. The shaded extension of MiS towards smaller spatial scales results from the fact that suspended particles in MiS are described in a Lagrangian framework. The extension towards smaller temporal scales is due to the application of a scale splitting of the fluid and particle dynamics in MiS allowing the particle dynamics to be resolved on much finer temporal scales (see Sec. 4.4.1). The extension towards larger temporal scales (indicated by dashed gray lines) illustrates the temporal extension of the simulation to obtain a better statistics over the measurements of viscosities and diffusivities.

We can estimate the computational effort of the HMM simulation as

\[ C_{\text{HMM}} = c_m \left( \frac{L_m}{\Delta x_m} \right)^{D_m} \frac{T_m}{\Delta t_m} + c_M \left( \frac{L_M}{\Delta x_M} \right)^{D_M} \frac{T_M}{\Delta t_M}. \]  

Here the dimensions of the models are \( D_m = 2 \) and \( D_M = 1 \). The effort of simulating the whole problem using the fully resolved suspension model, thus at \( \Delta x_m, \Delta t_m \), would be

\[ C_{\text{full}} = c_m \frac{L_M}{\Delta x_M} \frac{L_m}{\Delta x_m} \frac{T_M}{\Delta t_M}. \]  

Here we assume the otherwise quasi-1D system has a second dimension of \( L_m \) necessary to resolve the typical length scale of the micro-dynamics.

To obtain a measure of the gain through scale splitting with the HMM approach we calculate the ratio

\[ \frac{C_{\text{full}}}{C_{\text{HMM}}} \approx 1.1 \cdot 10^7. \]  

This means, using the HMM approach we can reduce the computational effort by seven orders of magnitude. If we would use \( L_{M,x} = \Delta x_M \) here, the speedup is \( 1.9 \cdot 10^8 \) and in the case of a full 2-dimensional macroscopical flow, i.e. \( D_M = 2 \) and \( L_{M,x} = L_{M,y} \), using the same scales we can estimate the speedup to

\[ \frac{C_{\text{full}}}{C_{\text{HMM}}} = \frac{c_m \left( \frac{L_M}{\Delta x_M} \right)^{D_M} \frac{T_m}{\Delta t_m} + c_M \left( \frac{L_M}{\Delta x_M} \right)^{D_M} \frac{T_M}{\Delta t_M}}{\left( \frac{L_m}{\Delta x_m} \right)^{D_m} \frac{T_m}{\Delta t_m} + \left( \frac{L_M}{\Delta x_M} \right)^{D_M} \frac{T_M}{\Delta t_M}} \approx 5.8 \cdot 10^9. \]
The reduction of the computational effort is not the only advantage of the HMM approach in this case. Applying the Lattice-Boltzmann method, due to the Mach number limit of lattice velocities a simulation of the whole system of size $L_M = 50000\Delta x_m$ at a resolution $\Delta x_m$ would limit the shear rate drastically to very small values. It would not be possible to study the effect of shear-thickening on the concentration and velocity profiles. Comparable to the situation with parallelization under usage of the Lees-Edwards boundary conditions in Sec. 4.4.2, splitting the whole domain into domains that can be Galilei-transformed and therefore can reach arbitrary absolute velocities (in the reference frame of the macroscopic domain). The problem of limited validity range is inherent to practically every method, not only numerical, so splitting and re-coupling scales of the considered problem can also be considered a problem solving strategy.

4.2.5 More on Coupling

In Fig. 4.6 the sub-execution loops for the single-scale models are shown together with their coupling. Although the models for the particle dynamics on the micro-scale, respectively the dynamics of their continuous representation on the macro-scale, are not of CA type we still can express their computational structure in the sub-execution loops conventionalized in the CxA formalism.

The MaF model being based on the LB method clearly fits into the CA description. In Fig. 4.6 we find a collision step as well as a propagation step that directly correspond to those of the LBM. In the collision step, however, information on the local relaxation parameter $\tau(x_M,t_M)$ is missing. Inside the iteration
loop, and, since \(\tau\) is also a function of \(x_M\), inside the loop over all nodes of the domain, observables \(u(x_M, t_M)\) and \(\dot{\gamma}(x_M, t_M)\) are computed from the state variable vector \(f_i(x_M, t_M - \Delta t_M)\). Those are passed to \(\text{MaS}\) and \(\text{MiS}\), respectively, and the execution is halted until \(\tau = f(\nu_{\text{app}})\) was made available by \(\text{MiS}\).

The execution of the coupled system as depicted in Fig. 4.6 is different to a CxA as introduced in chapter 2 due to not using MUSCLE to couple the submodels\(^1\). Realizing the coupling with MUSCLE would mean all submodels are brought to life by a centralized initialization routine called \textit{plumber} who also spawns the conduits who would contain the mapping functionality. The plumber would wire the submodels with the conduits and terminate leaving a fully decentralized system behind. Instead, the coupling to the micro-model was realized the “classical” way via a function call, which also has the effect that a send primitive is blocking instead of a non-blocking send provided by MUSCLE. This later fact has no consequence on the execution because the graphs in Fig. 4.6 and Fig. 4.7 are cyclic. The fact that the coupling via the collision operators of \(\text{MaF}\) and \(\text{MaS}\) was realized by implementing \(\text{MiS}\) as a subroutine of the combined \(\text{MaF}+\text{MaS}\) kernel, however, results in a master-slave coupling where the combined \(\text{MaF}+\text{MaS}\), once launched by the user, will trigger the execution and termination of \(\text{MiS}\).

The macro-model \(\text{MaS}\) for the advection and diffusion of the field of local volume fraction \(\phi(x_M, t_M)\) is based on a Lagrangian model for a finite number of massless point particles that are advected with a velocity interpolated from \(u(x_M, t_M)\) (passed by \(\text{MaF}\)) and a random step according to the interpolated value from the diffusivity field \(D(x_M, t_M)\) (passed by \(\text{MiS}\)) in real coordinate space. It could therefore be viewed as an agent based model with the tracer particles as agents whose dynamics are functions of external fields comparable to the situation in the CxA model for in-stent restenosis discussed in chapter 3 where the dynamics of the Lagrangian agents representing smooth muscle cells is influenced by the external fields of drug concentration and wall shear stress which were both provided by Eulerian models. In the ISR model mappers had the function to translate between lattice-based data and agent-based data. In the present situation of the coupling of \(\text{MaS}\) to the other models data for velocity and diffusivity is based on the same macroscopic lattice \(x_M\). It has to be mapped to each of the Lagrangian tracer particles. Vice versa, particle based data (their positions) have to be mapped back to the volume fraction lattice field \(\phi(x_M)\). We could therefore think of the Lagrangian particle advection-diffusion model (LPAD) be separated from the data mapping and include the mapping functionality into “smart conduits” for each of the couplings \(u\) from \(\text{MaF}\), \(D\) from \(\text{MiS}\), and \(\phi\) to \(\text{MiS}\). Same is true for the models of the fluid and solid phase in \(\text{MiS}\), respectively. When treating them as separate submodels, the functionality of mapping the positions of the suspended particles to the lattice of the fluid solver, the mapping of exchanged momentum per boundary node to forces and torque on single particles, the mapping from lattice-based shear stress in the fluid and internal stress per particle to a single value of \(\nu_{\text{app}}\), as well as creating the initial velocity field from \(\dot{\gamma}(x_M)\) and the initial particle distribution from \(\phi(x_M)\) could be defined as functionalities of appropriate mappers.

\(^1\)MUSCLE does not allow to spawn several instances of the same submodel agent, a functionality which necessary to run several micro-model realizations in a farming approach to parallelization. Later, we found that with the use of the database described in Sec. 4.6.2 a farming approach would not be of much advantage. However, at that time the implementation of the coupling was already too advanced to rebuild it using MUSCLE.
Figure 4.7: The complete coupling scheme shows also mappings of data structures internal to the submodels **MaF**, **MaS** and **MiS**. There are multiple ways to define submodels and their coupling based on which data mapping functionality is covered by them. In one extreme case, **MaF**, **MaS** and **MiS** can be viewed as agents that communicate via data structures defined on the lattice of the macro domain, i.e. $\nu_{\text{app}}(x_M)$, $D(x_M)$, $u(x_M)$ and $\phi(x_M)$. All internal mapping is hidden in this definition. The other extreme case is to consider the submodels of **MaS** and **MiS** as submodels. Then the mapping between lattice-based data and off-lattice data (Lagrangian particle models as “solid” in **MiS** and **MaS**) would be seen as functionality of mapper agents similar to the coupling schemes for the ISR CxA's presented in chapter Sec. 3. Throughout this chapter we will use the first definition.

The complete coupling scheme in Fig. 4.7 shows both ways to define submodels and their coupling including the data conversions which either could be seen as part of the submodels or as external “mappers”.

There is a third way to understand the coupling of **MaF**, **MaS**, and **MiS**, though not clear from Fig. 4.7. The submodel **MiS** is coupled to the collision operators of both macro-scale models, **MaF** and **MaS**, completing the information on the constitutive relations there. In principle, viscosity and diffusivity measurements could be split in two micro-models at the same (micro-)scales and sharing the same domain. This would result in a coupling scheme where the dynamics of fluid phase and that of the solid phase would have their own cycles between micro- and macro-model constituting a kind of dual micro-macro coupling.

As we did not make use of any multi-scale simulation framework, such as MUSCLE, we are free to chose one of both views (ignoring the third as this is not applicable), or a compromise of them, to base the construction and implementation of the multi-scale model and its coupling on. Throughout the remainder of this chapter we keep viewing **MaS** as a black box, i.e. “some” solver for the advection-diffusion of the lattice field $\phi(x_M, t_M)$, as the choice for a solver might change in the future. This is also true for **MiS** where the internal implementation of the solver for the flow of suspended particles could be changed in favor of a Stokesian dynamics solver, for example.
4.3 Macroscopic Non-Newtonian Flow (MaF)

To solve the flow of the suspension on the macro-scale the Lattice-Boltzmann method is applied. The first section will briefly introduce LBM in a general manner in order to base also the extension to suspended particles on it that is addressed in Sec. 4.4 which introduces the micro-model. The last paragraph will discuss the coupling to the other models.

4.3.1 Lattice-Boltzmann Method

The Lattice-Boltzmann method \[41, 126\] method is an alternative approach to hydrodynamics offering a very efficient way to solve the discretized Boltzmann equation on regular lattices. It has in common with its predecessor, the Lattice-Gas Automaton (LGA) \[127\], that particles kinetics are abstracted to a collision and streaming step but instead of single particles particle distributions collide at and propagate between the nodes of the lattice. This step from a microscopic to a mesoscopic representation comes with the advantage of eliminating the noise using all possible particle states. (See \[128\] for a derivation of LBM from LGA).

Despite being based on a description for dilute gases viscous flow behavior emerges automatically from this scheme (in the limit of small mean free path of the particles) so that with a properly chosen equilibrium function the Navier-Stokes equations are reproduced to second order of approximation, even for large viscosities. Due to its local nature implementation of complex boundaries and parallelization is comparatively straightforward.

When Richard Feynman was asked to explain why people spend a lot of time and money on building large parallel computers for the computation of particles that hop from one lattice node to the other (LGA) he put it this way\[2\]

\[
\text{We have noticed in nature that the behavior of a fluid depends very little on the nature of the individual particles in that fluid. For example, the flow of sand is very similar to the flow of water or the flow of a pile of ball bearings. We have therefore taken advantage of this fact to invent a type of imaginary particle that is especially simple for us to simulate. This particle is a perfect ball bearing that can move at a single speed in one of six directions. The flow of these particles on a large enough scale is very similar to the flow of natural fluids.}
\]

The same argument is true for Lattice-Boltzmann methods as direct successor of LGA. With this Feynman illustrates fundamental universality we find in physics which gives some freedom in constructing algorithms which produce similar behavior on a higher scale which we want to study.

There are several choices for the collision operator. The most simple is derived from the insight that the collision operator in the Boltzmann equation brings the particle velocity distribution closer to the equilibrium distribution \[129\]. This leads to the single-relaxation time collision scheme, named LBGK after Bhatnagar, Gross and Krook. More sophisticated collision operators exist, most notably the multi-relaxation-time collision (MRT) \[130, 131\]. The MRT lattice Boltzmann equation overcomes some obvious defects of the LBGK model, such as fixed Prandtl

\[http://www.longnow.org/essays/richard-feynman-connection-machine/\]
number \((\text{Pr} = 1\) for the BGK model) and fixed ratio between the kinematic and bulk viscosities. It can be shown that by applying the MRT scheme the numerical stability of LBM simulations can be significantly increased in comparison with LBGK \cite{131, 132}. However, choosing the numerous relaxation parameters, many of which correspond to the damping of modes that have no physical meaning, is not straightforward; a reason why they sometimes are called “magic parameters”. Another notable approach to increase the stability of LBM calculations is the entropic LBM collision operator \cite{133} which is characterized by an equilibrium function of exponential form which satisfies the first four moments of the Maxwellian and a relaxation parameter which is dynamically adjusted to satisfy a lattice version of the H-theorem. The later fact guarantees stability, but provokes criticism based on the fact that with the relaxation parameter the viscosity changes locally. Simulations of time-dependent problems therefore should be considered carefully. It also should be noted that efforts were undertaken to improve the stability of the LBGK scheme which allow simulations of unsteady flow at above \(\text{Re} = 10^4\) \cite{134}.

In this work a two-relaxation-times (TRT) relaxation scheme \cite{135} was used, which offers a good balance between quality of the results, computational cost and implementation effort. It achieves a slight improvement in comparison to the lattice-BGK scheme in terms of damping unphysical high-frequency modes, leading to more stable simulations at higher \(\text{Re}_p\).

At each time iteration \(t\) and at every node \(r\) of the lattice, \(f_i = f_i(r, t)\) are the particle densities traveling in directions \(e_i\), where \(\{i = 1, \ldots, b\}\) denotes the discrete velocity space. During the collision step, these distributions are relaxed towards an equilibrium distribution \(f_i^{\text{eq}}\). Unlike LBGK the TRT model uses two different relaxation times and a collision operator

\[
Cf_i = \frac{1}{\tau}(f_i^{\text{eq}}(f) - f_i)^+ + \frac{1}{\tau_-}(f_i^{\text{eq}}(f) - f_i)^- + g_i,
\]

(4.19)

where \(F_i^+\) and \(F_i^-\) denote the even and the odd part, respectively, of a function on the discrete velocity space:

\[
F_i^+ = \frac{F_i + F_i'}{2}, \quad F_i^- = \frac{F_i - F_i'}{2}
\]

(4.20)

where \(i'\) is such that \(e_i = -e_{i'}\). Then, defining a propagation operator \(P\), the propagation step reads

\[
Pf_i(r, t) = f_i(r + e_i, t + 1).
\]

(4.21)

In this notation the LB equation becomes

\[
Pf_i(r, t) = Cf_i(r, t).
\]

(4.22)

The relaxation parameter \(\tau\) is related to the kinematic viscosity by \(\nu = c_s^2(\tau - 1/2)(\Delta x^2/\Delta t)\) \cite{129, 135} where \(c_s\) is the speed of sound. The second relaxation parameter \(\tau_-\) is set to 1, a choice in favor of stability. Simulating flows of dense suspensions at relative high Reynolds numbers, setting \(\tau_- = 1\) effectively damps checkerboard effects and numerical pressure waves, therefore improving the robustness. With the choice for \(\tau_-\), also the position of the numerical boundaries \cite{136} can be controlled. The equilibrium function is a function of the local velocity distribution
through the density $\rho(f)$, and the velocity $u(f)$, in accordance to the kinetic theory of gases computed as the 0th and 1st moments of $f$

$$\rho(r, t) = \sum_i f_i(r, t) \quad (4.23)$$

$$\rho(r, t)u(r, t) = \sum_i e_i f_i(r, t). \quad (4.24)$$

For simplicity of notation, in what follows, we will identify $f_{eq}^i(r) = f_{eq}^i(\rho(f), u(f))$. Besides density and velocity, the momentum tensor can be computed as

$$\Pi_{\alpha\beta}(r, t) = \sum_i e_{i\alpha} e_{i\beta} f_i(r, t). \quad (4.25)$$

Conservation of mass, momentum and kinetic energy are constraints to any equilibrium function $f_{eq}^i(r, t)$ (and the Navier-Stokes equations for weakly compressible fluids are obtained [137]). Isotropy requires a multi-speed model which involves at least nine lattice velocities $e_i$ in 2D [129]. A suitable choice for $f_{eq}^i$ for this model is

$$f_{eq}^i = w_i \rho \left(1 + \frac{1}{c_s^2} e_i \cdot u + \frac{1}{2c_s^4} (e_i \cdot u)^2 - \frac{1}{2c_s^2} u^2 \right) \quad (4.26)$$

with the lattice velocities

$$e_1 = (1, 0), \quad e_2 = (0, 1), \quad e_3 = (-1, 0), \quad e_4 = (0, -1),$$
$$e_5 = (1, 1), \quad e_6 = (-1, 1), \quad e_7 = (-1, -1), \quad e_8 = (1, -1)$$

and $e_0 = (0, 0)$ for the rest particle distribution. The parameters $w_i$ and $c_s$ depend on the particular realization of LBM scheme. For the D2Q9 model with $f_{eq}^i$ given in (4.26) the speed of sound is defined as $c_s = 1/\sqrt{3}$ and the direction dependent weights $w_i$ read

$$\begin{array}{c|c|c|c}
  w_i & i = 0 & i = 1..4 & i = 5..8 \\
  \hline
  \frac{1}{3} & 1/18 & 1/36 \\
\end{array}$$

so that (4.26) agrees with an expansion of the Maxwell-Boltzmann distribution in small Mach numbers $u/c_s$ to the order of $u^2$.

The term $g_i$ in (4.19) is an extra momentum added to $f_i$ in every time step, therefore acting as a volume force $G$ that can be used to drive the flow according to the application of the simulation.

$$g_i = \Delta x^2 w_i c_s^2 e_i G \quad (4.27)$$

with the properties

$$\sum_i g_i = 0, \quad \sum_i e_i g_i = \Delta x^2 G. \quad (4.28)$$

It can be related to a pressure driven flow by

$$G = \frac{\Delta p}{L \rho} \quad (4.29)$$
where $\Delta p$ is the difference between the pressure at the inlet and the outlet of a pipe of length $L$, for instance.

The LBM scheme can be written as

$$P f_i(r,t) = C f_i(r,t)$$

(4.30)

using propagation and collision operators defined in (4.21) and (4.19).

It can be shown that in the asymptotic limit, the stress tensor,

$$S_{\alpha\beta}(r,t) = \left( \Pi_{\alpha\beta}(r,t) - C(r,t) \right) \frac{\Delta x}{\Delta t}$$

(4.31)

and pressure, $p = (\rho - \rho_0) c_s^2 \left( \frac{\Delta x}{\Delta t} \right)^2$, can be obtained locally. Here, $\rho_0$ denotes a constant reference density which is typically set to 1 in simulations. In practice, $\Pi_{\alpha\beta}$ has to be calculated as mean over pre-collision and post-collision densities $f_i$ and $C f_i$. $C(r,t) = \rho(r,t) u(r,t) u(r,t)$ is the local convection tensor.

We remark that in problem relevant units time step $\Delta t$ and space step $\Delta x$ satisfy

$$\frac{\Delta t}{\Delta x^2} = \text{const.}$$

(4.32)

a relationship called diffusive scaling in the literature being a prerequisite to recover the incompressible Navier-Stokes equations in the limit $\Delta x \to 0$ [138].

### 4.3.2 Coupling

As a CA-type of method there is a propagation of properties at one lattice node to the adjacent nodes and a collision at each node. It therefore can be directly translated into the sub-model execution loops as has been done in Fig. 4.7. In case of LBM, observables are obtained from the local state variables $f_i$, the phase-space one-particle distribution function, on which the propagation and collision rules of the CA are defined.

The local shear rate $\dot{\gamma}(x_M,t_M)$ can be determined from the local shear stress tensor and $u(x_M,t_M)$. However, a more direct approach was applied reading

$$\dot{\gamma}(u) = \sqrt{\frac{1}{2} (\partial_{\beta} u_{\alpha} + \partial_{\alpha} u_{\beta}) (\partial_{\alpha} u_{\beta} + \partial_{\beta} u_{\alpha})}.$$  

(4.33)

When passing the shear rate to MiS it is translated to the micro-scale by

$$\dot{\gamma}_m = s_{\dot{\gamma}}^{-1} \dot{\gamma}(x_M, t_M).$$

(4.34)

After MiS finished the simulation run for $\dot{\gamma}_m$ and $\phi(x_M,t_M)$ from MaS the apparent viscosity $\nu(x_M,t_M)$ is received from MiS. The relaxation parameter is then obtained through the relation $\tau(x_M,t_M) = (3\nu(x_M,t_M) + 1)/2$.

The local velocity $u(x_M,t_M)$ needed as input for MaS is obtained using (4.24) and does not have to be scaled since MaF and MaS share the same spatio-temporal domain.
4.4 Fully Resolved Suspension Flow (MiS)

In order to introduce the micro-model MiS for the simulation of fully resolved suspensions this section will first describe how Lagrangian particles can be coupled to a Lattice-Boltzmann fluid and how the time-scales of the two subsystems can be decoupled to increase the numerical stability of the whole scheme. Thereafter Lees-Edwards boundary conditions will be introduced for LBM suspensions which allow to simulate shear flow in a quasi-infinite system, thus not resembling Couette-type of viscometers but a simulation box that presents a realized small part of a (virtual) bigger system and which is in this sense very similar to the HMM idea of a micro-model. In the subsequent subsection a correction to the momentum exchange algorithm as a part of the coupling of the LBM fluid to the suspended particles is proposed to recover Galilean invariance of the particle dynamics. When validating simulations are presented the method to measure the apparent viscosity is addressed. At the end the preparation of the initial condition is discussed. How shear-induced diffusivity can be measured in MiS will be described later in a subsection of section Sec. 4.5.2.

4.4.1 Direct Simulation of Suspended Particles

When modeling particles living in continuous space but suspended in a lattice-Boltzmann fluid, interaction between fluid and solid has to be realized via the intermediate representation of the particle on the lattice. Except for techniques similar to immersed boundary conditions [139], this is done by appropriate boundary conditions applied in the propagation step. In two dimensions, a spherical, that is disk-like, particle is represented by solidification of all nodes \( \mathbf{x} \) for which \( |\mathbf{x} - \mathbf{x}_p| < R_p \) where \( \mathbf{x}_p \) is the location of the particle’s center and \( R_p \) its radius. Mapping a particle to the lattice therefore is like cutting cookies out of a flat dough. It leaves broken links at which we have to apply appropriate boundary conditions. In some of the methods the fluid on the nodes inside the particle shell is kept alive as either physical or virtual fluid. The (dis-)advantages of this approach will be addressed in the next sections.

To implement no-slip conditions at the surface of the particles, in this work the bounce-back at the links method (bbl) was applied, assuming the boundary to be always located at the midpoint of boundary links, i.e. of the lattice links that are cut by the solid-fluid interface. More sophisticated and accurate methods [140, 141, 142] (see also [143] for a review) exist and, in principle, can be combined with the ideas of the following sections. However, for clearness we stick here to the bbl description where in the standard formulation a distribution \( f_i(\mathbf{x}) \) is reflected if the associated link \( e_i \) intersects a particle surface. In case of bbl at moving solid surfaces, an additional term accounts for the change due to the velocity of the surface at the intersection point, and the propagation step is now given by

\[
\mathcal{P} f_i(\mathbf{r}, t_+)= \begin{cases} 
\mathcal{R}_{u_b} f_{i'}(\mathbf{r}, t_+) & \text{if } i' \text{ is BL}, \\
 f_i(\mathbf{r} + e_{i'}, t_+) & \text{else},
\end{cases}
\]  

(4.35)

using a reflection operator

\[
\mathcal{R}_{u_b} f_i = f_i + 2w_i c_s^{-2} u_b \cdot e_i
\]

(4.36)
where \( u_b \) is the velocity of the midpoint of the boundary link and \( i' \) denotes the opposite direction to \( i \). If the fluid node is not virtual, an amount of momentum

\[
\delta p_i = 2e_i'[Cf_i(r, t + 1) - \rho w_i(c_s^{-2}u_b \cdot e_i + T_{CJ})]
\]  

(4.37)

is transferred to the particle. \( T_{CJ} \) is a correction term. With \( T_{CJ} = 0 \) the Momentum Exchange Algorithm (MEA) as first published by Ladd [144, 125] is reproduced. As we will see later in this section, MEA causes strong deviations from Galilean invariance in some situations and an accordingly set \( T_{CJ} \) can successfully correct for these effects.

Regarding hydrodynamical properties of suspended particles, it has been shown that the use of the bounce-back rule for boundaries not orientated along the lattice results in an effective hydrodynamic boundary slightly displaced from the physical boundary. In the case of spherical particles with a radius \( a \) this leads to a hydrodynamic radius \( a_{hd} = a + \Delta \), where the deviation \( \Delta \) depends on the fluid viscosity [144, 145]. An \emph{a priori} correction of \( a_{hd} \) can achieve more accurate dynamics of the solid particles.

**Fluid-particles forces**

We begin focusing on two approaches to deal with the interaction between fluid flow and solid suspensions, the method proposed by Ladd [144, 125], and the ALD method [146, 147] (Fig. 4.8). Both techniques employ the Momentum Exchange Algorithm [144] to deal with hydrodynamical forces exerted on the particles. According to its original formulation, the MEA is used to approximate the momentum given to the particle by the surrounding fluid.

The transferred momentum \( \delta p_i \) gives rise to a force and a torque acting at the boundary point \( r_b = r + \frac{1}{2}e_i' \) according to

\[
F(r_b, t + 1/2) = \delta p_i / \Delta t
\]

\[
T(r_b, t + 1/2) = [r_b - R] \times F(r_b, t + 1/2),
\]

(4.38)

(4.39)

where \( R \) is the center of mass of the particle. The total force and torque on a particle are then obtained by a summation over the contribution from all boundary links \( i_b \).

In its original version [144] also the nodes inside the solid particles are treated as fluid, which contribute to the hydrodynamical forces as well. In other words, the summations (4.38) contain both fluid-solid and solid-fluid links. A particle is therefore composed of a rigid shell and the inner fluid in this description.

However, for solid phase densities \( \rho_s \leq \rho_f \) the explicit update of particle positions has been found to become unstable [125] and improvements on the update have been proposed [148, 145]. The applicability was later extended [145] to smaller ratios \( \rho_s / \rho_f \) by omitting the action of the inner fluid on the shell.

In the ALD method, the inner fluid is only virtual, but is still used to bypass an explicit treatment of the nodes which become fluid when the particle moves. In this description, the fluid momentum of a new fluid node has to be taken from the particles momentum, while, when the particle ‘absorbs’ a fluid site its momentum is given to the particle. Formally, the momentum exchange

\[
p_{f \rightarrow s} = \rho(n, j)[u(n, j) - U_p]
\]

(4.40)
leads then to another contribution $F_{f\rightarrow s}$ to force and torque on the particle.

When particles come closer than the lattice spacing $\Delta x$ the fluid between the particle can not be resolved if no grid refinement is applied in ever increasing depth. To overcome this lubrication breakdown problem, an explicit model for the lubrication forces based on (asymptotic) lubrication theory [145] has to be included. In most suspension models a single particle-particle force term is used. With the ALD method link-wise lubrication correction [147] were introduced which provide a better resolution of lubrication forces between arbitrary shaped particles or walls. In the present work we applied the lubrication correction as proposed in [149, 114]. It was derived from a first order approximation of the stress tensor based on the stream function for the flow in the gap between two particles that approach each other on their centerline. As a function of the gap $d_{12} = |r_2 - r_1|$ and speed of approach $u_{12} = -\partial_t (r_2 - r_1)$ it reads

$$F_{\text{lub}}(d_{12}, u_{12}) = \left( \frac{d_{12}}{R} \right)^{-\frac{3}{2}} \nu_l \rho u_{12} \left( F_0 + \frac{d_{12}}{R} F_1 \right)$$ \hspace{1cm} (4.41)$$

where $F_0$ and $F_1$ are numerical constant given in the references. The lubrication force is only applied for gaps $d_{12} < h_0 = 0.8\Delta x$ as for gaps larger $h_0$ the lubrication force is resolved by the LBM fluid. The actually applied lubrication force has to be reduced by the lubrication force at $h_0$, i.e.

$$F_{\text{lub,eff}}(d_{12}, u_{12}) = \begin{cases} F_{\text{lub}}(d_{12}, u_{12}) - F_{\text{lub}}(h_0, u_{12}) & \text{if } d_{12} \leq h_0 \\ 0 & \text{else} \end{cases}$$ \hspace{1cm} (4.42)$$

Unlike in [149, 114] where only the squeeze mode of the lubrication interac-
tion (4.41) was introduced, we also applied the according rotation mode which also leads to a contribution to the torque on the particles.

Combining hydrodynamical forces $F_{hd}$, possible $F_{r\to s}$, lubrication forces $F_{lub}$ and according torques, positions and velocities of the particles are updated by an integration of the equations of motion for rigid bodies. Usually this is done based on a time stepping $\Delta t$ equal to that of the LBM, and higher order schemes as 4th order Runge-Kutta integration methods can be applied.

**Time-Splitting of the Dynamics of Fluid and Particles**  
Ever increasing the shear rate the numerical simulation of the fluid and the particles suspended therein eventually leads to numerical instabilities even before reaching the rule of thumb limit for the lattice velocity $Ma(u) < 0.8$. This is not only due to the sensitivity of the LBM iteration to high pressure and velocity gradients, also the interaction with the moving boundaries give raise to unphysical oscillations that build up and lead to a breakdown when the relaxation step leads to over-relaxation and therefore to stripe and checkerboard effects. The fast change of boundary velocities at higher shear rates and primarily the frequent covering and release/refill of fluid nodes are sources for these misfeatures. Also, when two particles approach each other, small pressure waves induced by covering and refill of fluid nodes interact with the particle dynamics and their slightly delayed response might lead to building up oscillations in this coupled system. Of course, the accuracy of the whole scheme suffers already before these effects are visible. All these phenomena are high-frequency effects, spatial as well as temporal, i.e. clear discretization effects as consequences of $\Delta x$ and $\Delta t$.

However, the most sensible subsystem to an integration with a finite time step we found solving the dynamics of two approaching particles for small distances and high relative velocities, even in case of a resting center of mass of those two particles, therefore under exclusion of effects caused by singular events like covering/refill of fluid nodes. It is thus purely related to the integration of the lubrication correction (4.41).

In a model system we neglect the coupling to the LBM fluid and only consider the following ODE

$$\ddot{r} = -\frac{a}{r} \dot{r} - b \quad (4.43)$$

which has the same functional form as (4.41). With $r$ we denote the distance between the two particles, $a$ is a constant defining the strength of the coupling between the acceleration $\ddot{r}$ and the ratio of velocity $\dot{r}$ and $r$. The constant $b$ is an acceleration pushing the particles towards each other mimicking the interactions with other particles and the fluid in a shear flow. It is clear that in the limit $r \to 0$ the acceleration diverges. With any $b < 0$, the velocity $\dot{r} < 0$ and the gap inevitably $r \to 0$. With it the gradient $\partial_r \ddot{r}$ (constant $\dot{r}$ assumed here) also diverges leading to the breakdown of any numerical integration method with finite time step $\Delta t$.

In Fig. 4.9(a) the numerical solutions for $r(t)$ are shown for several values for $b$, all for the initial condition $r(t) = 1, \dot{r} = 0, \ddot{r} = 0$. The coupling constant $a$ was adapted so that always $a/b = \text{const}$ leading to the same $r(t)$ behavior but at higher $\ddot{r}(t)$. We see that the larger $b$ the earlier the numerical integration breaks down characterized by high frequency oscillations around a constant $r_{\text{break}}(b)$. In Fig. 4.9(b) the trajectories $\dot{r}(r)$ are shown. The coupling constant $a$ was adapted so that al-
Figure 4.9: (a) The time course of the gap $r$ in the system (4.43) for different $a > 0$ and $b > 0$ with constant $a/b$. The larger $b$ the earlier the integration breaks down. Initial conditions were $r(t = 0) = 1$ and $\dot{r}(t = 0) = 0$. (b) The trajectories in the $(\dot{r}, r)$ space show that the breakdown correlates with the convergence with $\dot{r}(r) \to \text{const} \cdot r$.

ways $a/b = \text{const}$. Larger $b$ lead to an earlier convergence toward the $\dot{r} = \text{const} r$ behavior.

As expected, comparing integration schemes we found improved stability for a 4th order Runge-Kutta scheme over the Velocity-Verlet integration and the 1st order Euler method. However, even the Runge-Kutta scheme improved stability only by one order of magnitude in the minimum distance $r_{\text{break}}$ reached. We therefore decided to treat the particle dynamics on a timescale $\Delta t_p$ much finer than the time step $\Delta t$ of the LBM fluid the particles are coupled to. This approach has been described earlier [150] but an explicit dependence of the splitting factor $\Delta t_p/\Delta t_{\text{LBM}}$ on the shear rate or mass-density ratio has not been given. Incorporating the Krieger-Dougherty relation (4.1), the approximate scaling of $\nu \sim S^{1/2}$ we find in shear-thickening (see Sec. 5), and the viscosity-stress relation we define the particle-dynamics time step according to

$$\Delta t_p = c \cdot \frac{\nu_{\text{KD}}(\phi)}{\nu_t} S(\rho_p/\rho_f, \nu_t, \dot{\gamma})^{3/2} \Delta t_{\text{LBM}}.$$ (4.44)

If this formula gives values below $\Delta t_{\text{LBM}}$, $\Delta t_p = \Delta t_{\text{LBM}}$, of course. For the prefactor a value $c = 1.5$ was found to be a good choice, leading to a first increase at approximately $\text{Re}_p \approx 0.3$ and approximately $\Delta t_{\text{LBM}}/\Delta t_p \approx 1000$ for the highest Reynolds numbers for which data is presented in Fig. 4.14.

Ideally, one would like to decrease the time step $\Delta t_p$ only for those particles for which $\dot{r}/r$ is large. However, due to multi-particle interactions this is not straightforward and we keep this idea for future works.

The application of the time-splitting of fluid and particle dynamics together with the improvements described before in this section allowed us to simulate sheared suspensions of hard-spheres at Reynolds numbers up to 10 before instabilities occurred again. In Fig. 4.14 we plotted data of the apparent viscosity only
Table 4.1: Comparisons of the approaches of Ladd [144, 125], ALD [146, 147] and CMES combined with an accurate refill (section 4.4.3), concerning the main algorithmic issues for LB suspension flows. In the case of Ladd method, the algorithm from [144, 125] has been combined with the later proposed lubrication correction [145]. Some features are formally independent from each other, which leads to numerous combinations. For simplicity, we restricted this work to the original formulations.

up to $\text{Re}_p = 4$ because of finite size effect for larger $\text{Re}_p$ (particle clusters that percolate the system).

The additional computational costs due to the smaller time step $\Delta t_p$ are admittedly not negligible. However, for a constant strain over one simulation of 100 the total time steps needed increase with inverse shear rate. By chance costs for the computation of the fluid and that of the particles added up to an almost constant total runtime, regardless of the shear rate.

To summarize the methods, in Tab. 4.1 the main features of the approaches are compared. Note that some features are independent from others (e.g. refilling new fluid nodes and lubrication corrections) and this produces several combinations. For simplicity, we focused on setups reflecting the original ideas.

4.4.2 Lees-Edwards Boundary Conditions

When simulating sheared suspensions, Lees-Edwards boundary conditions allow more realistic computational setups as they remove the need of a domain bounded by shearing walls (like in Couette-type flow) which bias typical flow structures. Lees-Edwards boundary conditions therefore allow to investigate pure bulk properties in a quasi-infinite system. Besides that, they improve the computational efficiency of the simulations as the whole domain can be used to calculate averages.

We propose an implementation of Lees-Edwards boundary conditions for Lattice-Boltzmann simulations of particulate suspensions, combined with an accurate treatment of fluid-particle interaction. The algorithm is validated using a simple single-particle benchmark and further applied to a fully resolved suspension flow. Shearthickening behavior, that is prolonged to higher shear rates as compared to Couette flow results, could be observed.

Lees-Edwards boundary conditions (LEbc) [153] are routinely used in MD simulations to maintain a constant shear over a periodically continued simulation box. They allow to realize a sheared system without the need of explicit shearing solid

---

Periodic boundary conditions are applied in the directions perpendicular to a shear velocity gradient while in the direction of the velocity gradient it is assumed that a copy of the system is moving with a velocity $\mathbf{u}_{LE} = (u_{LE}, 0)$ with respect to the original system and a velocity $\mathbf{u}_{LE} = (-u_{LE}, 0)$ on the opposite side. This induces a shear rate $\dot{\gamma} = u_{LE}/L_y$ over the system height $L_y$.

The need to develop LEbc for suspension simulations arises from the fact that the application of the common planar Couette viscometer scheme, where two moving parallel planar walls shear the fluid in between, gives results that are unavoidably affected by the presence of the walls. Interested in pure bulk properties we can only use a limited system fraction for measurements. Wall effects are the formation of a depletion zone near the wall and the resultant wall slip, both clearly observable in real suspensions [154, 155, 111] as well as in numerical simulations [156, 149]. Suspended particles are limited in their mobility near the wall leading to different particle structure and lower solid-fluid density near the wall resulting in a lower apparent viscosity in that region. Also, the wall slip decreases the shear rate over the bulk leading to a different situation than intended.

The use of viscometer theme in LBM simulations has even greater impact. Here fluid velocity is limited to low Mach numbers and viscosity has a lower bound to ensure stability of the relaxation scheme. This limits the system size at higher $Re_p$. Additionally, the system size itself is limited in lattice and particle size units to keep the problem computational feasible. Therefore, with numerical simulations we cannot escape boundary effects. However, appropriate boundary conditions as proposed here can minimize such effects.

In Computational Fluid Dynamics (CFD) formulations of LEbc exist using other methods for particulate suspensions, for example a finite-element method together with a rigid-ring description of the particle [157]. For Lattice-Boltzmann methods (LBM) LEbc for suspensions in 3D [158] were reported but unfortunately
Implementation

It what follows we present a method for suspension simulations that allows the consistent treatment of solid particles crossing the boundary between systems which move respect to each other. It is an extension of the method by Wagner and Pagonabarraga for fluid-only Lattice-Boltzmann systems [161], where fluid densities that cross a LE boundary have to undergo two steps in addition to normal periodic propagation. The first problem is the implementation of a velocity shift in a method with a fixed set of velocities of densities instead of particles with real coordinates. A Galilean transform has to be applied to the distributions, \( f_{i,u_0} \rightarrow f_{i,u_0 + u_{LE}} \) where \( u_0 \) denotes the reference velocity of the computational domain, and \( u_{LE} = (u_{LE}, 0) \) denotes the velocity shift between the system copies. Following [161] a transform rule can be derived from an approximation of (4.19),

\[
f_i = f_i^{eq} + \tau(\partial_t f_i^{eq} + c e_i \nabla f_i^{eq}) + O(\partial^2). \tag{4.45}
\]

All terms of order \( O(\partial) \) can be skipped, because their zeroth and first moments are of negligible order \( O(\partial^3) \). The \( u \)-dependence is kept and one arrives at a rather simple expression for the change in the boundary-crossing density due to the transform which reads

\[
f_{i,u_0 + u_{LE}} - f_{i,u_0} \approx f_i^{eq}(\rho, u + u_{LE}) - f_i^{eq}(\rho, u), \tag{4.46}
\]

where \( \rho = \rho(f(r,t)) \) and \( u = u(f(r,t)) \). From that we define a Galilean transform operator

\[
\mathcal{G}_{u_{LE}} f_i = f_i + f_i^{eq}(\rho, u + u_{LE}) - f_i^{eq}(\rho, u) \tag{4.47}
\]

that has to be applied to all densities that cross a LEbc. The local density is conserved under this transform.

Another issue in defining a consistent LEbc for Lattice-Boltzmann systems is that the shift between the system copies \( s_{LE} \) does not necessarily correspond to an integer multiple of the lattice-spacing. In practice, densities crossing a LE boundary have to be mapped to the destination lattice by an interpolation. More precisely, they have to be distributed over the two cells that partly overlap the
Figure 4.11: Illustration of the cases to be distinguished during the propagation of $f_5$ densities in a model case. Shown are some of the nodes of the upper boundary layer and a part of the lower layer at a Lees-Edwards boundary. Yellow (light gray) nodes belong to a particle crossing the boundary, blue (mid gray) are fluid nodes. $\mathcal{G}$ denotes the application of the Galilean transform (4.47), $\mathcal{R}$ the application of the reflection operator (4.36).

virtual destination cell. For the simple case of only fluid nodes, for the propagation of densities that are fed from nodes in a shifted reference frame, we can write

$$ P_{\text{LE}} f_i(r) = s_1 \mathcal{G}_{\text{u,LE}} f_i(r_1 - e_i) + s_2 \mathcal{G}_{\text{u,LE}} f_i(r_2 - e_i) $$

with

$$ s_1 = \text{mod}(s_{\text{LE}}, 1) \quad s_2 = 1 - \text{mod}(s_{\text{LE}}, 1) \quad r_1 = (x + \text{int}(s_{\text{LE}}) + 1, y) \quad r_2 = (x + \text{int}(s_{\text{LE}}), y). $$

Including solid-fluid interaction given by (4.35) for the propagation of densities that may cross a LE boundary we can write

$$ P_{\text{LE}} f_i(r) = s_1 \left\{ \begin{array}{ll} \mathcal{R}_{u_b} f_i'(r) & \text{if } L_1 \text{ is BL} \\ \mathcal{G}_{\text{u,LE}} f_i(r_1 - e_i) & \text{else} \end{array} \right\} + s_2 \left\{ \begin{array}{ll} \mathcal{R}_{u_b} f_i'(r) & \text{if } L_2 \text{ is BL} \\ \mathcal{G}_{\text{u,LE}} f_i(r_2 - e_i) & \text{else} \end{array} \right\}. $$

This approach is motivated by sub-grid scale boundary conditions proposed by Verberg and Ladd [140]. There also, partially covered destination nodes caused a splitting of propagated densities.

To illustrate the application of the propagation operator $P_{\text{LE}}$ Fig. 4.11 depicts the fate of $f_5$ densities which propagation vectors point outside the original domain. Density $f_5(r_{(1)})$ is split into two parts $s_1 f_5(r_{(1)})$ and $s_2 f_5(r_{(1)})$ which are propagated to the two nodes of the shifted copy that partly overlap the ghost node it would go to by the application of $P$. On both parts the action of $\mathcal{G}_{u_b}$ will be applied. In the case of $f_5(r_{(2)})$ one part, $s_1 f_5(r_{(1)})$, is reflected by the action of $\mathcal{R}_{u_b}$ because the according destination node belongs to the particle. In this case, $u_b'$ is transformed to $u_b' = u_b + (u_{\text{LE}}, 0)$ because the particle is described by its center residing in the reference frame moving with $u_{\text{LE}}$ in respect to the original frame.
In the case of \( f_5(\mathbf{r}_3) \) both parts are reflected by \( R_{u'} \). For the case the suspension model includes the description of inner fluid, the density \( f_5(\mathbf{r}_{(4)}) \) undergoes a similar procedure as \( f_5(\mathbf{r}_{(1)}) \).

**Numerical results**

**One particle crossing LEbc** As an essential validation of the implementation of LE boundary conditions we let a single particle cross the LE boundary. No external forces are applied to the particle. Using LEbc where no solid objects are needed to drive the flow and therefore no preferred reference exists, we are able to superimpose the sheared flow with any constant velocity as long as actual lattice velocities are kept in a range where LBM is valid. Ideally, conserving Galileian invariance, the velocity of the particle should stay constant. However, serious deviations therefrom could be observed using the existing suspension methods. This led to a separate investigation and the proposal of a correction to MEA, which is addressed in Sec. 4.4.3 together with the results for the one particle crossing LEbc experiment.

**Suspension in sheared flow** Next, we simulate sheared suspensions with a solid-fluid fraction \( \phi = 0.40 \) using the LEbc method and the Couette scheme for comparison of the flow fields. In a squared system of size \( L_x \times L_y = 259 \times 259 \) lattice nodes we suspended 133 particles with a radius \( R_p = 8.0 \). The suspension was sheared at particle shear Reynolds numbers \( \text{Re}_p = 0.005, 0.2, 1.0 \) and profiles of velocity and the solid-fluid ratio were measured. Averages were obtained over a shear of \( s_{\text{LE}} = 10L_x \).

Employing the Couette condition a clear deviation from the Newtonian velocity profile can be seen for \( \text{Re}_p \gtrsim 0.1 \), as shown in Fig. 4.12. Close to the walls an increasing slip becomes clearly detectable which lowers the velocity gradient over the bulk. This behavior could already be seen in simulations of hard-sphere suspensions [162] as well as experimental setups [155].

Fig. 4.12 shows the plot of \( \phi(y) \), demonstrating that the profile deviates from the linear Newtonian profile in two aspects

(i) Especially close to the walls, particle densities oscillate with a period of slightly more than one particle width. This suggests that particles tend to align to the walls forming sheets [163]. Such structures of particles parallel to the shear velocity decrease the average inter-particle friction.

(ii) The average particle density near the wall is less than in the bulk. The presence of the wall alone causes a lift force on the particles [164] and causes a depletion zone, according to observations in experimental setups [154]. Another effect of sheet-like structures of particles is that the mobility of particles is inhibited. Particles of sheets sliding along each other mostly collide with small collision cross sections, pushing the particles back into the sheet.

Both effects lower the viscosity and therefore lead to the observed wall slip. Repeating the same experiments using LEbc’s we obtain improved behavior. Here fluid and particles experience homogeneity. Particles are free to form configurations typical for a certain shear rate all over the domain (on the right of Fig. 4.12). In fact, the plots of the profiles on the left of Fig. 4.12 show a flat distribution of
Figure 4.12: (a) Snapshot of a sheared suspension of 133 discs of radius $R = 8.0$ in a domain of $259 \times 259$ lattice sites. The particle-shear Reynolds number $Re_p \approx 1.5$. The gray scale code indicates low (dark) and high pressure (bright) areas. Note the typical particle and pressure field structures that would be inhibited by walls in a Couette scheme. (b) Comparison of profiles as obtained by application of the Couette scheme (dotted line) and Lees-Edwards boundary conditions (solid line). The local average particle density $\phi(y)$ shows strong wall effects in the Couette case. Using LEbc particles are homogeneous distributed over the system height. Both profiles were measured at $Re_p = 1.0$. The velocity profiles $v_x(y)$ as obtained using Couette boundary conditions show increasing slip effects near the wall for $Re_p = 0.1, 1.0$. Using LEbc at $Re_p = 1.0$ no systematic deviations from the Newtonian profile can be seen.
Figure 4.13: The increase of the relative apparent viscosity $\nu_{\text{app}}$ as a function of the solid-fluid ratio $\phi$ for $\text{Re} \ll 1$ as obtained by a simulation using CMES compared to the theoretical behavior proposed by Krieger-Dougherty and Einstein's approximation for small $\phi$. A comparison between results between Couette and LEbc results shows no significant difference in this regime.

Particles over the system height and no systematic deviations from the Newtonian velocity profile.

**Shear-thickening suspension** A highly relevant aspect of hard-sphere suspension rheology is shear thickening at higher particle shear Reynolds numbers $\text{Re}_p$. Already in the Stokes regime, the existence of solid particles causes a higher apparent viscosity. The increase of the relative apparent viscosity with higher solid-fluid fractions $\phi$ is best described by the semi-empirical Krieger-Dougherty relation [165]

$$\frac{\nu}{\nu_t} = \left(1 - \frac{\phi}{\phi_{\text{max}}}ight)^{-[\eta]\phi_{\text{max}}}$$

where $\phi_{\text{max}} = 0.82$ is the solid-fluid fraction in 2D at which particles are packed that dense that any particle flow is impeded and therefore the viscosity diverges. $[\eta]$ is an intrinsic viscosity of the suspension and equals 2 in 2D. It has been shown that ALD and Ladd94 methods can reproduce the Krieger-Dougherty relation in a convincing accuracy over the range of implementable fluid-solid ratios in the range of small Re [156, 149]. In Fig. 4.13 we present measurements of the increase of $\nu_{\text{app}}$ over a wide range of solid-fluid ratios $\phi$ as obtained by CMES. The same system settings as for the simulations in the previous sections were used. The results agree well with the expected Krieger-Dougherty behavior.

For shear thickening measurements we used a system with size $259 \times 259$ and 133 suspended disks with a radius of $R = 8$, resulting in a solid-fluid ratio of $\phi = 0.40$. The density ratio was set to $\rho_s/\rho_f = 10$. We used the later value to be able to give direct comparisons with the shear-thickening simulations presented in [149]. To obtain the apparent viscosity we followed the method presented in [156, 162] which allows shear stresses to be measured in the bulk instead of measuring frictional forces at the walls. Shear stresses can be calculated along a horizontal plane as the sum over the contributions from the fluid and solid phase as outlined in [162]. We
obtained stresses by averaging measurements over 10 equally distributed planes and a shear of at least $10L_x$. In the case of Couette boundary conditions we took care that only bulk properties were measured by distributing the stress planes in such a manner that none of them was closer than 50 lattice sites to the wall. Looking at the profiles in Fig. 4.12 we can safely assume that direct influences of the walls are absent at this distance. Using LEbc measurements can be carried out over the whole system height. This increases the bulk area we are interested in significantly.

In Fig. 4.14 apparent viscosities are shown, normalized by the Krieger-Dougherty relation obtained for a range of $Re_p$ from 0.001 to 4.0. The maximum achievable Reynolds number is limited by stability conditions of the LBM scheme giving a lower bound for the fluid viscosity and limiting the maximum particle lattice speed. For the highest $Re_p = 4$ we used a fluid viscosity $\nu_f = 0.02$ and set the Lees-Edwards shift velocity to $u_{LE} = 0.0405$.

For comparison in Fig. 4.14 we also plotted shear-thickening measurements obtained by Kromkamp et al. with the ALD method, Couette conditions and comparable system settings [149]. Our results for the Couette case agree very well with their data. At about $Re_p = 0.01$ the apparent viscosity begins to increase and levels off at about $Re_p = 0.2$. Also at the highest $Re_p$ we could carry out stable simulation $\nu_{app}$ stayed roughly constant. We repeated the experiments using the Lees-Edwards boundary conditions proposed in this work and obtained a behavior that significantly differs from the Couette results for $Re_p > 0.1$. Instead of a saturation behavior $\nu_{app}$ shows a continued increase. The profiles in 4.12 offer a straightforward explanation for this. Using the Couette conditions we see a slip near the wall for $Re_p > 0.1$ which lowers the velocity gradient and the effective shear rate over the bulk.

A direct relation between the actual bulk shear rate and $\nu_{app}$ independent from $Re_p$ could not be found to be clearly visible over the plateau range. This may be caused by the different characters of momentum exchange events for different
slip velocities. At higher slip velocities particles may collide with much higher momentum exchange and impact on the particle structure in the bulk but, on the other hand, such events are less frequent in that regime.

When LEbc’s are used these wall effects are not present and the suspension can be sheared at the actually intended shear rate enabling unbiased studies of sheared hard-spheres suspensions. The apparent viscosity obtained with LEbc shows a continued increase of $\nu_{\text{app}}$ giving rise to the assumption that shear-thickening either levels off at higher $Re_p$, or that there is a crossover to jamming behavior. There the shear rate is so high that the suspending role of the fluid becomes negligible and particles collide and jam as in a granular medium.

Using this implementation of LEbc, we could also start extended studies on cluster formation, investigating the growth of typical structures in sheared suspensions. Results will be presented in Sec. 5. Concerning the present work, we only took care that particle cluster size did not exceed system size to prevent percolation of particle clusters which would lead to a jammed suspension in a Lees-Edwards system.

For an estimation of the increase in computational effort due to the proposed implementation of LEbc, we measured the ratio between the runtime of both to be 1.06. The additional effort of 6% is still much less than the approximately 30% of the system height at the highest shear rate that cannot be used for measurements due to the bias by the wall. Increasing the distance between the walls to correct for this, as well as increasing the wall speed to maintain the same shear rate in a Couette flow simulation, would lead to a Mach number that is approximately 30% higher and would therefore lead to a potentially higher error of the LB method and to the onset of instabilities. Furthermore, LEbc do allow for simulations of systems where the typical correlation length (e.g. particle cluster size) would exceed the unbiased bulk width in a Couette flow simulation.

As a final remark, although only of technical importance, we observe that, using LEbc, the global momentum is not exactly conserved due to numerical inaccuracies of Lattice-Boltzmann schemes and the absence of solid obstacles which would define a preferred reference frame. We could improve the conservation by a correction of fluid and particle velocities equivalent to a Galilean transform of the whole system, without affecting the physics.

4.4.3 Corrected Momentum Exchange

We are interested in the rheology of sheared suspensions, i.e. the change in viscosity when the system is subject to an external shear flow. In such situations, it becomes useful to demand Galilean invariance of the computation of fluid-solid interactions to ensure homogeneous behavior of the particles, not dependent on the external velocity and therefore not dependent on the position in a sheared system (see Fig. 4.15).

In this section the effect of non Galilean invariance of the LBM methods on the dynamics of sheared particle-fluid systems are investigated. Selecting particular cases of one particle crossing a Lees-Edwards boundary, we show that standard

---

methods for the simulations of suspension flow might lack sufficient accuracy. Using asymptotic expansion techniques, we motivate the need for a correction term and implement a possible solution. The obtained Corrected Momentum Exchange for Suspension (CMES) preserves local consistency and Galilean invariance in relevant orders.

**Asymptotic Analysis**

Using the asymptotic expansion technique [138], it can be shown that the numerical solution of (4.22) can be approximated by the expansion

\[
F_h(n, j) = f^{(0)} + hf^{(1)}(nh^2, h\dot{j}) + h^2 f^{(2)}(nh^2, h\dot{j}) + O(h^3)
\]  

(4.53)

with smooth and \(h = \Delta x\)-independent coefficients defined as

\[
\begin{align*}
f_i^{(0)} &= w_i, \\
f_i^{(1)} &= w_i c_s^{-2} c_i \cdot u_{NS}, \\
f_i^{(2)} &= w_i c_s^{-2} p_{NS} + \frac{w_i c_s^{-4}}{2} \left( |c_i \cdot u_{NS}|^2 - c_s^2 u_{NS}^2 \right) \\
&\quad \tau w_i c_s^{-2} c_i \cdot \nabla u_{NS} \cdot c_i,
\end{align*}
\]

(4.54)

where \(u_{NS}\) and \(p_{NS}\) solve a Navier-Stokes problem.

From (4.54), we conclude that (4.23) yields a second order accurate velocity, while a first order accurate pressure can be obtained with

\[
p = c_s^2 \rho - \frac{1}{h^2}.
\]

(4.55)

Similarly a first order approximation of the viscous stress tensor \(S[u] = \nu(\nabla u + \nabla u^T)\) can be extracted using

\[
S_{\alpha\beta} = -\frac{\nu}{h^2 c_s^2 \tau} \sum_i w_i (f_i - f_i^{eq}(f)) c_{i\alpha} c_{i\beta},
\]

(4.56)
The results of the analysis actually provide more than these accuracy results. They contain relevant information concerning the structure of the solution, which can be used to improve the algorithm.

**Numerical tests**

We aim on a method which resolves the dynamics of particles independently from any constant velocity superposed to the movement of local centers of mass as present in all non-trivial flow problems. In case of errors due to the violation of Galilean invariance, such errors would be amplified if the same physical problem is described using coupled reference systems, like in the application of LEbc. We used the LEbc proposed in Sec. 4.4.2 (also published as [113]) to allow simulations of homogeneously sheared systems and to superpose them with a small, constant velocity.

For the simulations presented in the following we used a system size \( L_x = L_y = 128 \) and a particle radius \( R = 4.8 \) (all numbers given in lattice units). The density ratio was set to \( \rho_s/\rho_f = 10 \) and the kinematic viscosity of the fluid \( \nu = 1/6 \).

**Single Disk (D_1)**. We simulate a single particle in a periodic shear flow. In detail, initially a particle is placed in the midpoint of a the squared domain, with a velocity equal to the fluid velocity at that point \( \mathbf{v}_p(t=0) = \mathbf{u}_s(t=0) = (u_x, 0.005) \) (Fig. 4.16). Additionally the flow is initialized and maintained at a horizontal shear \( \dot{\gamma} = u_{LE}/L_y \) and the particles angular speed is accordingly initialized to \( \omega(t=0) = \dot{\gamma}/2 \). With that an equilibration of near-particle fluid field and \( \omega \) was reached after approximately \( t = 3000 \). In any case, this equilibration process is physical and should therefore be Galilean invariant, too.

Throughout the simulation we measure the vertical velocity of the particle \( \mathbf{v}_{p,y}(t) \), which should remain constant in the ideal case. However, as shown in Fig. 4.17, in the case of the ALD method a significant deviation from \( \mathbf{v}_{p,y}(t=0) = 0.005 \) can be observed already from early times on, not having reached the LEbc yet. When Ladd method is used, no deviations can be seen during that time, which finds its explanation in the fact that every error made in link-wise calculations is compensated by the use of physical inner fluid, where every momentum exchange at a boundary link has its inner inverted counterpart. Using the ALD method, e.g. applying MEA only at the outer surface, the particles feel a non-physical vertical force in situations where points at the upper surface move at a different speed than points at the lower surface.
Deviations in both methods appear when the particle crosses the LEbc. Notice that in this situation boundary links exist that connect nodes in different reference frames (in practice, different \( u_b \) are used in (4.37)). Here, errors cannot be compensated by the inner fluid. The non-Galilean effect grows linearly when varying \( u_x \). A shear rate \( \dot{\gamma} = 5 \cdot 10^{-5} \) was present in all these measurements.

Fig. 4.18 shows similar measurements. Here the curves are recorded for different \( \dot{\gamma} \) and the superposed initial velocity \( u_s = (0.05,0.005) \) was kept constant. All the curves show similar behavior compared to the measurements in Fig. 4.17, both in absolute size and order. MEA in its form (4.37) (with \( T_{kj} = 0 \)) leads to a non-zero force integration over the surface depending on both, the absolute particle speed as well as the difference between surface velocities at the upper half and the lower half of a particle.

**Approaching Disks (D_2).** The next example aims to demonstrate the effects of combining LB fluid-structure interaction with suspension dynamics, when different flow velocities are used as background. We consider two equal particles moving toward each other, driven by a small external force (Fig. 4.19 and 4.20). The gap at time \( t = 0 \) is \( r_{1,2} - R_1 - R_2 \approx 1.19 \), in order to have at least one fluid node between the particles at the beginning. Without adaptive grid-refinement lubrication forces have to be implemented explicitly. Regarding the different suspension methods, lubrication corrections were applied as given in Tab. 4.1. For simulations using CMES we applied a lubrication correction in the form proposed in [149] for 2-dimensional systems. Such forces were invoked for gaps smaller than \( h_c = 1.0 \approx 0.21R \).

Both particles are always aligned in x-direction. To mimic the fact that particle-particle collisions happen at different locations in a shear flow, the flat (non-sheared) flow field is superposed by a constant translational velocity \( u_s \), and the experiment is repeated using different absolute values and directions of \( u_s \). The particles are always initialized with the same velocity as the fluid. Measuring the gap between the particles as a function of time, we can investigate the combined effect of force computation, lubrication corrections and nodes re-initialization.

Fig. 4.19 shows the results obtained with superimposed velocities \( u_s = (u_x,0) \). Looking at the problem from the lattice reference frame, one particle is following the other while they get closer. In an ideal case, the lubrication-damped collision of the particles should not be affected by the absolute velocity of their center of mass. However, using Ladd’s method at higher superposed velocities the particles tend to approach themselves faster in comparison to a simulation where the center of mass is at rest (which is the usual way lubrication behavior investigations are carried out in the literature). Fluid between particles is of higher pressure than the fluid at the opposite sides of the particles. When particles move fluid nodes will turn into inner fluid. Not instantly in equilibrium with the rest of the inner fluid, such a high-pressure inside fluid causes forces that counteract lubrication forces. As an additional effect, adopting high-pressure gap nodes also leads to a kind of pumping: the following particle gains mass with the increase of inner-fluid pressure which biases the particle dynamics. The ‘pumping effect’ has another consequence: When inner fluid turns into outside fluid behind the particle the increased pressure also contributes to a less damped collision behavior of the particles.

Although the last argument could also apply in the case of simulations using ALD, results obtained with this method show a better behavior. Here, by definition
Figure 4.17: The vertical speed \( v_y \) of a single particle in a sheared system (\( \dot{\gamma} = 5 \cdot 10^{-5} \)) crossing a LEbc for different superposed velocities \( u_{s,x} \). At about \( t = 12300 \) the particle reaches the LEbc and crosses it within a time \( t_c \approx 2000 \). In the ALD case the deviation from the constant velocity present from early times causes a slight shift of the minimum towards later times (also seen in Fig. 4.18).
Figure 4.18: The vertical speed $v_y$ of a single particle in a sheared system crossing a LEbc for different shear rates $\dot{\gamma}$. The particle is moving with the fluid at a superposed background velocity $u_s = (0.05, 0.005)$. 
of the model, fluid nodes always exist between the particles. As a consequence we could observe that the 'pumping effect' was less pronounced.

When particles are very close and therefore lubrication forces very sensitive we would have expected that the additional momentum exchanged when nodes are covered or uncovered lead to deviant behavior. However this seems to have little importance for the particle dynamics in these tests. Although small deviations can be seen, the action of a physical inner fluid turns out to be the most important source of errors.

In Fig. 4.20 we show the results for the case of a constant absolute superposed velocity $|u_s| = 0.075$ but for different directions. The curves for $\alpha = 0, \pi/6, \pi/3, \pi/2$ show only slight deviations from each other if ALD is used. In the case of active inner fluid (as in Ladd's method), a strong angle-dependent behavior can be observed. For $\alpha = \pi/6, \pi/3$ another effect appears. If the background velocity has an inclination $0 < \alpha < \pi/2$ with respect to the vector $r_{1,2} = x_{p2} - x_{p1}$, we still observe a small 'pumping'. With the broken symmetry of the problem the different properties of the two particles cause the particles to start to tumble around each other. The results in Fig. 4.20 show the results for the case a constraint $y_{p1}(t) = y_{p2}(t)$ was applied and curves that cross the x-axis in case of tumbling.

**Galilean invariant force computations**

The previous results show that Ladd’s method and ALD have shortcomings. Using the asymptotic expansion technique it is possible to identify the main source of errors, and to derive leading order corrections to the original algorithms. Starting from (4.54), which approximates the solution of the LBM (4.22), a prediction for the momentum exchange contributions (4.37) (with $T_{CJ} = 0$) can be derived.

Let us consider a boundary node $k$, and an outgoing boundary link $i$ at $k$. We denote with $b_i(k)$ the intersection between the link $c_i$ and the fluid-solid interface. Inserting ((4.54)) into ((4.37)) we obtain (dropping the time dependence for brevity)

$$p_i(k) = p_i^{(0)}(x_k) + h^2 p_i^{(2)}(x_k) + O(h^3),$$

with

$$p_i^{(0)}(x_k) = 2w_i c_i$$

$$p_i^{(2)}(x_k) = 2w_i c_i^{-2} \left( p + \frac{c_s}{2} \left( |c_i \cdot u_b| - c_s^2 u_b^2 \right) - c_s^{-2} \nu c_i \cdot \nabla u_b \cdot c_i \right),$$

where the quantities on the right hand sides are evaluated at $b_i(k)$.

Using this approach, it can be proved [152] that MEA yields a first order accurate approximation of the force acting on a particle $P$,

$$F_P(t) = \int_{\partial P(t)} \left( -p(x) n + n \cdot S(x) \right) \, d\gamma.$$  

(4.59)

Beyond this general result, (4.58) provides additional useful information. We focus on the second order coefficient of (4.58), which contains a term not related to the boundary force, explicitly depending on the boundary velocity, and responsible for breaking the Galilean invariance.
Figure 4.19: Test D2 a: Gap size between two particles that feel horizontal forces pushing them together for different absolute values of the superposed velocity $u_s$. The direction is $u_s \parallel r_{1,2}$. The three plots show results obtained with different suspension models.
Figure 4.20: Test D₂ b: Gap size between two particles for different angles $\alpha = \angle(u_s, r_{1,2})$. Results for the three suspension models are shown. In the Ladd case, particles (if $y_{p1} = y_{p2}$ is not fixed) start to tumble leading to negative $x$-distances (dashed lines).
It must be remarked that in most cases this term produces a small global contribution [166]. However, this does not apply using Lees-Edwards BC, when the solid-fluid interaction of one particle are evaluated in different reference systems within the same time step (in practice, in this case the MEA depends on the contributions of two part of the interfaces, with different velocities).

Aware of (4.58) we can easily define a correction for the momentum exchange algorithm which reads

\[ p^{\text{CME}}_i(k) = p_i(k) - 2w_i c_i - h^2 w ic_i \left( |c_i \cdot u_b(b_i(k))|^2 - c_s^2 u_b(b_i(k))^2 \right) c_i. \]  

(4.60)

Together with \( p_i(k) \) in (4.37) and the summation over the particle surface in (4.38) this defines a corrected momentum exchange algorithm (CMES) for the calculation of hydrodynamical forces and the torque on the particle.

The violation of Galilean invariance by MEA, when used locally, has been already observed in [167] and cured by the introduction of additional virtual fluid nodes inside the solid domain. However, this is a special solution and care has to be taken at the implementation of such an idea when boundary links cross a Lees-Edwards boundary. The correction (4.60) offers a consistent analytical solution and, in most cases, allows an easier implementation.

Initialization of new fluid nodes

Reinitialization of new fluid nodes is a common task dealing with moving boundary LBM. Both Ladd’s and the ALD approach make use of inner fluid to deal with this. In general, this is justified if inner nodes hold densities which approximate the right characteristics when turning from an inner surface node to an outer surface node. This is likely only in situations where the particle acceleration is low. If a particle is accelerated the fluid directly behind the particle is typically of lower pressure while the adjacent node inside the shell would be of rather high pressure.

The problem can be solved by a more accurate initialization of LB densities according to the expansion coefficients (4.54). This can be done in an efficient way, separately approximating the equilibrium and the non-equilibrium part, as described in [166, 151]. In detail, denoting with \( k \) a new fluid node, first velocity \( u(k) \) and pressure \( p(k) \) are extrapolated using a set of available neighboring fluid node, to construct the equilibrium distribution

\[ \tilde{f}^{\text{eq}}_i(k) = E_i \left( \rho_0 + c_s^2 h^2 p(k), u(k) \right). \]

Then, a low order extrapolation for the non-equilibrium part \( \tilde{f}^{\text{neq}}_i(k) \) is added, initializing (omitting the time dependence)

\[ f_i(k) = \tilde{f}^{\text{eq}}_i(k) + \tilde{f}^{\text{neq}}_i(k). \]  

(4.61)

In practice, \( \tilde{f}^{\text{neq}}_i(k) \) can be copied from a neighboring node [151]. This approach is by definition locally consistent with the inner LB solution, and yields the same accuracy as the standard LBM [166, 151], providing a good balance between computational effort and quality of results. We remark that similar approaches, based on extrapolation techniques have been proposed in [141, 168], which also achieved accurate refill. However, these are based on more complicated extrapolations and
become less practical when dealing with the flow of dense suspensions. In practice, in case of colliding particles it might happen that not enough nodes are available in order to implement algorithm (4.61) with the required accuracy. In those cases, we have used lower order approximations for the equilibrium distribution, based on simpler averages (but still including a non-equilibrium approximation).

**Numerical Results**

Including the modifications described in the previous section, both tests D$_1$ and D$_2$ are solved more accurately.

Correction (4.60) for the momentum exchange algorithm significantly reduces non-Galilean deviations in D$_1$ (see figures 4.17 and 4.18) in comparison to the results obtained by standard MEA, applied only at the outer surface (ALD). Still, it can be observed that CMES does not behave as accurate as the original method by Ladd if the particle is treated in only one reference frame, but this is mainly due to additional symmetry properties of Ladd’s methods in the specific benchmark. In general, figures 4.17 and 4.18 show that the particle dynamics has significantly improved by the use of CMES when particle-surface interaction has to be treated partly in different reference frames.

The benchmark D$_2$ demonstrates that the use of an accurate equilibrium + non-equilibrium refill improve the resolution of particle dynamics, avoiding effects that are caused by just turning physical inner fluid nodes into outer fluid nodes and vice versa (figures 4.19-4.20).

In D$_2$ the ALD method also produces results almost independent of $u_s$. However, it still requires the computation of an inner fluid. Aiming at simulations of dense suspensions the cost for this is in the order of $\phi = V_s/V_f$. With the application of a refill procedure as proposed in this paper the computation of inner fluid can be omitted, providing a theoretical speed-up factor of $1/(1 - \phi)$.

**4.4.4 Initial Conditions and Observables**

When the micro-model is invoked in order to return values for the apparent viscosity $\nu_m = \nu_{app}$ and shear-induced diffusivity tensor $D_m$, the shear rate $\dot{\gamma}_M$ and volume fraction $\phi$ are passed as arguments. In Fig. 4.6 it is shown that the execution of an instance of MiS is preluded by the setup of the spatio-temporal domain. As discussed in Sec. 4.2.4 we consider a two-dimensional domain of $72\Delta x_m$ in both dimensions at a resolution of $\Delta x_m$ and the particle radius used was $R = 3.15\Delta x_m$.

Although this resolution of the spatial domain is smaller than that used to validate the Lees-Edwards boundaries introduced in Sec. 4.4.2, we found that this choice is a good compromise between accuracy and computational effort. Having set the solid-fluid mass density ration to $\rho_s/\rho_f = 10$, the measurements of the increase of $\nu_{app}$ with $\phi$ at $Re \ll 1$ and the shear-thickening behavior $\nu_{app}(\dot{\gamma})$, both presented in Sec. 4.4.2 could be reproduced with agreement within statistical errors. The runtime of a single MiS instance has been approximately 1 hour on a single CPU$^5$.

---

$^5$Averaging over a strain of 100, which was the same for all values of $\dot{\gamma}$, the number of iterations scales as $\dot{\gamma}^{-1}$. However, with the increased resolution of the temporal scale of the suspended particles (see (4.44)), and therefore the increased computational effort to solve the particle dynamics, the total runtime of a MiS instance was approximately constant.
When the micro-model is launched the fluid velocity is initialized according to
\[ u_m(x, y, t = 0) = \frac{L_m \dot{\gamma}_m}{2} - \dot{\gamma}_m y \] (4.62)
resulting in a homogeneous shear flow field with zero velocity at the center line, \( u_m(x, y = L_m/2) = 0 \).

Initializing the suspended particles is a bit more tricky since we expect also volume fractions \( \phi \) larger than for which a random distribution of non-overlapping particles can be done. Instead, particles are placed on a hexagonal grid that fully covers the domain and satisfies the periodic boundary conditions. The grid has 10 x 12 nodes on which a maximum number of \( N_{\text{max}} = 120 \) particles can be placed. Thus, the maximum volume fraction we can reach with a fully occupied grid is \( \phi_{\text{max}} = 120 \pi R^2 / L_m^2 \approx 0.72 \). Instead of adapting the particle radius \( R \) to set \( \phi \) exactly to the value passed from MaS, \( N_{\text{max}} - N \) randomly chosen particles are removed from the grid. The number of particles \( N \) left on the grid is calculated according to
\[ N = \text{int} \left( \frac{\phi}{\phi_{\text{max}}} N_{\text{max}} + 0.5 \right) \] (4.63)
where \( \text{int}() \) truncates the real valued argument to an integer. With that we introduce a deviation from the volume fraction \( \phi \) as passed from MaS and the actual volume fraction at which the simulation is carried out is \( \phi_{\text{sim}} = N_{\text{max}} \pi R^2 / L_m^2 \). The maximum deviation from \( \phi \) is approximately 0.003, a value much smaller as the confidence range in \( \phi \)-dimension of the parameter space for which extra/inter-polation are trusted to give sufficiently accurate results when values for \( \nu_{\text{app}}(\phi, \dot{\gamma}) \) and the diffusivity tensor \( D(\phi, \dot{\gamma}) \) are requested from the database (see Sec. 4.6.2 for the introduction of and the discussion on such a database and confidence ranges). In any case, the values for \( \nu_{\text{app}} \) and \( D \) obtained from MiS are properly added to the database for \( \phi_{\text{sim}} \) and \( \dot{\gamma} \). Through extra/inter-polation \( \nu_{\text{app}} \) and \( D \) will be returned to MaF and MaS, respectively, for the exact \( \phi \).

After \( N \) particles are placed on the grid their positions are randomly varied. Beginning from a maximum distance of \( d = 2R \) from the original position, in repeated attempts with random angles the particle is tried to put on a new position. When an attempt was not successful, i.e. the new position results in an overlap with another particle, the distance \( d \) is decreased and a new attempt is made with a new angle. This is repeated until a valid position is found which is at the latest satisfied for \( d = 0 \). This method could be described as “melting grid”. Particles are put in perfect order after which as much as possible disorder is introduced.

To obtain the apparent viscosity we followed the method presented in [156, 162] which allows shear stresses to be measured in the bulk instead of measuring frictional forces at the walls which can be done in Couette flow simulations. Shear stresses can be calculated along a horizontal plane \( S \) as the sum over the contributions from the fluid and solid phase as outlined in [162] according to
\[ \Sigma = \sigma_f + \sigma_s + \tau_f + \tau_s \] (4.64)
where \( \sigma \)’s refer to stresses due to pseudo-turbulent motion of the phases. The \( \tau \)’s refer to viscous stresses of the two phases, in the case of the particles it is internal elastic stress which can be calculated as \( \tau_s = (F_{\text{pp}} + F_{\text{fp}})/A_p \). In that \( F_{\text{pp}} \)
denotes the force on a particle due to collisions with other particles, $F_{fp}$ refers to the interaction between fluid and particle. $A_p$ is the intersection area of $S$ and a particle (For further explanations on the calculation of $F_{pp}$ and $F_{fp}$ we refer the reader to [162]). Measurement are carried out in intervals of a strain of $1/4$ and averaged over a total strain of $100$. The time $t^{\text{equil}} = 1/\dot{\gamma}_m$ after initialization is considered as equilibration time needed by the system to evolve into typical flow fields and particle configurations. Only after that measurements contribute to the average of the total shear stress $\langle S \rangle$. The apparent viscosity is then obtained by computing $\nu_{\text{app}} = \langle S \rangle / \dot{\gamma}_m$.

4.5 Shear-induced Diffusion

In this work we concentrate on monodisperse hard-sphere suspensions that are relatively well understood, at least in the limit $Re \ll 1$ and $\phi \ll 1$. This also due to the tremendous theoretical work of Brady and coworkers over the last decades and the Stokesian dynamics simulations they support their theories with, and numerous experimental studies since the first publication by Eckstein, Bailey and Shapiro [169].

In the absence of external fields the transport of particles, and therefore a possible change of the local macroscopic volume fraction $\phi(x_i)$, in a sheared suspension can be attributed to three processes. One of them is Brownian motion due to thermal fluctuations in the momentum exchange between suspended particles and the molecules of the surrounding fluid. It results in short-time diffusivity, i.e. the short-range Brownian motion which, in a dense suspension, is quickly disturbed by the interactions with the surrounding particles. A second process is the shear-induced motion of particles. Although for $Re \lesssim 1$ the interaction of two particles in a shear flow is symmetric and reversible, and might not lead to a permanent displacement (see for example [114]), for larger $Re$ it does (see for example [170] and Sec. 5.2.1). In case of complex hydrodynamical interactions between more than two particles in a dense suspension a permanent displacement of the particles is the normal case. This leads to a shear-induced self-diffusion contribution to the total diffusivity that scales as $D_s \dot{\gamma} \phi$ in leading order for monodisperse hard-sphere suspensions. The “standard” diffusivity induced by a gradient in the local volume fraction, i.e. $\Delta \phi(x)/\Delta x \neq 0$, can be well described by Fick’s law. This gradient diffusivity is defined as the proportionality factor between the concentration gradient and the resulting net flux of particles, i.e. a balancing change in the macroscopic $\phi(x_i)$. However, assuming to be in the regime $Pe \to \infty$, Brownian motion is disabled which would have been the driving process behind long-time diffusivity. In the limit $\nu_f \to 0$, where in the absence of dissipation due to hydrodynamical interaction the suspension would behave as a gas where the kinetic energy of the particles is conserved, we could define a suspension temperature and a Brownian diffusion on the particle scale. However, the suspensions we consider are at finite Reynolds numbers and any motion of the suspended particles would decay quickly if the driving process is turned off.

We are therefore left with shear-induced diffusion as the only underlying process of particle transport. First experiments that showed a diffusion-like migration of particles from high-shear regions to lower shear were published by Leighton & Acrivos [171]. Also a gradient in $\phi(x)$ provokes a gradient in the collisions
rate. This effect is also reflected in an anisotropy in the pair distribution function of the suspended particles. The net effect of these interactions is that particles will migrate from regions of high concentration to low concentration, and from regions of high shear to low shear [172]. Leshansky and Brady could show that this shear-induced gradient diffusivity $D^\nabla$, as being based on the same microstructural processes, is linear proportional to $D^s$, the shear-induced long-time self-diffusivity and can be approximated by [173]

$$D^\nabla \approx \frac{D^s}{S^{eq}(0)}.$$ (4.65)

Here, $S^{eq}(0)$ is the static structure factor corresponding to the hard-sphere suspension at thermodynamic equilibrium. For dilute suspensions it scales as [173]

$$S^{eq}(0) \sim 1 - 8\phi$$ (4.66)

based on excluded volume effects.

In the regime of larger shear rates particle migration and hydrodynamic diffusion are still not well-understood. This is due to the difficulties arising when trying to mathematically model the changing microstructure and collective behavior of the particles. Models considering single particles colliding with other particles can not be applied anymore. The scaling of the long-time self-diffusivity

$$D^s = \dot{\gamma}R^2\hat{D}(\phi)$$ (4.67)

still holds. However, $\hat{D}(\phi)$ can be found to be a strongly increasing function of the particle volume fraction. A best fit to experimental data of sheared hard-sphere suspensions can be found as [172]

$$\hat{D} = 0.5\phi^2 (1 + 0.09 \exp(7\phi))$$ (4.68)

which, with the same functional form but with different numerical coefficients, can also be found in [174]. This is in stark contrast to

$$D^s \sim \dot{\gamma}R^2\phi$$ (4.69)

derived from scaling arguments in the regime of small $\dot{\gamma}$ and $\phi$.

With the multi-scale simulations we aim for maximum flexibility of the framework in respect to the micro model, that is, we want to preset as less as possible about the behavior of the micro-model. We will make use of (4.65) and (4.66), however, not of $D^s \sim \dot{\gamma}\phi$ as this scaling can strictly only be derived under assumptions as $Re \ll 1$ and two-particle interactions. To our knowledge, no analytical derivation of $D = f(\dot{\gamma}, \phi)$ exists today which would be valid also for high shear rates where the typical microstructure differs clearly from that at low or medium shear rates. Furthermore, the linear dependence on $\dot{\gamma}$ and $\phi$ might not hold for suspended particles of non-spherical shape, if they are deformable, or in the presence of additional inter-particle forces.

At the moment we cannot model a concentration gradient in the micro-model by appropriate boundary conditions because it would make particle insertion and deletion necessary. This is a highly non-trivial task in a dynamic simulation because with the artificial deletion/insertion of particles the microstructure will be
frequently disturbed. However, we can measure self-diffusivity of single particles in the homogeneous micro-model. Doing so we are forced to make use of (4.65) and (4.66) to link the self-diffusivity behavior to the gradient diffusivity needed to describe the evolution of the particle concentration on the macro-scale. The scaling (4.66) is only valid for small \( \phi \) and would give negative \( S^{eq}(0) \) for \( \phi > 1/8 \). As no analytically derived extension to larger \( \phi \) for the effect of excluded volume on the structure factor exists, we model the structure factor as a curve

\[
S^{eq}(0) = a \exp(-8b\phi) - c \tag{4.70}
\]

where \( a, b \) and \( c \) are chosen such that \( S^{eq}(0) \) itself and its first derivation in respect to \( \phi \) agrees with (4.66) in the limit \( \phi \to 0 \). The third condition is that it should become zero for \( \phi = \phi_{max} \) enforcing a divergence of \( D^{\nabla} \) at this volume fraction. This condition models the effect that a particle moving towards a region with maximum packing density will not find a way to enter it.

### 4.5.1 Diffusivity Measurements on the Micro-Scale

Self-diffusion coefficients can be calculated using the integral of the velocity auto-correlation function [175, 176], using the relation

\[
D = \frac{1}{2R} \int_{0}^{\infty} \langle v_{p}(t + t_{0})v_{p}(t_{0}) \rangle, \tag{4.71}
\]

It could also be derived with a novel method in which use is made of the fact that an initial suspension microstructure relaxes towards the microstructure typical for long-time shearing proportionally to the gradient diffusion coefficient [175]. In the later case, averaging over a large number of initial conditions is necessary. We however want to measure the diffusivity in the same run we measure also the apparent viscosity. So instead of using an ensemble average we seek for a time-average method. In an interesting work Leshansky and Brady [173] show that the gradient diffusivity can be obtained from measurements of the dynamic structure factor of the solid phase. Based on the relaxation of the average fluctuation in particle number density the structure factor approach seem to offer a much better statistics. However, we leave the exploration of this possibility for future work and will, for now, base our computations on the “classical” direct measurements of the particle displacements in the long-time regime.

The components of the shear-induced self-diffusion tensor can be determined in a simulation by measuring the increase of the mean-square displacements with time after diffusive motion has been established according to

\[
D = \lim_{t \to \infty} \frac{1}{2t} \partial_{t} \langle (x_{p}(t) - x_{p}(t_{0}) - x^{a}(t))(x_{p}(t) - x_{p}(t_{0}) - x^{a}(t)) \rangle \tag{4.72}
\]

\[
\approx \frac{1}{2t} \langle (x_{p}(t) - x_{p}(t_{0}) - x^{a}(t))(x_{p}(t) - x_{p}(t_{0}) - x^{a}(t)) \rangle. \tag{4.73}
\]

Here angle brackets denote an average over all particles in the system. The affine displacement \( x^{a}(t) \) accounts for the movement of a particle in \( x \)-direction with the shear flow. It has to be evaluated from the displacements at every time step and can be based on a linear shear flow assumption. The \( y \)-component in velocity
gradient direction is $x_{y}^a = 0$, the $x$-component is computed according to

$$x_{x}^a(t) = \sum_t \left( v_{p,y}(t) - \frac{\gamma}{2} L_y x_{p,y}(t) \right) \Delta t$$

(4.74)

where $v_{p,y}(t)$ is the $y$-component of the particles velocity at time $t$, $L_y$ is the height of the system and it is assumed that the linear shear flow is that way that $u(x, y = L_y/2) = (0, 0)$. Care has been taken in case a particle crosses a Lees-Edwards boundary in $y$-direction. The current shift of the systems copy should then be added or subtracted, respectively, from the $x$-displacement of that particle.

The preparation of the initial conditions as described in Sec. 4.4.4 aims at initial particle configurations that are very similar to configurations found at a later stage of the simulations. However, an equilibration time is needed to let the system evolve into the typical microstructure characteristic for long-time shearing. This can be also seen in measurements of the displacements $x_p(t) - x_p(0)$. In Fig. 4.21 the flow gradient component of the mean-square displacement is shown for a simulation of 42 particles in the micro systems of size $72 \times 72$ ($\phi \approx 0.25$). For very short times the increase of the displacement scales as $t^2$ characterizing the initial reordering dynamics which are different from the long-time behavior. We can not speak of short-time diffusion here, as in the absence of Brownian motion the initial dynamics of a particle are deterministic. From a strain of approximately $\gamma t \approx 1$ the displacement curve scales as $t$ validating a diffusion description for the shear-induced movement of the particles in the long-time regime. This crossover was found to be independent of the shear rate and volume fractions. Therefore, for all micro-simulations measurements were carried out only after an equilibration time of $t_{\text{equil}} = \gamma^{-1} = t_0$. 

Figure 4.21: Mean square displacement in $y$-direction measured in a micro pocket run with $\gamma$ and $\phi = 0.316$. A strain of $\gamma t \approx 1$ is needed to let the suspension evolve into typical particle configurations after the creation of the initial distribution. The reordering dynamics is characterized by a $\sim t^2$ increase of the squared particle displacements, the long-time dynamics is diffusive, i.e. the mean square displacements scale as $\sim t$. The crossover at $\gamma t \approx 1$ defines an equilibration time $t_{\text{equil}}$ only after which measurements are carried out to determine the diffusivity tensor and the apparent viscosity.
4.5.2 Anisotropic Advection-Diffusion ($\text{MaS/LPAD}$)

When modeling the advection-diffusion of suspended particles to track changes of the local volume fraction $\phi(x_M)$ on the macro-scale we need a method that is capable of a wide range of Peclet numbers and space-varying anisotropic diffusivity tensors.

In general, there are two classes of methods to solve advection-diffusion problems, Eulerian and Lagrangian methods, respectively. In Eulerian approaches the problem is treated based on a mesoscopic description of the advected-diffused particles, the local concentration $c(x)$, at fixed points or volumes. PDE’s are then used to describe the evolution of $c(x)$ over time. Typically, $c(x)$ is defined on a lattice of nodes $x_i$ and finite difference methods for the iterative solution would make sense here. The Lattice-Boltzmann Method has been proven to be a suitable choice in many types of applications (see for example [177]).

However, Eulerian methods are prone to a number of problems such as the not guaranteed positiveness or the conservation of mass [178, 179] in case of large concentration gradients. Eulerian methods are also susceptible to numerical dispersion and artificial oscillations [180]. In particular, the limitation of Peclet number (see for example [181]) is the decisive factor not to use an Eulerian approach. In the present cases of a diffusivity that vanishes for very small shear rates, Eulerian methods would cause a considerable numerical diffusion, for example in a bulk flow with $\dot{\gamma} \approx 0$ and a velocity $|u| \neq 0$ inclined to the lattice where we actually would expect a sharp concentration pattern not to dissolve.

Using a Lagrangian approach the dynamics of individual particles is solved. The dynamics of the particles is modeled based on stochastic differential equations which are consistent with the advection-diffusion equation. Random walk models allow to avoid a number of problems related to the Eulerian approach, and this makes them very attractive in a number of applications (see, for instance, [182, 183]).

A random walk simulation model enables the observation of phenomena on scales much smaller than the spacing of the underlying fluid grid, as well as the tracing of the movement of individual particles, thereby describing the natural processes more accurately, although we will see in this section that limits to this exist. Furthermore, information on integrated properties like residence/settling time or individual tracks can be easily obtained from the simulations. Local concentrations $c(x)$ can be directly calculated from the spatial positions of the particles and when and where required. This can be easily exploited when splitting of the temporal scale offers gain in computation speed.

The advection-diffusion equation for a concentration field $C$ reads

$$\partial_t C = -\nabla \cdot (uC - D \cdot \nabla C)$$  \hspace{1cm} (4.75)

where the change in concentration is coupled to a velocity field $u$ and a diffusion process characterized by the diffusivity tensor $D$. Here, we only consider diagonal diffusivity tensors, e.g.

$$D = \begin{bmatrix} D_{xx} & 0 \\ 0 & D_{yy} \end{bmatrix}$$  \hspace{1cm} (4.76)

in two dimensions. The diffusion dynamics is thus decoupled in these two dimensions. In sheared suspensions small off-diagonal elements might be found [184] due to effects of collective behavior in the presence of concentration gradients. However, we can neglect them in the present work as they are only found significant
for small $P_e$ and are dominated by Brownian contributions in $D$ which are not modeled.

Instead of solving (4.75) one can introduce a number of point particles whose local number density represents $C(x_i)$. To define the dynamics of such particle one can then reinterpret the PDE as a Fokker-Planck type of Langevin equation. We define a stochastic differential equation in the Ito sense

$$d r_p(t) = u(r_p(t)) dt + \sqrt{2D}dW(t)$$

(4.77)

where $r_p(t)$ is the trajectory of a tracer particle that is subject to advection with the velocity field $u$ and a Wiener noise. We assume $W(t)$ to be Gaussian process with mutually independent values. It has a zero mean, $\langle W(t) \rangle = 0$ and a variance $E(W(t)) = \sqrt{\delta t}$I. Note that in our case we expect $D$ to be a function of $r$ and $t$. We therefore cannot treat the terms on the right-hand side of (4.77) independently.

In order to compute a numerical approximation for (4.77) we apply the Euler scheme, i.e.

$$r(t + \Delta t) = u(r(t), t)\Delta t + \sqrt{2D}\Delta W(t).$$

(4.78)

There exist higher order approximations, e.g. those named after Milstein or Heun, the later employing a predictor-corrector step. Although offering higher accuracy, the application of such methods either involve knowledge about spatial derivatives of $D$ whose determination would be an additional source of numerical errors. Or it requires knowledge on the time evolution of $u$ for which an approximation also would lead to numerical difficulties.

The increments $\Delta W$ are computed by a random number generator that produces random numbers in the interval [0, 1], which are then transformed to a Gaussian distribution using the Box-Müller algorithm [185]. We have chosen an implementation of the Mersenne Twister pseudo random number generator [186] for its speed, large periodicity and the good quality in terms of uncorrelatedness of numbers in higher dimensions.

**Mapping Macro Lattice Fields $\rightarrow$ LPAD: Interpolation**

When solving the dynamics of the Lagrangian tracer particles we make use of the advective field $u(x_M)$ and the field of diffusivity $D(x_M)$ which are both data-structures on the regular lattice of the macro scale models with a spacing $\Delta x_M$. The later fact is a consequence of running MiS and interpreting its results only at lattice sites $x_{M,i}$.

To obtain $u(x_p)$ a simple linear interpolation

$$u(r_p) = \sum_i a_i(d_i) \begin{cases} -u(x_1) & \text{if } x_i \text{ is boundary node} \\ u(x_i) & \text{else} \end{cases}$$

(4.79)

over the four nearest lattice nodes at $x_{M,i}$ is carried out. The weights $a_i$ are functions of the normalized distance vector $d_i = (r_p - x_i)/\Delta x_M$ defined as

$$a_1(d_i) = (1 - d_x)(1 - d_y)$$

(4.80)

$$a_2(d_i) = d_x(1 - d_y)$$

(4.81)

$$a_3(d_i) = d_x d_y$$

(4.82)

$$a_4(d_i) = (1 - d_x)d_y$$

(4.83)
which is illustrated in Fig. 4.22(b). This definition is always so that the particle is located in the volume around node $x_1$, the home cell. The neighbor nodes are identified from the quadrants.

This interpolation scheme is well defined in the bulk, however, at the boundaries, special care has to be taken as to the location of the real boundary where $u(x_b) = 0$. Using a Lattice-Boltzmann scheme to solve the evolution of the macroscopic flow field $u(x,t)$ with the LBGK collision operator and a no-slip bounceback-at-the-links boundary condition we expect the hydrodynamic boundary to be located halfway between the fluid boundary node and the solid node the link points towards. To enforce the interpolated $u(r_p)$ at half the distance between fluid and solid boundary nodes, for nodes inside the solid boundary we set $u(x_b) = -u(x_1)$. When solving the particle dynamics, a reflective boundary condition is applied at half the distance between fluid and solid nodes. As the particles do not interact with each other, such a reflection boundary condition is straightforward to implement.

**Advection Test** Testing the advection scheme, a Newtonian laminar flow through a channel confined by vertical walls was simulated and the resulting equilibrated velocity field $u(x_i)$ used to advect tracer particles which were initially $(t = 0)$ randomly distributed over the volume/area of all the inlet nodes, e.g. all fluid nodes for which $x_i = 1$, with a number density of 100 particles per node. In Fig. 4.23 the particle positions after a time $t = 4 \cdot 10^4 \Delta t$ are shown. Note, that the resulting distribution of particles does not show any discretization effects. Also, the closer to the boundary the least the particles have moved, demonstrating the capability of Lagrangian particle methods to resolve even smaller scales than the spacing of the flow field lattice.

\footnote{Strictly speaking, this is only true for boundaries that are aligned with the lattice. For any other inclined or curved boundary, the fluid behaves as if the boundary is slightly shifted towards the fluid by an amount that is a function of the viscosity and the angle of inclination.}
Figure 4.23: Advection test 1: A distribution of tracer particles is initialized in such a way that the particles are randomly distributed over the area of all the lattice nodes at \( x = 1 \). Shown is the tracer particle distribution after 40000 iterations of the advection algorithm. Initially the fluid was at rest after which it was driven in positive \( x \)-direction using a volume force. Note that the smooth particle distribution. The gray scale corresponds to the fluid speed.

However, when repeating the test with an initially homogeneous distribution of tracer particles we see unwanted effects on the evolution of the particle distribution caused at boundaries which are not straight aligned with the lattice of the underlying LBM fluid solver. In Fig. 4.24 the positions of the particles after a time \( t = 1 \cdot 10^5 \Delta t \) are shown. Downstream the corners of the confining vertical walls we see a very dense line of particles and a deserted area next to it. The width of the deserted area is clearly larger than the lattice spacing \( \Delta x \) of the fluid so that this effect is also clearly visible in the integrated local density \( \phi(x_i) \) which is determined on the same lattice as the fluid using a kernel of width \( \Delta x \). Looking for a reason for this, we see that the particle distribution gets compressed on the upstream side of the corners. This is caused by the definition of the boundaries of the particle system to be aligned with the regular rectangular grid of the fluid model. This leads to the interpretation of volume represented by a lattice node as a rectangular shape. Solving the LBM system on a D2Q9 lattice, this is actually not true. Half of the diagonal fluid densities do actually flow “through” a corner created by a solid node. The particles, however, bounce back at the corner. They do this repeatedly with a net movement towards the tip of the corner. Downstream the corner this sweeping leaves particle-empty areas with fluid that flowed “through” a corner. When the flow expands after the constriction, the width of this area exceeds the lattice spacing.

To circumvent this, the boundary conditions of the LBM system can be modified such that also diagonal fluid densities, which actually would have a legal fluid destination node during propagation, are bouncing back. However, this is not enough. Also the interpolation scheme for \( u(r_p) \) has to be modified to make a particle flow parallel to the boundary as it nears it. Using the standard interpolation scheme a particle that just entered the rectangular volume of a node adjacent to a solid node at a corner already feels the non-zero \( x \)-component of \( u \) at the fluid node directly
Figure 4.24: Advection test 2: In this case, the particles were distributed homogeneously over the whole fluid domain. After the same number of iterations as in Fig. 4.23 the particle distribution in the bulk is still homogeneous, as expected for advection with non-compressible flow. However, we see strong deviations from the homogeneity downstream the corners as an effect of boundary conditions for Lagrangian particles. See text for discussion. The gray scale corresponds to the fluid speed.

at this corner. This can be cured by enforcing the interpolation only to be done between \( u(x_1) \) and \( u(x_2) = -u(x_1) \) in this case. However, we found that applying both of the modifications, or only one of them, it does not remove the sweeping effect sufficiently. Anyway, modifying the boundary treatment of the LBM system will lead to a relocation of the hydrodynamic boundary and might be source of instabilities. Given that the macro system in the multi-scale simulations presented in this work does only have straight boundaries aligned with the lattice, we leave the investigation of the boundary problem to future work.

Mapping LPAD \( \rightarrow \) Macro Lattice Fields: Kernel Estimator

The tracer particles are used as an alternative representation of the local volume fraction \( \phi(x_i) \) which we need at the lattice nodes \( x_i \). To map the state of the Lagrangian particle system to a density function defined at the lattice nodes particle positions are integrated to a number density \( P(x_i) \), using a weight function \( K(r_p - x_i, \lambda) \) localized and symmetric around \( x_i \). In its general form it reads

\[
P(x_i) = \frac{1}{N_p \lambda^D} \sum_{p} K \left( \frac{r_p - x_i}{\lambda} \right)
\]

with a bandwidth \( \lambda \). The so-called kernel function \( K(d) \) can be any function satisfying the condition of normalization, i.e. \( \int K(d) du = 1 \). In the literature \( K \) can be found modeled as a Gaussian or Epanechnikov function, amongst others [187]. However, to be consistent with the interpolation scheme (4.79) and (4.80) we use a triangular function

\[
K(d) = \begin{cases} 
(1 - |d_x|)(1 - |d_y|) & \text{if } d_x \leq 1 \land d_y \leq 1 \\
0 & \text{else}
\end{cases}
\]

(4.85)
with \( d = (r_p - x_i)/\Delta x \), \( \Delta x \) equaling the bandwidth \( \lambda \) in (4.84). Because the kernel has a range limited to \( \Delta x_M \) in every dimension, sharp boundaries in \( \phi(x_i) \) can be realized.

Depending on the average/global number density \( P_0 \) we set to represent the global volume fraction \( \phi_0 \) the local suspension volume fraction \( \phi(x_M) \) is then calculated as

\[
\phi(x_M) = \frac{\phi_0}{P_0} P(x_M). \quad (4.86)
\]

How many particles, i.e. what \( P_0 \), we need to represent \( \phi(x_M) \) in sufficient accuracy we will estimate in the diffusion test at the end of this section.

**Flow aligned Diffusivity Tensor**

As described earlier, when mapping local quantities of macro scale fields to initial and boundary conditions of the micro scale model we make use of Galileian invariance in the sense that the reference frame of the micro-model is aligned with the flow, i.e. \( e_{m,x} = u(x_i)/||u(x_i)|| \), and Galilei-transformed in such a way that the mean velocity in the micro model is zero and the velocity gradient has only an \( e_{m,y} \)-component (assuming non-compressible flow). When translating \( D_m \) to \( D_M(x_i) \), \( D_m \) has to be rotated back to the reference frame of the macro scale models. Because diffusive processes are Galileian invariant the Galileian shift does not need to be reversed. In general, a transformation rule can be defined as

\[
D_M(x_i) = sD \cdot R^{-1}(\theta) D_m(x_i) \quad \text{with} \quad \theta = \angle(u, e_{M,x}), \quad (4.87)
\]

using the diffusivity-scale factor \( sD \). \( R \) is the rotation matrix

\[
R(\theta) = \begin{bmatrix}
\cos \theta & -\sin \theta \\
\sin \theta & \cos \theta
\end{bmatrix}. \quad (4.88)
\]

with the property \( R^{-1}(\theta) = R(-\theta) \). The transformation (4.87), involving matrix calculation with trigonometric functions as elements, is rather costly in computation. Instead, having all necessary information at hand at this stage of the computations, we can use the diagonal flow-aligned diffusivity tensor directly. The random walk is carried out in the flow aligned reference frame with the base vectors \( e_{m,\parallel}(x_i) \), and the Euler scheme (4.78) becomes

\[
\hat{r}_{p,x}(t + \Delta t) = [u_x(r_p(t), t) + w_\parallel \cdot e_{m,\parallel}(x_i) + w_\perp \cdot e_{m,\perp}(x_i)]\Delta t \quad (4.89)
\]

\[
\hat{r}_{p,y}(t + \Delta t) = [u_y(r_p(t), t) + w_\parallel \cdot e_{m,\parallel}(x_i) + w_\perp \cdot e_{m,\perp}(x_i)]\Delta t \quad (4.90)
\]

where

\[
w_\parallel = \sqrt{2D_{m,\parallel}} \Delta W_1 \quad (4.91)
\]

\[
w_\perp = \sqrt{2D_{m,\perp}} \Delta W_2. \quad (4.92)
\]

The time step \( \Delta t \) does not necessarily correspond to the time step of \( \text{MaF} \). The time scales can be split and a number of \( k \) sub-steps can be carried out in the interval

\[
\Delta t = \frac{1}{k} \cdot \Delta t_M. \quad (4.93)
\]
**Diffusion Test** The flow-aligned anisotropic diffusion we validate by an approximation of the solution of a well known diffusion problem, the decay of a Dirac-distributed initial concentration field with a time-independent diffusivity tensor $D$. The solution of the temporal evolution of the concentration field and the initial condition $C(x, t) = \delta(x - 0)$ reads

$$C(x, t) = \frac{1}{(4\pi t)^{3/2} \sqrt{\det D}} \exp \left( -\frac{x D^{-1} x}{4t} \right).$$

(4.94)

Three otherwise similar runs of the LPAD system have been initialized with $N_p = 10^3, 10^4, 10^5$ particles at $r_p(t = 0) = r_0$, i.e. a delta distribution. The components of the flow-aligned diffusivity tensor were set to $D_\parallel = 3 \cdot 10^{-3}, D_\perp = 5 \cdot 10^{-4}$ and it was assumed that the diffusion takes places on top of the advection in a constant homogeneous velocity field $u(x_i) = (0.05, 0.02) \forall x_i$. The LPAD model has then been run for $10^3$ iterations. Plots of the resulting $P(x_i, t)$ are shown in Fig. 4.25. As expected, the deviations from the analytical solution decrease with the number of tracer particles used. To estimate the total error we measured an error function

$$\epsilon = \sum_{x_i} |\phi(x_i, t) - C(x_i, t)|$$

(4.95)

and can confirm that the decrease of the total error behaves as $\epsilon(N_p) \sim N_p^{-2/(D+4)}$ [187] with $D = 2$. Choosing a small time step $\Delta t$, i.e. larger $k$, to increase the accuracy of the Euler scheme does not make much sense if $N_p$ is too small to allow for a good sampling of $\phi(x_i)$. However, also a large $N_p$ does not necessarily decrease the total error if the error is dominated by deviations caused by a too small $\Delta t$. Further elaboration and practical suggestions concerning this discussion can be found in [188]. We found that for $N_p = 10^5$ the LPAD approximation gives a mean relative error of less than 0.015. Facing the relatively high computational costs of tracer particle systems we consider this a good approximation of the solution (4.94). Translating these findings into the application of (4.86) we estimated $P_0 = 1000$ and $k = 1$ (in (4.93)) to be a good choice to represent the advection-diffusion on the macro-scale.

### 4.6 A Database coupled in

#### 4.6.1 Serial vs. concurrent coupling

Regarding the availability of data resulting from the micro-model, e.g. its solution for a sufficiently large range of parameters, as input to the macro-model one can distinguish between concurrent coupling methods [189] and serial coupling. The first case is the most common and is sometimes also referred to as “equation-free” approach [190]. In it, the communication between micro-scale and macro-scale models is “on-the-fly”, e.g. in a data-driven multi-scale model the micro-scale model is initialized, run and evaluated for specific boundary condition and parameters whenever the macro-scale model needs its input, that is, as the computation goes on. The second way of coupling is called serial coupling in which the solution of the micro-scale model is precalculated for an expected range of parameters. The coupling merely exists between the macro-scale model and a precalculated database.
Figure 4.25: Tracer particle positions (left) and number densities $P(x_M, t)$ (right) as obtained by (4.84) from the tracer particle positions $r_p$ after a time $t = 10^3$ starting from an initial distribution $r_p(t = 0) = (50, 80)$. The diffusion according to $D_\parallel = 3 \cdot 10^{-3}$, $D_\perp = 5 \cdot 10^{-4}$ took place on top of the advection with a homogeneous flow field with velocity $u = (0.05, 0.02)$ to which the diffusivity tensor was aligned to. As expected, $P(x_M, t = 10^3)$ has an inclined anisotropic Gaussian shape centered at $x = (100, 100)$. With increasing global number density $P_0$ the numerical approximations converge to the analytical solution (4.94).
and can be described as parameter passing. This is advantageous if sufficient data is already available at the beginning of the macro-scale simulation. Examples for serial coupling are the usage of data from MD simulations to define transition probabilities in Monte Carlo schemes, atomistic interaction potentials derived from quantum mechanics calculations and used in MD models, or the pre-estimation of transport coefficients later used in advection-diffusion problems. In principle, also series of data derived in experiments from a lab model which is then used in a computational model leads to the same classification. As long as the constitutive equation needed depends on only a few variables and the variable space that will be visited by the macro-model is more or less known so that precalculation can be limited to this region in variable space, serial coupling is reasonable.

At runtime of the macro-scale model the two strategies might differ only in execution time of the macro-model run. Whether the data given to the macro-scale model comes from the micro-model directly or a database in between - it is supposed to be the same.

4.6.2 A Hybrid: Database filled on-the-fly

During the course of an HMM simulation \( \left( \frac{X_M}{\Delta x_M} \right)^D \frac{T_M}{\Delta t_M} \) local iterations/collisions have to be carried out on the macro level and thus the same number of micro-pockets would have to be called to provide \( \nu_{\text{app}} \) and \( D_{xy} \) as collision parameters at every point \( (x_M, t_M) \). It is obvious that we, following this scheme, would repeatedly launch micro-pockets for the same set of parameters \( (\phi, \dot{\gamma}) \), or for a set of parameters that does not differ too much from a parameter set already sampled. It is clear that if we write micro-pocket results to a database and apply extra/interpolation schemes to extract \( \nu \) and \( D_{xy} \) in state-space regions we already sampled in sufficient density, we could reduce the number of micro-pockets which have to be launched by several orders of magnitude.

In doing so we combine the advantages of serial and concurrent coupling. If the needed data does already exist in the database, or can be extracted from it by extra/interpolation, we make use of precalculated data. If no data is available, a micro-pocket is run and, when terminated, a copy of the results is archived in the database. This way only as many micro-pockets are run as are needed to solve the macro-model. And we have the possibility to use solutions that already exist for a range of parameters prior to the execution of the macro-model. This data might come from precalculations with the same micro-model carried out for another study, another micro-model (e.g. for parameter ranges the first micro-model is not valid or instable for), or from experiments.

**Extra- and Interpolation**

Of course, the database approach makes only sense when a majority of requests can be answered through a fast interpolation between existing data points. Additionally we need an according extrapolation scheme that allows advances into unknown parameter space regions with the same step size we assume for a tolerable distance between data points as a good compromise between accuracy and speed of the simulation. Because obtaining data points is expensive we want to keep the density of data points low. On the other hand, even the best extra- and
interpolation scheme misses important characteristics if data density is too low. A certain knowledge about the smoothness of the behavior $\nu_{app}$ and $D_{xy}$ as functions of $\dot{\gamma}$ and $\phi$ is therefore necessary to set the tolerance range in $\phi$ and $\dot{\gamma}$ dimension, respectively.

We have chosen hard spheres as suspension particles because, albeit numerous open questions, its general behavior is relatively well known (see section 4.1). From the plots of $\nu_{app}(\phi)$ in fig. 4.13 we see that sampling the smooth and monotonously increasing curve can be done with a step size of $\tau_{\phi} = 0.25$ to capture its behavior in sufficient accuracy (what “sufficient” means remains open, see discussion on sensibility of the coupled system in section 4.8). We do not expect characteristics of the curve that are on a smaller $\phi$-scale. From the plots of $\nu_{app}(\Re(\dot{\gamma}))$ in fig. 4.14 we also see smooth and monotonously increasing $\nu_{app}$ behavior, this time on a logarithmic scale. The curve can be well represented by data points with a distance of $\dot{\gamma}_2 - \dot{\gamma}_1 = 0.25 \dot{\gamma}_1$ where a factor is used to achieve equidistant points on the logarithmic scale.

One reason for not choosing an approach in which a multi-dimensional function of some form and number of degrees of freedom is fitted to the data points is, that although we know the general behavior in this particular application (hard-sphere suspension), the aim of the simulation framework is to be as general as possible. It should be possible to easily replace the micro-system MiS by another micro-model for which we do not know which behavior to expect. This will however require some adaptive accuracy control. With the functional form of the fitting function we would imply such knowledge to keep the number of free parameters small in favor of the feasibility of the regression problem. Particularly when extrapolating, a fitted curve is subject to great uncertainty since it may reflect the method used to construct the curve as much as it reflects the data. A second reason is, addressing the feasibility of the regression, that with increasing dimensionality of the problem the stability and the accuracy of the parameter-search algorithm becomes a critical question. In particular, if we increase the number of parameters to increase the flexibility of the fitting function. Also, the initial value problem would require knowledge of the behavior which should be the more complete and accurate the more free parameters we allow the function to have. For more details and description of problems in fitting functions to datasets the reader is kindly referred to appropriate literature, for example [191].

Another disadvantage of the described approaches to inter/extra-polation in more than 1 dimension is an unwanted correlation between the behavior in these dimensions. This is also tru for piecewise polynomial inter/extra-polation (splines). In other words, through the undiscriminated spread of $\partial^{(0,1)} Q_j/\partial Q_i^{(0,1)}$ also in $Q_{k \neq i}$ dimension, particularly where there is only one data point neighbor in the $Q_i$ dimension, this might lead to strong over- or underestimation.

To minimize the effects addressed so far, we implemented the following approach. Let $Q_{1..q_{max}}$ be the set of $q_{max}$ given points in the $D_p + D_d$ dimensional data space, where the subspace of dimension $D_p$ is spanned by the parameters $Q^p_{i,p}$, i.e.

\begin{align}
Q^p_{1,q} &= \phi_q, \\
Q^p_{2,q} &= \dot{\gamma}_q
\end{align}

in the present use case with $D_p = 2$. The data subspace of dimension $D_d$ is spanned
by the data \(Q_{d,j,q}^d\) attached to the parameter vector \(Q_p\), i.e.

\[
Q_{1,q}^d = \nu_{\text{app}}(\phi_q, \dot{\gamma}_q), \quad (4.98)
\]

\[
Q_{2,q}^d = D_{xx}(\phi_q, \dot{\gamma}_q), \quad (4.99)
\]

\[
Q_{3,q}^d = D_{yy}(\phi_q, \dot{\gamma}_q). \quad (4.100)
\]

Figure 4.26: To obtain an estimate for \(R_d^d\) for the given parameters \(R_p\) as a point lying in the confidence range around at least one existing data point \(Q_1\), in a weighted average ((4.103)) slopes \(P_{ji,pq}\) are computed from the vectors \(a_{qp}\) between existing database entries \(Q_p\) and used to calculate \(R_p\) as a point on \(A\). If more than two points \(Q_p\) can be found in an extended confidence range around \(Q_q\), an average over all computable slopes \(P_{ji,pq}\) is used.

As described above, we trust an inter/extra-polation to retrieve \(R_d^d(R_p)\) as long as there is at least one parameter point \(Q_p^d\) which is within confidence range, i.e.

\[
\left( \sum_{i=1}^{D_p} \frac{(Q_{i,q}^p - R_i^p)^2}{\tau_i^2} \right)^{\frac{1}{2}} \leq 1. \quad (4.101)
\]

Together with \(Q_q^d\) also the slopes \(P_{ji} = \partial Q_{i,q}^d / \partial Q_{i,q}^p\) are stored. This allows to compute \(R_d^d(R_p)\) as a point on the plane \(A\) from

\[
P_{j,q}^d = Q_{j,q}^d + \sum_{i=1}^{D_p} P_{ji,q}(R_i^p - Q_{i,q}^p). \quad (4.102)
\]

In case the condition (4.101) is met with other data points \(q\), too, a simple linear interpolation is done according to

\[
R_d^d = \frac{\sum_q (1 - \frac{1}{\tau_i}(Q_i^p - R_i^p))R_{d,q}^d}{\sum_q (1 - \frac{1}{\tau_i}(Q_i^p - R_i^p))} \quad (4.103)
\]

with weights \((1 - ..)\) which are 0 at the boundary of the confidence area.

The slopes \(P_{ji,q}\) at \(Q_p^q\) can be computed if there are at least two more points \(Q_q^p\) and \(Q_p^q\) in a, by a factor \(C\) extended, confidence range. If not then \(P_{ji,q} = 0 \forall ji\). The extension by a factor of approximately \(C = 1.5\) is necessary to allow the determination of \(P_{ji,q}\) at points that are just outside the confidence range given by
in (4.101), a situation characteristic of data point distributions that are enriched on-the-fly. The slopes $P_{ji,q}$ are determined from $D^p$ vectors $\mathbf{P}_i = 1 \ldots D^p$ in the plane defined by two linear independent vectors $\mathbf{a}_{qp} = \mathbf{Q}_p - \mathbf{Q}_q$ and $\mathbf{a}_{qr}$, pointing from one data point to another, and the point of origin, $\mathbf{Q}_q$. The vectors $\mathbf{P}_i$ are another basis of the subspace spanned by $\mathbf{a}_{qp}$ and $\mathbf{a}_{qr}$, and the origin $\mathbf{Q}_q$, with the condition that $P_{k,i} = 0 \forall k \neq i$, $1 \leq k \leq D^p$. Then the slope is simply $P_{ji} = P_j / P_i$.

When searching for points $\mathbf{Q}_p^p$ and $\mathbf{Q}_r^p$, an additional condition,

$$\frac{\mathbf{a}_{qp} \cdot \mathbf{a}_{qr}}{|\mathbf{a}_{qp}| |\mathbf{a}_{qr}|} < 0.7,$$

is applied to make sure that $\mathbf{a}_{qp}$ and $\mathbf{a}_{qr}$ are “sufficiently” linear independent so that variations (statistical errors) in $Q^d$ do not lead to great variations in $P_{ji}$.

Furthermore, by including another weight factor $w_{ipq} = \frac{1}{\tau_i}(Q_{i,p}^p - Q_{i,q}^p)$ the isotropy in respect to the direction to which respect the gradient is computed is broken with the intent to minimize the correlation of $P_{ji}$ in other directions than $Q^p_i$.

The calculations described are rather intense and would have to be repeated every time the database is asked for some value of $\nu_{app}$ and $D_{xy}$. However, quite a part of the calculation is the same for a large range of $\phi$ and $\dot{\gamma}$ so we decided to precalculate and hold in memory lattice fields $\nu_{app}(\phi_i, \dot{\gamma}_j)$ and $D_{xy}(\phi_i, \dot{\gamma}_j)$ every time a data point is added to the database. A lattice spacing $\phi_{i+1} - \phi_i \approx 0.2\tau_\phi$ is found to be fine enough. Using a data structure based on a lattice allows for quick access of the data with the help of simple linear interpolation schemes.

In Fig. 4.27 subsets of the databases as created while running test simulations similar to those described in Sec. 4.8 are shown together with the according lattice fields for $\nu_{app}$ and $D_{yy}$.

Classification of the Role of the Database

The coupling of a database between the macro-scale models and the micro-model can be classified in different ways, depending on the role one gives the database and which component of the multi-scale system is supposed to be so smart to decide whether a micro-scale simulation has to be carried out or not.

- One way would be in the sense of the HMM coupling. As a part of the macro-model it could serve as the replacement for the constitutive equations needed in the collision step. Only if the macro-model is incomplete, i.e. its constitutive equations are not valid in a range of parameters, e.g. at a localized singularity, the micro-model needs to be launched to provide this information.

- The second way to classify the system macro-model-database-micromodel could be to understand and implement the database as a smart conduit, following the CxA idea and the functionality the MUSCLE framework provides. The macro-model does call the micro-model via the database conduit which has a memory and the functionality to process stored data into the information demanded by the macro-model.
Figure 4.27: Example database for two test runs of the multi-scale system for $\phi_0 = 0.3$ (crosses) and $\phi_0 = 0.5$ (stars) showing the entries of $\nu_{app}$ (top) and $D_L$ (bottom) together with the field of extra/inter-polated values points (dots) representing the confidence regions around data points obtained from runs of the micro-model.

- Another way would be to consider the database itself as the micro-model as it actually provides the information needed by the macro-scale model. A fully resolved simulation of a sheared suspension would then be considered as another functionality of the database agent that is employed if extra/interpolation can not be applied.

**Parallelization?**

Although spatial dependency in the macro model doesn’t allow for asynchronous updates of the macro fluid nodes, it might be worth to think about parallelization of the collision step within one iteration at $t_M$. From computational point of view the HMM approach leads to a coupling where the macro model functions as a master calling an arbitrary number of slaves, being the micro “pockets”. The later are independent from each other. No information has to be exchanged between pockets at their runtime. In this situation a farming approach as parallelization strategy could be applied. In that, micro pockets delivering $\nu_{app}(x_M, t_M)$ and $D_{xy}(x_M, t_M)$ for each macro lattice site $x_M$ and time $t_M$ could be launched in parallel running on
different processors. However, using the database coupled in between macro and micro models reduces the actual number of calls of the micro model so significantly that hardly any two sets of $\nu_{\text{app}}(\phi_{1,2}, \dot{\gamma}_{1,2})$ and $D_{xy}(\phi_{1,2}, \dot{\gamma}_{1,2})$ have to be called at the same $t_M$. Except for the first iterations, where the barn is almost empty and the biggest changes in the state of the macro model take place, only very seldom a new data point has to be added. In systems with very complicated geometries, and therefore very different local flow conditions, the number of database requests might be increased during the first time steps of the integration. However, due to the intrinsic exploitation of rotational and Galileian symmetry, the actually visited parameter space will not be much larger, and we do not expect big changes from the situation of low request density, even for problems in 3 dimensions.

![Figure 4.28: The time line of calls of the micro-model (points) and a density plot binning those events in time bins of width 1000$\Delta t_M$ illustrating the rareness in which the micro-model has to be called when a database is used to prevent redundant simulation runs.](image)

In Fig. 4.28 the number of micro pockets called are plotted over macro time steps from one of the HMM suspension runs together with a density curve measured as calls per 1000 iterations. We see that in the beginning of the simulation much more pockets have to be called in comparison to later stages of the equilibration process. Nonetheless, it happens only once that two pockets are called at the same time. A farming approach would not make sense here and was not implemented. The only parallelization strategies appropriate would be domain decomposition of the micro models which is, in principle, realizable in a straightforward manner due to Lees-Edwards boundary conditions (see Sec. 4.4.2 for discussion on that), or by extrapolation of the systems way through the parameter space and opportunistically calling a number of micro-models to fill the database in advance.

### 4.7 Other Suspension Flow Models

In this section two macroscopic suspension flow models will be briefly introduced. They will be used to discuss properties and resulting profiles together with those obtained with the multi-scale model described in this work.
4.7.1 Suspension balance model

Following [192], for the suspension as a whole and for the particle phase conservation of mass and momentum can be used to set up a balance model. Averaging these quantities over the whole domain and all phases we can write

$$\nabla \cdot \langle \mathbf{u} \rangle = 0 \tag{4.106}$$

$$D_t \langle \rho \mathbf{u} \rangle = \langle \mathbf{b} \rangle + \nabla \cdot \langle \Sigma \rangle \tag{4.107}$$

where $D_t = \delta t + \langle \mathbf{u} \rangle \cdot \nabla$ is the convective derivative and $\langle \mathbf{b} \rangle$ the average over body forces acting on fluid and solid phase. Both, fluid and solid phase contribute to the stress in the suspension,

$$\langle \Sigma \rangle = -\langle p \rangle_f \mathbf{I} + 2\eta_f \langle \mathbf{e} \rangle + \langle \Sigma \rangle_p \tag{4.108}$$

where $\langle p \rangle_f$ is the averaged pressure in the fluid, $\eta_f$ the suspending fluid viscosity, and $\langle \mathbf{e} \rangle$ the rate of strain tensor (for the bulk). The scalar shear rate in terms of this tensor is given by $\dot{\gamma} = (\langle \mathbf{e} \rangle \cdot \langle \mathbf{e} \rangle)^{1/2}$.

The conservation of mass for the solid phase governs the evolution of the particle concentration $\phi$, i.e.

$$\delta t \phi + \nabla \cdot \phi \langle \mathbf{u} \rangle_p = 0 \tag{4.109}$$

where $\langle \mathbf{u} \rangle_p$ is the average solid phase velocity. The particle flux, relative to the average suspension velocity, is given by

$$\mathbf{N} = \phi (\langle \mathbf{u} \rangle_p - \langle \mathbf{u} \rangle) \tag{4.110}$$

and combining it with the mass conservation,

$$\delta t \phi + \nabla \cdot (\phi \langle \mathbf{u} \rangle - \mathbf{N}) = 0, \tag{4.111}$$

gives

$$\delta t \phi + \langle \mathbf{u} \rangle \cdot \nabla \phi = \nabla \cdot \mathbf{N}. \tag{4.112}$$

With that we have an equation for the evolution of the particle concentration on the macro scale which assumes knowledge about the fluxes $\mathbf{N}$. Thinking about where to split the scales, treating particle concentrations with help of (4.112) and measuring particle fluxes at the micro level could be one option. However, there would be the demand to explicitly model the gradient in the particle concentration on the micro level to account for the fluxes induced by it. This is not straightforwardly implementable in MiS without full knowledge of the spatio-temporal evolution of particle distributions in the surrounding areas of the pockets. We therefore need to model the concentration gradient-induced particle fluxes on the macro scale. The local diffusivity tensor is then obtained in a round-about way via the shear-induced self-diffusivity.

Following the suspension balance model into more detail we can relate the flux to gradients in the solid phase stress.

The difference between suspension velocity and solid velocity is related to the drag force on the particles. The local single particle Reynolds number in suspensions is generally low (order of $10^{-4}$) and we can assume the drag force to scale linearly with the relative velocity to the fluid,

$$\langle \mathbf{u} \rangle - \langle \mathbf{u} \rangle_p = M(\eta_f, a, \phi) \langle \mathbf{F} \rangle_p. \tag{4.113}$$
\( M \) is the inverse drag coefficient and accounts for the hindered mobility of an average particle due to the other particles in the suspension and is an unknown function of the fluid viscosity \( \eta_f \), the particle radius \( a \), and the particle density \( \phi \). We also assume isotropy of \( M \) here. Considering the momentum balance for the solid phase
\[
\rho_p \phi D_t^p \langle \mathbf{u} \rangle_p = \langle \mathbf{b} \rangle_p + \langle \mathbf{F} \rangle_p + \nabla \cdot \langle \Sigma \rangle_p
\]  
where the convective derivative \( D_t^p \) is now following the average particle motion. \( \langle \Sigma \rangle_p \) is the stress in the solid phase, and \( \langle \mathbf{b} \rangle_p \) the solid body force which is \( \langle \mathbf{b} \rangle_p \) for neutrally buoyant particles. If the motion on the macro scale is also at low Reynolds number the inertial term in (4.114) can be neglected and the drag force can be written as
\[
\langle \mathbf{F} \rangle_p = -\nabla \cdot \langle \Sigma \rangle_p.
\]  
This means, the particle flux is driven by stress gradients which is the point where micro models have to be found for.

Following the approach of Nott and Brady [192] the averaged solid-phase stress consists of the particle pressure, a deviatoric shear stress, and normal-stress differences. Later, Brady and Morris [119] could show that due to Brownian motion, short-range non-hydrodynamic particle interaction, or surface roughness, normal stresses scale linearly with the shear rate. Differences in normal stresses have the same microstructural origin as the solid-phase pressure, and therefore have the same scaling with the shear rate and volume fraction. Following [193] we can thus combine these two types of stresses into one quantity \( \Pi Z \) where \( \Pi \) is the isotropic solid-phase pressure and \( Z = I - (\eta_f/\Pi)\langle \chi \rangle \) incorporates the normal stresses. For the constitutive equation for the particle stress at small particle Reynolds number one can write
\[
\langle \Sigma \rangle_p = -\Pi Z + 2\eta_f \eta \langle \phi \rangle (\mathbf{e}).
\]  
The solid-phase pressure \( \Pi \) as a function of the volume fraction and shear rate is given by
\[
\Pi = \eta_f \dot{\gamma} p(\phi)
\]  
where \( p(\phi) \) is a monotonically increasing function of \( \phi \) which can be given by
\[
p(\phi) \sim \phi^2 \left(1 - \frac{\phi}{\phi_{max}}\right)^{-2}
\]  
following the derivations from a microstructural model that puts the total stress in relation to the microstructure [119]. The solid phase viscosity can be modeled via the Krieger-Dougherty relation for the relative viscosity \( \eta_r \) of the suspension. In general, \( \eta_r = 1 \) for \( \phi = 0 \) and we write
\[
\eta_r = 1 + \eta_s
\]  
Using the semi-empirical Krieger-Dougherty relation \( \eta_r \) can be modeled as (4.1).

### 4.7.2 Diffusive-Flux Model

Assuming neutral buoyancy and the absence of Brownian motion, Leighton and Acrivos [171] proposed to model the particle flux due to diffusive motion as
\[
\mathbf{N} = \mathbf{N}_d + \mathbf{N}_c
\]
with $N_\eta$ the flux due to spatial variations in viscosity and $N_c$ the contribution to the flux induced by hydrodynamical particle interaction, i.e. shear-induced diffusion. Phillips et al. [194] proposed these flux terms to be

$$
N_\eta = -K_\eta a^2 \phi \dot{\gamma} \phi \nabla \ln \eta \quad (4.121)
$$

$$
N_c = -K_c a^2 \phi \nabla \dot{\gamma} \phi \quad (4.122)
$$

which are based on the scaling arguments derived by Leighton and Acrivos [171]. In (4.121) $a$ is the diameter of the particles and $\eta(\phi)$ the apparent viscosity as a function of $\phi$ only. The coefficients $K_\eta$ and $K_c$ have to be determined empirically. In [193] $K_\eta = 0.62$ and $K_c = 0.42$ were used in a comparison of resulting particle concentration profiles $\phi(y)$ with those resulting from the suspension balance model and experimental works in different settings resembling viscometer geometries. Keeping the ratio $K_\eta/K_c$ constant, and varying the coefficients the time scale of the dynamics of $\phi(y,t)$ can be changed. This is one of the consequences of the conservation equation

$$
\partial_t \phi = -\nabla (N_\eta + N_c). \quad (4.123)
$$

The model describes the particle concentration field $\phi(x,t)$ as resulting from the particles moving from regions of high interaction frequency to regions of lower interaction frequency [172]. This is analogous to a statistical gas model and in that similar to the modeling of the particle flux in this work (see Sec. 4.5.2). The main difference is the flux $N_\eta$ due to a gradient in the viscosity. This is motivated by the fact that the strength of hydrodynamic interactions increases with the viscosity $\eta$ analogous to an increased effective cross section in the collision dynamics of two approaching particles. In this, the diffusive-flux model is based on an assumption we do not make in our multi-scale model where a fully resolved micro-model is used. In a micro-model with traceable particles this effect will be included in the measured shear-induced diffusivity $D$.

The diffusive-flux model was successful in predicting particle concentration profiles in several benchmark problems [195]. However, its profiles do not satisfactorily agree with those seen for cone-and-plate and parallel plate geometries. In [193] a possible explanation for this is given pointing out that the interaction-induced flux $N_c$ is isotropic. The fact that the component of the shear-induced diffusivity tensor in direction of the shear gradient is only approximately half the other components might indeed have an impact on the dynamics of $\phi(x,t)$ in curved geometries. In our micro-model the components of the flow aligned diffusivity tensor are measured independently, not simplifying diffusion to an isotropic process. In [193] Fang et al. introduce a flow-aligned diffusivity tensor into the diffusion-flux model. However, also with this modification the diffusive-flux model does not predict particle concentration profiles as good as the suspension balance model.

### 4.8 Simulations and Results

To test the concept of the HMM model for suspensions we applied it to the simulation of the pressure driven flow of a 2-dimensional hard-sphere suspension through a straight channel. As we apply periodic boundaries in $x$-direction, instead of a pressure difference at the in- and outlet we mapped $\Delta p$ to a volume force according to $G_x = \Delta p/L_x$ which is used in (4.19) and (4.27). We carried out simulations with
volume forces of different magnitude $|G| = G_x = 1 \cdot 10^{-14}, 1 \cdot 10^{-13}, 1 \cdot 10^{-12}, 1 \cdot 10^{-11}$ to drive the flow parallel to the channel axis. Furthermore, we repeated this series of simulations for global volume fractions of $\phi_0 = 0.3$ and $\phi_0 = 0.5$.

The macro-models are initialized with the fluid at rest, i.e. $u(x_M, t_M = 0) = 0$ for all $x_M$, and a random but homogeneous distribution of tracer particles of the MaS model. More specifically, a number of 1000 tracer particles are randomly distributed over the volume of each node around $x_M$, for both cases of global volume fraction $\phi_0$. This results in a homogeneous volume fraction field $\phi(x_M, t_M = 0)$ with small variations of less than 2% (see Sec. 4.5.2 for a discussion on that).

We measured not only the volume fraction and velocity profile that are typically accessible by experiment and the primary result of models for the macroscopic flow of suspensions. We also measured shear rate profiles $\dot{\gamma}(y)$, the diffusivity component $D_\perp$ perpendicular to the flow direction, and kept track of the viscosity profile $\nu_{\text{app}}(y)$ to be able to gain insight into the coupling of those quantities and their dynamics.

As discussed in Sec. 4.6.2 the database approach allows to re-use any knowledge on the micro-model prior to the multi-scale simulation by putting it into the database in appropriate format. We defined a minimum shear rate $\dot{\gamma}_{\text{min}} = 5 \cdot 10^{-14}$ which is small enough to assume that viscosity and diffusivity will assume values close enough to their values for $\dot{\gamma} = 0$ and inserted data points sampling the semi-empirical Krieger-Dougherty relation for the increase of $\nu_{\text{app}}$ with $\phi$ (see (4.1)) and components of the diffusivity tensor $D_\parallel = D_\perp = 0$ which equals the zero-shear limits for shear-induced diffusivity. The lower shear-rate limit was not hit again after the equilibration phase.

### 4.8.1 Profile Development Dynamics

Starting from a homogeneous volume fraction $\phi(x, t = 0)$ over the channel and zero velocity, the system of coupled flow and diffusion needs an equilibration time $t_{\text{equil}}$ to fully develop the profiles. Only for $t \geq t_{\text{equil}}$ averages over the fluctuating profiles can be taken. Actually, the total equilibration dynamics consists of the dynamics of flow and diffusion, and we can distinguish between a $t_{\text{equil,flow}}$ and $t_{\text{equil,diff}}$. Over all ranges of the shear rate and all volume forces applied we find for the ratio of the characteristic time scales

$$\frac{\nu_{\text{app}}}{D} \gg 1$$

and therefore $t_{\text{equil,diff}} \gg t_{\text{equil,flow}}$. The velocity profile develops from a flat zero profile within a few thousand $\Delta t_M$. The diffusive dynamics of the volume fraction, however, is much slower, a behavior also found in experiments [196]. To illustrate this behavior and the effects associated with it in Fig. 4.29 the shear rate and volume fraction profiles for two distinguished times in the development of one system is shown which had a global volume fraction of $\phi = 0.3$ and was driven with a volume force $G_x = 1 \cdot 10^{-11}$. At this highest volume force the shear rate close to the walls reaches values corresponding to shear Reynolds numbers $Re_p > 1$ where the micro-model shows the onset of shear-thickening. At $t_1 = 7000 \Delta t_M \approx t_{\text{equil,flow}}$ the volume fraction profile does not yet differ much from the homogeneous initial distribution. The velocity profile, however, has fully developed by then and shows the characteristics of a shear-thickening fluid, i.e. a higher curvature around the
center than in the rest of the channel a fact also reflected in the slightly steeper $\dot{\gamma}$-profile in this region. Over the time to $t^{\text{equil,diff}}$ the volume fraction profile develops the typical maximum at the center of the channel, seen in profiles for $t_2 > t^{\text{equil,diff}}$. In this course the shear rate profile changes from a form typical for shear-thickening fluids to that of a shear-thinning fluid due to the stronger increase of the viscosity with $\phi$ as with the shear rate $\dot{\gamma}$.

This difference in the time scales of diffusion and momentum transport also influences the way the system moves through the parameter space during equilibration. In Fig. 4.31 the sampled regions in the parameter space are shown as recorded in the databases for the different simulation runs.

Nott and Brady have shown [192] that an estimation for the total length $L^{\text{equil}}$ the suspension needs to travel before equilibration is established can be given by

$$\frac{L^{\text{equil}}}{H} \sim \frac{1}{12 \hat{D}(\phi)} \frac{H^2}{R}$$

(4.125)

where $H$ is the width of the channel $H = L_M$, and $\hat{D}(\phi)$ is the scaling of the self-diffusivity with $\phi$ (see (4.68)). Whether the maximum velocity or a mean velocity based on the total flux has to be taken as a reference $v_{\text{ref}}$ for a calculation of $L^{\text{equil}} = t^{\text{equil}} v_{\text{ref}}$ is not clear but might make a difference since the velocity profiles change for the different settings of volume force and global volume fraction. In Fig. 4.30 the equilibration times $t^{\text{equil}}$ are plotted for the different simulation settings that were obtained in test runs prior to the actual runs that were used for the measurement and averaging of profiles and that therefore were much longer. In the plot we see that indeed the equilibrium time behaves as

$$t^{\text{equil}} \sim \frac{1}{v_{\text{ref}}(G_x) \hat{D}}$$

(4.126)

Using the assumption $v_{\text{ref}} \sim G_x$ here, we find that the mean velocity $v_{\text{ref}}$ does approximately linearly increase with the applied volume force despite the presence
of shear-thickening for the highest-shear regions in the runs at maximum volume force.

We have made use of this relation of $t_{\text{equil}}$ to $G_x$ and $\phi_0$ and scaled the diffusivity in such a manner that $t_{\text{equil}} \approx 10000$. With that we could accelerate the diffusive dynamics and reduce the runtime of the macro-model by a factor of almost four orders of magnitude in case of $\phi_0 = 0.3$ and $G_x = 1 \cdot 10^{-14}$ without violating the temporal scale separation of flow and diffusive dynamics during equilibration. This can also be seen in the chronology of the entries in the database. In the beginning points with increasing $\dot{\gamma}$ are added without deviations from $\phi = \phi_0$. Only after that the maximum shear rate for $\phi = \phi_0$ is reached the diffusive dynamics leads to a widening of the distribution of entries in the $\phi$-dimension.

### 4.8.2 Equilibrium Profiles: Results and Discussion

In Fig. 4.32 and Fig. 4.33 the equilibrium profiles are shown for $\phi_0 = 0.3$ and $\phi_0 = 0.5$, respectively. They all were obtained by averaging them over at least $N = 300$ independent samples at times separated by an autocorrelation time $t_{\text{corr}} \approx 5000$ estimated from the exponential decay of the correlation function $C(\delta t)$ for measurements of the suspension velocity $v_x(x_M)$ at times $t_1$ and $t_2 = t_1 + \delta t$ (averaged over all $x_M$ and $t_1$). The error bars on the data points show the average fluctuation of the considered quantity over the measurement time instead of the statistical error which would be a factor $\sqrt{N}$ smaller. Fluctuations in the profiles have their origin in the random nature of the Lagrangian model for the macroscopic advection-diffusion of $\phi(x_M)$. In that it might be similar to the measurements on real suspensions that have a finite particle number and are also averaged over a finite time. However, because the non-interacting tracer particles in MaS have a different dynamics these two types of fluctuations cannot be directly related here.

In favor of a better understanding of how the features of the profiles develop in a suspension that is driven through a straight channel, how one quantity influences the other, and the feedback loops that exist in the coupled system Fig. 4.34 shows...
Figure 4.31: Shown are the entries to the database in the parameter space spanned by φ and ˙γ together with the confidence range around the entries. Every point corresponds to a run of the micro-model. Starting from the top left plot the successive enrichment of the database is shown with each simulation run under different conditions of global volume fraction φ₀ and volume force. All runs started from a zero velocity, i.e. zero shear profile ˙γ(y), and flat φ(y) profile. Before the first run (4.1) was used to prepare the viscosity entries for a minimal ˙γ_{min} = 5 \cdot 10^{-14}. The diffusivity for those entries were set to D∥ = D⊥ = 0.
Figure 4.32: Equilibrium profiles of local volume fraction $\phi$, diffusivity component $D_\perp$ in shear gradient direction, velocity $v_x$ (normalized to a total flux of 1), shear rate $\dot{\gamma}$, and viscosity $\nu$ for $\phi_0 = 0.3$. All in macro-scale lattice units.
Figure 4.33: Equilibrium profiles of local volume fraction $\phi$, diffusivity component $D_\perp$ in shear gradient direction, velocity $v_x$ (normalized to a total flux of 1), shear rate $\dot{\gamma}$, and viscosity $\nu$ for $\phi_0 = 0.5$. All in macro-scale lattice units. A straightforward explanation for the secondary peaks for the highest $G_x$, also seen for $\phi_0 = 0.3$ in Fig. 4.32, we cannot offer at this stage of the research. However, we also find similar effects in the experimental results by Lyon and Leel [123], for comparison shown in Fig. 4.35.
With \( a \rightarrow b \) we denote the positive coupling between \( a \) and \( b \). More precisely, \( b \) is a monotonously increasing function of \( a \). With \( a \rightarrow b \) we denote the fact that \( b \) will decrease when \( a \) increases. A number of feedback loops can be identified. With \( \Gamma \) we denote the flux of particles away from the considered volume element as a result of the local \( D \).

The general shape of the volume fraction and velocity profiles resulting from our simulations agree quantitatively well with those predicted by the diffusive flux model, the suspension balance model and the experimental results by Lyon and Leal which are reproduced for comparison in Fig. 4.35 and Fig. 4.36. In all cases of global volume fraction \( \phi_0 \) and applied volume forces \( G_x \) the suspended particles tend to move to the center region of the channel. This can be explained by the positive coupling \( \dot{\gamma} \rightarrow D \) whereby particles from high-shear regions near the wall diffuse much stronger than those in the lower-shear regions at the center which leads to an effective flux towards the center. This tendency is enhanced through the viscosity \( \nu_{\text{app}}(y) \) which increases with \( \phi(y) \) and lowers the shear rate \( \dot{\gamma}(y) \).

This positive feedback loop can be found as coupling \( D \rightarrow \Gamma \rightarrow \phi \rightarrow \nu \rightarrow \dot{\gamma} \rightarrow D \sim D \rightarrow D \) in Fig. 4.34. The effective flux between positions \( y_1 \) and \( y_2 \) will vanish if \( D(y_1)\phi(y_1) = \Gamma_{1\rightarrow2} = \Gamma_{2\rightarrow1} = D(y_2)\phi(y_2) \), expressed as the negative feedback \( D \rightarrow \Gamma \rightarrow \phi \rightarrow \nu \rightarrow \dot{\gamma} \) in Fig. 4.34. Since \( \nu_{\text{app}} \) is a monotonously increasing function of \( \phi \) the viscosity in the center region is much higher than in the rest of the channel. This coupling is depicted as \( \phi \rightarrow \nu \) in Fig. 4.34. The higher viscosity leads then to a smaller shear rate in the center region \( (\nu \rightarrow \dot{\gamma}) \) resulting in blunted velocity profiles. The negative feedback \( \nu \rightarrow \dot{\gamma} \rightarrow \nu \) for shear rates large enough for shear-thickening exists but does not play a prominent role as discussed below.

The global volume fraction \( \phi_0 \) has a clear effect on the equilibrium profiles. In general, we find that for \( \phi_0 = 0.3 \) the profile of the local volume fraction over the channel width has a much more pronounced maximum at the center of the channel. This is in agreement with the experimental results by Lyon and Leal which have been included in Fig. 4.36 for comparative purposes. This tendency is also immediately clear from the plots of the diffusivity Fig. 4.32(b) where we find a much more pronounced valley within the central region of the channel for \( \phi_0 = 0.3 \) in comparison to that for \( \phi_0 = 0.5 \). Comparing the velocity profiles in 4.32 and 4.33 we see that for \( \phi_0 = 0.3 \) the profile only slightly differs from a Newtonian velocity profile for the lowest volume force whereas the profiles for \( \phi_0 = 0.5 \) are equally strongly blunted for all applied volume forces. Also Lyon and Leal find this
Figure 4.35: Experimental results for the velocity and the volume fraction profiles as obtained by Lyon and Leal [123] together with the equilibrium profiles as predicted by the diffusive flux and the suspension balance models. (Pictures taken from Fang et al. [193]). On the left the results for $\phi_0 = 0.3$ are shown, on the right for $\phi_0 = 0.5$.

Figure 4.36: Experimental results by Lyon and Leal comparing profiles of different global volume fractions $\phi_0 = \phi_{\text{bulk}}$. (Taken from [123])
influence on the global volume fraction. All their experiments were carried out at very small particle Reynolds numbers of the order of $10^{-6}$. The particle Reynolds numbers of the lowest volume force applied in our work compare well with that and so do the profiles for $G_x = 10^{-14}$.

Comparing the profiles for different volume forces $G_x$ applied we find a strong dependency in the case of $\phi_0 = 0.3$. The overall shape of the profile $\phi(y)$ stays approximately the same except for secondary peaks for the highest $G_x$, for which we cannot offer a consistent explanation here. However, we also find similar effects in the experimental results by Lyon and Leal [123]. Comparing the profiles for different volume forces $G_x$ applied we find a strong dependency in the case of $\phi_0 = 0.3$. The overall shape of the profile $\phi(y)$ stays approximately the same. However, the difference of the local volume fraction at the center $\phi(x_{\text{center}})$ − $\phi(x_{\text{wall}})$ is a clearly increasing function of $G_x$. This is not so prominent for $\phi_0 = 0.5$ where volume fraction profiles are more compressed. The dependence on $G_x$ is in clear contrast to the predictions by the diffusive flux and the suspension balance model which both result in profiles independent from the total flux. We find this dependency also for shear rates for which no shear-thickening could be observed which actually agrees with the Newtonian flow assumption in both models. The reason for this must therefore be a much stronger shear-rate dependency of the diffusivity resulting from our micro-model. In both other models the shear rate is a linear function of $\dot{\gamma}$, with an additional dependency on the viscosity gradient in case of the diffusive flux model.

One of the most obvious differences in all the profiles measured by Lyon and Leal is that in their measurements the volume fraction in the outer 20% of the channel width drops down to zero at the wall for almost all settings of flow speed, global volume fraction and the ratio $H/R$. However, Lyon and Leal themselves believe that this is an artifact of their experimental setup namely a lower signal-to-noise ratio that is an intrinsic property of the measuring technique (LDV) they were using. In later comparative measurements with optical microscopy they indeed found higher volume fractions near the walls and confirmed this assumption on the shortcoming of LDV. Although presented for bidisperse suspensions, and therefore strictly speaking not comparable to the monodisperse systems discussed here, regarding the values near the wall the profiles in [124], the local volume fractions near the wall obtained by optical microscopy are much more similar to those presented here and the predictions of the diffusive-flux and suspension balance models.

Lyon and Leal also investigated the influence of the ratio $H/R$ on the profiles but could not find changes larger than the experimental uncertainty for the particle and channel sizes they used. Also the diffusive flux model results in profiles that do not know a dependence on the particle size. On the other hand, the suspension balance model has an explicit dependency on the ratio $H/R$. However, this dependency is relatively weak and based on this model we can estimate that changes would hardly be distinguishable from the fluctuations in our simulations. The suspension balance model contradicts the results from Lyon and Leal regarding the $H/R$ dependence of the velocity fluctuations and is therefore reason for discussion. Also, the influence of the particle size on the profiles vanishes for $H/R \to \infty$ which matches very well the HMM modeling approach based on the separation of the scales of the micro and the macro-dynamics. As described in Sec. 4.2.4 the choice of the models on the micro and macro-scale puts strong limits on the range of parameters we
can investigate, so investigating the influence of the spatial scale separation is not straightforward with our setup and the reason for differences found in the results will be hard to track down since with the spatial scale separation other scalings will change as well.

The Reynolds numbers in the experiments by Lyon and Leal, taking the maximum channel velocity as a reference, were no larger than $10^{-5}$. Even when these are converted to particle shear Reynolds numbers we find $Re_p \ll 1$. That means, no shear-thickening will have occurred in their experiments. For the used particle size Brownian motion was negligible as well, so no significant dependence of the apparent viscosity on the shear rate can be expected at all. In our simulations presented here we deliberately also applied volume forces which resulted in particle shear Reynolds numbers larger than 1 where the suspension began to thicken. However, for the highest volume force applied, $G_x = 10^{-11}$ (macro-scale lattice units), also the difference in particle concentration $\phi(x_{center}) - \phi(x_{wall})$ was maximized and the influence of $\phi$ on $\nu_{app}$ was stronger than that of $\dot{\gamma}$. As a consequence, no clear effect of shear-thickening could be observed in the equilibrium profiles. The differences in the profiles with increasing volume force can be attributed to the increase of the shear-induced diffusivity which was stronger than the decrease caused by the $\nu_{app} \rightarrow \dot{\gamma} \rightarrow D$ feedback. To investigate this behavior the application of even higher volume forces would be desirable. In the current micro-macro coupling, however, the limits of the micro-model, both, in terms of maximum $\phi$ as well as $\dot{\gamma}_m$ were reached.

Most prominent for the simulations of the suspension with a global volume ratio $\phi = 0.3$, we find a dependency on the flow rate, i.e. volume force the suspension is driven through the channel. This is already true for flow rates that do not lead to shear-thickening ruling out this phenomenon for an explanation. It suggests that the functional dependence of $\nu_{app}$ on the volume fraction, and the functional form of $D$ in dependence on shear rate and volume fraction are of a form that lead to different equilibrium profiles in contrast to the equilibrium profile resulting from diffusive-flux or suspension balance model which predict no dependence on the flow rate. With the numerous micro-model runs at different combinations of $\dot{\gamma}_m$ and $D_m$ we, in principle, have enough data collected to discuss these dependencies. However, as the primary intent of this work was to present the computational side of the multi-scale model a discussion in appropriate level of detail is postponed to a future, more physics related work.

A discussion on the error in the presented results proves difficult. As the dynamic coupling of the field quantities $\phi(x), D(x), \nu(x)$ and $\dot{\gamma}(x)$ themselves form a highly sensible feedback network, an estimation of the total error is a highly non-trivial task. Besides that, a sensitivity analysis with respect to the spatio-temporal resolution of the macro-model could not be carried out because changing $\Delta x_M$ and $\Delta t_M$ would change the scaling between micro- and macro-models and the error associated with it. Too many variables change in an almost unpredictable way when one of them is varied. Although we deliberately reduced the dimensionality of the macro-model to exclude phenomena that are related to the break of such enforced symmetries, the coupled system has to be approached almost like an experimentalist approaches his subject of study. That is, it is not always possible to fix one degree of freedom to study the other under pure conditions. Additional tricks, as that of using a database, also come with the trade-off of errors related to
Of course, a main shortcoming of the simulations in this chapter is the fact that it all has been modeled in only two dimensions. The functional dependence of $\nu_{\text{app}}$ on $\phi$ and $\dot{\gamma}$ will be different in three dimensions where suspended particles can avoid high stresses by moving in the extra dimension. The same is true for the shear-induced diffusivity where hydrodynamical particle-particle interactions induce a movement of the particles in the extra dimension and therefore smaller components in the first two. The functional form of the dependencies won’t change. Coefficients and exponents might, however, and will lead to a different equilibrium on the macro-scale. The profiles resulting from the 2D model in this work nevertheless show the same features of those obtained in experimental channel flow proving that the behavior in 2D is very similar to that in 3D. Another aspect is that in the experiments by Lyon and Leal and the two models the macroscopic flow is assumed to be laminar and fluctuations in the volume fraction are small in size. In this case, the 2D flow as modeled in this work is equivalent to the 3D flow between two plates.

4.9 Outlook

We are convinced that the multi-scale model and the simulations carried out so far summarized in this chapter offers a good base for the extension to other problems. While building the software care has been taken of the flexibility to include other micro-models and to extend the coupling between micro and macro-models.

As mentioned in the preceding discussion the current model was reduced in dimensionality in several aspects. Besides the obvious need to replace the 2D micro-model by a 3D version to more realistically model apparent viscosity and particle diffusivity of a real suspension, the extension of the dimensions of the macro-model offers the ability to study additional phenomena. When the periodic simulation box of current 2D channel is extended from 1 lattice node to a length comparable or larger than the channel width, non-laminar flow can be studied on the macro-scale. In case of a density ratio $\rho_s/\rho_f \neq 1$ the suspended particles will feel an additional centrifugal acceleration in curved flow. It would make it necessary to model this additional coupling of the macroscopic curvature to a particle force. The result will be an altered apparent viscosity and shear-induced diffusivity due to the non-zero average fluid flux through the array of moving particles. The centrifugal or -petal (depending on $\rho_s/\rho_f$) particle drift will lead to a break of the left-right symmetry of the advection-diffusion dynamics of $\phi(x_M)$ perpendicular to the flow lines. Phenomena like or similar to the clustering of particles in turbulent aerosols, i.e. due to the sweep-stick mechanism, will be possible to investigate based on a realistic modeling of the dynamics of the microstructure.

Another possible extension goes back to the original intent of the suspension model, namely the simulation of blood flow on the cell level. The complex rheology of blood is the result of the many degrees of freedom of the cells suspended in the plasma and their various ways to interact. With a hematocrit of approximately 45% blood is a rather dense suspension. Most of the “solid” phase volume red blood cells (RBC) account for. RBC’s are shaped as biconcave disks, with a dumbbell-shaped cross section and a torus-like rim. This distinctive shape has large influence on the flow in large blood vessels where it can optimize the flow properties of blood,
e.g. stabilization of laminar flow at higher Reynolds numbers and minimization of platelet scatter, which suppresses their positive influence on the formation of atherosclerosis in those large vessels. In large blood vessels, RBC’s sometimes form rouleaux’s, i.e. stacks of RBC’s flat side next to flat side. This occurs more or less often in dependence on levels of certain serum proteins, e.g. during inflammation. Furthermore, the deformability of RBC’s lets the blood adapt to flow effectively through tiny capillaries with less resistance than plasma by itself. As a comparison, if all human hemoglobin were free in the plasma rather than being contained in RBC’s, blood would be too viscous for the cardiovascular system to function effectively. The deformability is also dependent on chemical factors, as is the attractive interactions between RBC’s that causes RBC’s to stick together, with negative effects on the capacity to transport oxygen due to the reduction of their free surface. Modeling of the blood’s microstructure under consideration of such additionally triggered interactions and coupling this to a continuum model as described in this chapter could offer a great tool to study blood flow properties under variation of certain proteins, chemical factors etc.

The multi-scale coupling with the use of the database as described in Sec. 4.6.2 could also offer great improvement in the simulations of hemodynamics in complex geometries. As the parameters shear rate $\dot{\gamma}$ and $\phi$, and other scalar quantities, are invariant under the scale transformation (except for a constant scale factor), results can be re-used in a running simulation although the absolute velocity of the flow at the macro point $(x_M, t_M)$ and its direction might be different. This Galileian and Rotational invariance reduces the actual calls of the micro-model considerably, also in very complex flow geometries.

Besides RBC’s, other important constituents of blood are white blood cells, or leukocytes, and platelets. Leukocytes make up for only 0.7% of the blood volume and might therefore be negligible in bulk flow, however, they have considerable influence on the flow along the vessel walls and therefore particularly on the flow through narrow vessels [197, 198]. Leukocytes tend to stick to and slowly roll along the endothelium driven by the momentum exchange with the remaining blood. In a HMM type of multi-scale simulation a micro-model for the boundary could include leukocytes and therefore resolve their influence on the flow, an effect difficult to model at the continuum level.

Platelets, or thrombocytes, are small (2-3 $\mu$m), irregularly-shaped anuclear cells with an average lifespan of normally just 5 to 9 days. Platelets play a fundamental role in hemostasis (and therefore thrombosis, see also Sec. 3.4.1) and are a natural source of growth factors that play a significant role in the repair and regeneration of connective tissues. Platelets have the feature that they can be activated, i.e. brought into a state in which they aggregate and adhere to collagen that is exposed by damaged endothelial tissue. This activation is triggered, besides by stimulation through inflammatory products, by shear stress they experience in the flow. In a HMM type of multi-scale simulation a platelet concentration field and an additional scalar activation field could transport platelet properties over the macro-scale. On the micro-scale, single platelets are modeled and their activation and adhesion to walls simulated. The formation of platelet deposit in arbitrary complex geometries could be studied.

All the above mentioned extensions will introduce additional parameters necessary to describe the states of the particles with increased degree of freedom. Also
their representation in macroscopic fields that show advective and/or diffusive dynamics will have to be tracked, e.g. a field describing the probability density to find an activated platelet or the average strength of activation of the platelets in a volume \( \Delta x_M^3 \). In principle, they will be treated similarly to the field of particle concentration, i.e. local volume fraction, as demonstrated in this work. With the increase of complexity of the micro-model, reflected in the increase of the number of parameters, the coupling using an intermediate database with inter/extra-polation functionality will show its strength because with every new dimension in the parameter space the actually sampled parameter space will tend to be much smaller than the full hypercube which corners are defined by the minimal and maximal values of each parameter. The effective speedup of the multi-scale computation will therefore increase with the number of parameters.

### 4.10 Conclusion

As a proof of concept we demonstrated that the described multi-scale model is capable of the simulation of the macroscopic flow of a suspension of hard spheres that are several orders of magnitudes smaller than the channel width. The profiles of the local volume fraction \( \phi(y) \) and the velocity \( v_x(y) \) resulting from the simulations at volume forces varied over four orders of magnitude not only show the same general trend as the diffusive-flux and the suspension balance model. They also show tendencies that can only be found in measurements of real suspensions. Moreover, we find a dependency on the flow rate, i.e. volume force used to drive the suspension through the channel on the profiles which is not present in the diffusive flux and suspension balance model. Whether this is a behavior also found for real suspension is not clear from the literature known to us.

We could show that the application of the HMM idea to the macroscopic suspension flow using a fully resolved micro-model as a replacement for the constitutive equations for \( \nu_{app}(x_M) \) and \( D(x_M) \) in the macro-models not only reduces the computational effort by several orders of magnitude. It is clear that by exploiting of Galileian invariance the HMM approach enable that type of simulations in the first place.

We also could show that with the use of a database and extra/inter-polation functionality in a confidence range around data points the actual number of micro-model calls could be reduced to a maximum of 30 for each multi-scale simulation carried out. This limited the total runtime of the multi-scale simulation to 2 days on a single CPU. Without the database, e.g. at every macroscopic point in space-time \( (x_M, t_M) \) a micro-model would have to be executed to be able to carry out collision steps in the macro-models, a standard HMM type of multi-scale simulation for the considered suspension system would be infeasible.

Besides the main spatio-temporal scale separation between the macro-models and the micro-model leading to the HMM multi-scale modeling approach we identified further scale separations that were exploited in a more technical manner to either increase the numerical stability or to accelerate the dynamics. First, in the micro-model the dynamics of particles in close approach has to be resolved on a much finer spatio-temporal scale to maintain stability of the numerical integration of their trajectories. The fluid was assumed to be constant at the sub-time steps the particle dynamics was solved at. With that, we could increase the applied
shear rates by almost an order of magnitude allowing us to study shear-thickening in a larger region of particle Reynolds numbers with the fully resolved micro-model. The other scale separation we made use of is the separation of the time-scales of the slow diffusive dynamics in $Ma_S$ from the fast fluid dynamics modeled in $Ma_F$. We identified how this separation scales with the volume fraction $\phi_0$ and the volume forces applied to the macro fluid and made use of it by accelerating the diffusive dynamics.
Chapter 5

Particle Clustering in Shear-Thickening Hard-Sphere Suspensions

In this chapter a more detailed look at the microstructure and its dynamics of non-Brownian suspensions of monodisperse neutrally buoyant hard-spheres is taken.

5.1 Microstructure of Sheared Suspensions

At low particle Reynolds numbers $Re = 4\dot{\gamma}R_p/\nu_f$ the behavior of viscous suspensions is relatively well understood. In the dilute limit, hydrodynamic interactions between the particles can be neglected and the suspension exhibits Newtonian rheology with a relative viscosity $\nu_{app}/\nu_f = 1 + 2.5\phi$, where $\nu_{app}$ is the apparent viscosity of the suspension and $\nu_f$ the constant viscosity of the suspending fluid. For larger volume fractions $\phi$ the relative viscosity is well characterized by the Krieger-Dougherty relation (4.1) describing the non-linear increase of the apparent viscosity with the increase of solid volumes in the system. The relation (4.1) is semi-empirical. When leaving the Stokes limit the understanding of the suspension’s behavior can be even less supported by analytical derivations. Simulations offer to study the dynamics of the mesoscopic particles and the suspending fluid and serve as an important tool to probe particle microstructure in flow. Parallel to experimental works [199] simulation studies have explored the structural asymmetry [119] and related this to shear-induced normal stresses [117, 200, 201].

In the dilute limit, simple-shear flow with finite shear rates over a sphere was considered using matched asymptotic expansions in the Stokes limit [202]. In this work it was found that weak inertia causes an increase in the particle contribution to the viscosity scaling as $Re^{3/2}$ and the breaking of the symmetry of the Stokes flow particle trajectories which was related to the increase of normal stresses scaling with $Re$. In addition it is found that inertial effects give rise to a non-isotropic normal stress. Single particles and their positive effect on the apparent viscosity have also been studied in [203]. In the same work it was also observed that for $Re > 1$ the angular velocity of single particles as well as that of clusters of particles decreases...
which is then shown to be correlated with the increase of the viscosity.

Expressing the suspension microstructure through the pair-distribution function with the help of accelerated Stokesian dynamics in [201] it is demonstrated how the resulting anisotropy in the pair-distribution function is correlated with the non-Newtonian behavior of the suspension. Allowing for fully resolved flow fields, and therefore for a maximum of realism describing hydrodynamical interactions between particles, the Lattice-Boltzmann method (LBM) has been used to study several aspects of sheared suspensions. In two-dimensional systems the hydrodynamical interaction of pairs of particles and the consequences on their trajectory have been examined [149, 204]. Detailed motion of of isolated particles was studies in [205] and the authors could show how fluid inertia changes the trajectories of particle pairs in simple shear flow. In this work closed trajectories [206] of particles approaching in Stokes flow have been found to be replaced by spiraling and reversing trajectories for $Re > 1$. The authors also discuss how Reynolds stress due to fluctuational motion and stress due to particle acceleration are caused by inertia as an additional mechanism for momentum transport. In [207, 208] distributions of gap widths between particles was studied and found that particle-particle distances become ever smaller with higher shear rates leading to a divergence of inter-particle forces. The resulting shear-thickening behavior was also investigated by means of LBM [115, 103]. In [115] it was found that shear thickening is related to the enhanced contribution from the solid phase to the total shear stress as $Re$ increases. In [209] the clustering of particles in a sheared suspension at Reynolds numbers up to 10 was studied and the increased particle clustering was argued to lead to the increase in viscosity of the suspension. The clustering of suspended particles as the main mechanism leading to shear-thickening was also identified in [102, 103, 208].

In studies of suspension rheology so far attention was mostly paid to the suspensions response to variations of the volume fraction $\phi$ in the Stokes limit and/or the particle Reynolds number $Re \sim \dot{\gamma}$. In this work we will concentrate on the effect of particle inertia on microstructure and shear-thickening of the suspension by varying the ratio $M = \rho_s/\rho_f$ between the mass density of the solid phase $\rho_s$ and that of the fluid $\rho_f$ of a dense suspension system of which particles occupy a volume fraction $\phi = 0.4$. The mass densities $\rho_s$ and $\rho_f$ apply only to the inertia of the corresponding phase. Gravitational effects were not implemented resulting in a neutrally-buoyant system. The volume fraction was chosen such that it is high enough to undergo clearly visible continuous shear-thickening (disorder→order transition) but low enough not to show ordering effects and discontinuous shear-thickening (order→disorder transition) present for $\phi > \phi_c \gtrapprox 0.6^1$.

5.1.1 Simulations

Using the LB methods described in Sec. 4.4.1 we carried out shear-flow simulations of 2-dimensional neutrally-buoyant monodisperse hard-sphere suspensions with a Newtonian suspending fluid. Within this framework the improved momentum exchange algorithm [210] (or see Sec. 4.4.3) was used to restore Galilean invariance of the LBM particles. To shear the suspension we applied Lees-Edwards boundary conditions [113] as described in Sec. 4.4.2. Lees-Edwards boundary conditions

---

1An exact value for the critical volume fraction marking the crossover between continuous and discontinuous shear-thickening is not known to us for 2D but should be higher than that for 3D, $\phi_c^{3D} \approx 0.55$. See also Sec. 4.1
(LEbc) are particularly useful in studies of a suspension's microstructure since it is not necessary to introduce impenetrable moving walls to drive the shear flow. Thus, the particles can freely move in a homogeneous quasi-infinite system. Of course, finite-size effects will set in when the correlation length (e.g., the typical size of particle clusters) in the system reaches the dimensions of the simulation box. We set the system size to \( L_x \times L_x = 259 \times 259 \) and the particle radius to \( R_p = 8.044 \) to maximize the reachable Reynolds number which is limited by the typical cluster size to expect\(^2\) and on the Mach number constraint while keeping the accuracy of the spatial resolution at a rather high level. Lubrication corrections were applied to the force and torque of the particles whenever particles come so close that the fluid between them cannot be resolved by LBM (see Sec. 4.4.1). However, the dynamics of pressure, velocity, and stress field of the fluid between particles might deviate considerably from the pure two-particle assumption in dense suspensions (see also Sec. 4.4.3). Aiming at investigations of the microstructure we chose for the higher resolution over reduction of computational costs. In all simulations of suspensions reported here a number of 132 particles was suspended in the fluid resulting in a volume fraction \( \phi = 0.4 \). The statistics presented in this chapter were obtained over a total strain of \( \dot{\gamma} t = 100 \) for all combinations of \( M \) and \( \dot{\gamma} \). For the lowest shear rates considered, \( \dot{\gamma} = 1 \cdot 10^{-5} \), a CPU time of approximately 72 hours were necessary. Measurements were carried after strains of 0.5.

### 5.1.2 Geometrical properties of Particle Clusters

In Fig. 5.1 snapshots from the evolution of sheared suspensions with solid-fluid mass density ratios \( M = \rho_s/\rho_f = 1, 10, 100 \) are shown for low (\( \text{Re} = 0.0155 \)) and high (\( \text{Re} = 1.5529 \)) shear particle Reynolds numbers. At low Reynolds numbers particles hydrodynamically interact at comparatively larger distances. As for the Stokes flow limit, the small pressure increase between two slowly approaching particles is sufficient to maintain distances seldom less than \( \approx R_p/5 \). At medium Stokes numbers \( \mathcal{S} \sim MR_{\text{e}} \) particle inertia leads to closer particle approaches and particles begin to cluster in small chains connected by regions of short-range deviations from the average fluid pressure. At higher \( \mathcal{S} \) particle clusters grow longer and wider, and also the correlation length of fluid pressure increases.

In the simulations we recorded the statistics of particle-particle separation vectors. From the magnitude of such vectors it was determined if particle pairs fulfill the cluster condition, i.e. \( r_{12} < 2.2 \cdot R_p \) where \( r_{12} = |r_2 - r_1| \) is the distance between the centers of the particles. This criterion is derived from the observation that the pair-distribution function \( g(r_{12}) \) has a maximum at this distance. Applying this cluster condition we aim to identify those particles that have a strong hydrodynamical bond with each other. The strength of hydrodynamical inter-particle friction\(^3\) diverges as \( \sim (r_{12} - 2R_p)^{-1} \) the closer the gap between the particles.

---

\(^2\) Which should not be larger than half the system size to minimize the possibility of percolating clusters which would lead to diverging stresses in shear flow.

\(^3\) It is possible to term this interaction 'friction' because for small \( r_{12} \) it scales linearly with the relative velocity \( \mathbf{F} \sim -\dot{r}_{12} \).
Figure 5.1: Snapshots of sheared suspensions with a volume fraction $\phi = 0.40$ for different values of mass density ratio $M$ and Reynolds numbers. Darker areas of the fluid correspond to higher pressure. The shear is that way that fluid and particles at the top border move to the right and those at the lower border in approximately the same speed to the left.
Cluster Pair Distribution Function

In contrast to the classical pair distribution function $g(r_{12})$ which takes every possible pair of particles into account, we measured a cluster pair distribution function defined as the probability to find another particle inside the same cluster at a separation $r_{12}$. In Fig. 5.2 such cluster pair distributions are shown for small and large Reynolds numbers and mass density ratios $M = 1, 10, 100$. In contrast to the classical pair distribution secondary rings of high probability can be clearly found for distances a multiple of $2R_p$ which makes it possible to investigate strength and orientational tendencies of particle clustering. For $M = 1$ we find hardly any change with the Reynolds number. The still relatively low Reynolds number does not lead to a clearly visible asymmetry in the pair distribution which can be found for larger $Re$ (see for example [205, 201, 200, 211]). This corresponds to the still rather uncorrelated distribution of particles in Fig. 5.1(d). When the mass density ratio is increased to $M = 10$ an asymmetry can be clearly found for $Re = 1.5229$. Also, for these parameters an overall increase of the probability can be observed. Corresponding to that, in [200] it was found that $g(r_{12}) \sim Pe^{0.7} \sim Re^{0.7}$ in direction of compression. Both facts are related to the onset of formation of chain-like particle clusters that are mostly aligned in the compression direction of the shear flow. For $M = 100$, while at $Re = 0.0155$ no changes can be found to the measurements for the smaller $M$, at the high Reynolds number the asymmetry effect is clearly present as well as an increased probability to find cluster pairs also at larger distances. The asymmetry in Fig. 5.2(f) is comparable to that in Fig. 5.2(e) but a fuzzy fifth ring becomes visible. This suggests that principle geometrical characteristics of particle cluster do not change with increasing typical size.

Fractal Dimension of Particle Clusters

From the statistics we obtained on the particle clusters the probability to find clusters of a certain mass $N$, i.e. the number of particles in the cluster, and of a radius of gyration $R$ is shown in Fig. 5.3 for the same subset of $M$ and $Re$. Together with it the two extreme cases of cluster forms are plotted in dashed lines. The line corresponds to the increase of $R$ with $N$ for a perfect chain of particles whereas the lowest curve shows the behavior of $R$ with $N$ for the most compact disk-shaped agglomeration of particles in a perfect hexagonal packing. Regardless the parameters $M$ and $Re$, the cloud of higher probability to find a cluster of mass $N$ and dimension $R$ can be found closer to the case of chain-like than that of compact cluster shape. Hardly any cluster can be found that could be characterized as compact. The probability distribution has a maximum for $N = 1$ and $R = 2R_p$ which means that still most of the particles are well separated from all other particles. However, for larger $M$ and $Re$ the probability increases to find larger clusters as well while the mean of the distribution follows a power law $R \sim N^{1/D}$. The fractal dimension $D$ could be found to agree for all combinations of $M$ and $Re$ within statistical errors. Its mean could be determined to $D = 1.15$ illustrating the sparse volume occupation of particle clusters. This together with the condition that particles have to be ‘connected’ implies that clusters tend to be more of chain-like shape.
Figure 5.2: Cluster pair distribution function for a subset of considered parameters $M$ and $Re$ taking into account all particles in the same cluster as the reference particle. At low $St$ the distribution of cluster pairs is almost isotropic while for larger $St$ the strength of anisotropy increases. The probability to find another particle in the compression direction increases with higher momentum faster than the probability to find particles of the same cluster in direction of extension.
Figure 5.3: Probability density to find a cluster consisting of $N$ particles and a radius of gyration of $R$ for the same parameter sets as in Fig. 5.1 and Fig. 5.2. Regardless the mass density ratio $M$ and Reynolds number, an average fractal dimension of $D = 1.15$ can be found which agrees for all $M$ and $Re$ within statistical errors. The low value of $D$ illustrates the tendency of clusters to assume elongated shapes. The dashed lines show the extreme cases of a straight chain of particles and that of densest packing.
Cluster Size Distribution and Typical Cluster Size

In Fig. 5.4 the distributions \( p_c(N) \) of the cluster size are shown for the considered combinations of mass density ratios \( M \) and Reynolds numbers \( Re \) which are defined as the probability to find a cluster of certain mass \( N \). While for \( M = 1 \) the increase of the Reynolds number only slightly effects \( p_c(N) \) for larger \( M \) an increase of the Reynolds number results in a \( p_c(N) \) which is much more stretched towards larger cluster sizes \( N \). This agrees with the overall appearance of particle distributions in Fig. 5.1 and the findings for the cluster pair distributions in Fig. 5.2. To characterize the distributions \( p_c(N) \) we used the prediction for \( p_c(N) \) from the kinetic clustering model (KCM) [209] which describes a power law decay for small \( N \) and an exponential tail, i.e.

\[
p_{c}^{\text{KCM}}(N) = C_{\text{KCM}} N^{-3/2} \exp \left( -\frac{N}{N_t} \right),
\]

and fitted it to the simulation results obtaining a parameter \( N_t \) for the stretch of the exponential part and a normalization factor \( C_{\text{KCM}} \). Despite being based on the very simple assumption of randomly distributed particles and not taking any hydrodynamical interaction into account the distributions (5.1) resulting from the KCM model can be very well matched to our data for all combinations of \( M \) and \( Re \). A similar functional form for the distribution of cluster sizes can even be found for spherical and RBC-like particles [212]. The KCM model also makes predictions for \( N_t \) as a function of the volume fraction \( \phi \) which can be very well matched to findings from simulations for low Reynolds numbers [209]. However, it fails to predict \( N_t \) for higher \( Re \) as a consequence of neglecting hydrodynamical interactions. In any case, the functional form of \( p_{c}^{\text{KCM}}(N) \) allows the definition of \( N_t \) as the typical cluster size at which the crossover between the power law and exponential decay takes place.

In Fig. 5.5 the obtained \( N_t \) for all combinations of \( M \) and \( Re \) are plotted over the Stokes number \( St = M Re \). Except for the data for \( M = 1 \) all typical cluster sizes \( N_t \) agree very well with a power law scaling

\[
N_t - N_t^0 \sim St^{1/2}
\]

with an offset of \( N_t^0 \approx 1.7 \). The rather small deviations for large \( N_t \) can be explained by the finite size of the considered system. Although \( N_t \) is always less than half the system size \( L \) the probability for the existence of clusters comparable to \( L \) is non-zero and is enhanced through the application of periodic boundaries. When fitting (5.1) to the data this might lead to an overestimation of the real \( N_t \). Explaining the deviations from the scaling (5.2) for \( M = 1 \) is not so straightforward. It might be related to the leveling of particle and fluid inertia so that particle inertia plays a role subordinate to that of the fluid in particle clustering.

In the simulations also stresses of the fluid as well as solid phase were evaluated to obtain the apparent viscosity \( \nu_{\text{app}} \) of the suspensions for the considered mass density ratios \( M \) and Reynolds numbers \( Re \). For this the procedure described in Sec. 4.4.2 based on the method presented in [115] was applied. In Fig. 5.6 \( \nu_{\text{app}} \) normalized by the viscosity increase due to the volume fraction of \( \phi = 0.4 \) is shown for all \( M \) and \( Re \) considered in this work. In all cases we find shear-thickening but much more pronounced the larger \( M \). When plotted over the Stokes number
Figure 5.4: Cluster size distributions $p_c(N)$ for a set of mass density ratios $M$ and Reynolds numbers $Re$ as obtained by LBM simulations. The lines correspond to (5.1) fitted to the data.
Figure 5.5: The typical cluster size $N_t$ as a function of $M \Re$ for all considered combinations of mass density ratios $M$ and Reynolds number $\Re$. Except for the data for $M = 1$ which lies significantly below the other data for $M > 1$, and the highest typical cluster sizes, all data points collapse very well to a power law with exponent $1/2$ (dashed line). The deviations from it for $M = 1$ could be explained by a too weak particle inertia in comparison to that of the suspending fluid. The upwards shift at large $M \Re$ for $M = 100$ might be a result of finite system size.

$St = M \Re$: all curves fall on top of each other, i.e. for a given $\phi^4$ a master curve for shear-thickening exists which scales with $St$. A small systematic deviation towards higher $\nu_{app}$ can be found for $M = 1$.

When $\nu_{app}$ is plotted as a function of the typical cluster size $N_t$ (Fig. 5.7) we find that all data collapse within statistical errors to one curve which scales as $\nu_{app} \sim N_t^{0.4}$. Together with the other findings of this section this gives rise to the assumption that particle inertia plays the dominant role in shear-thickening. When particles approach each other in sheared flow they come the closer the higher their own inertia compares to that of the fluid. The decelerating effect of the fluid on the particle dynamics scales linearly with the fluid mass density. Once that particles have come very close lubrication forces lead to a strong bond between the particles which can only be loosened by accelerations due to interaction with other heavy particles. The mediating long range action of hydrodynamical forces becomes less important in this process. As a consequence clusters preferably aligned in compression direction of the shear flow are formed from solid particles which serve as a much more direct mechanism to transport momentum through the system.

In the following sections we first will take a step back and study the dynamics of two particles approaching each other in a shear flow in dependence on $\dot{\gamma}$ and $M$. We will then try to formulate a theory on how clusters of more than two particles form and break and will derive a distribution of cluster sizes, again as a function of $\dot{\gamma}$ and $M$. The aim of it is to understand the role of particle clustering in shear-thickening by relating resulting predictions for the typical cluster size $N_t(\dot{\gamma}, M)$ to the apparent viscosity as a function of these parameters.

$^4$Of course, shear-thickening is very sensitive to $\phi$, see Sec. 4.1.
\[ \nu_r = \nu_{\text{app}} / \nu_f / \nu_{KD}(\phi) \]

\[ M = 2.50 \]
\[ M = 10.0 \]
\[ M = 40.0 \]
\[ M = 160.0 \]

\[ \nu_r = \nu_{\text{app}} / \nu_f / \nu_{KD}(\phi) \]

Figure 5.6: Shear-thickening curves for different mass density ratios \( M \). When plotted over the Stokes number \( St = M \text{Re} \) the curves collapse.

## 5.2 2-Particle Collision Model

### 5.2.1 Trajectories

To investigate how close two suspended particles get in a shear flow and to quantify the probability that they are found to fulfill the cluster criterion we set up a simple toy model of two particles approaching each other in a shear flow with shear rate \( \dot{\gamma} \).

Resembling a monodisperse two-particle suspension both particles have the same radius \( R_p \) and same mass \( m \). Initial conditions are such that the velocities of each particle equals the local shear flow velocity, i.e. \( \dot{\mathbf{r}} = (\dot{\gamma}y, 0) \). The initial positions are chosen such that \( \mathbf{r}_1(t = 0) = (24d, d) \) and \( \mathbf{r}_2(t = 0) = -(24d, d) \) creating a symmetric situation with the center of mass at \( \mathbf{r} = (0, 0) \). In dependence on the parameter \( d \) the particles will approach each other and will scatter due to hydrodynamical interactions.

Hydrodynamical interactions between two particles can be approximated by a linear approximation of the lubrication forces that is valid for small particle distances. In 2D the resulting forces on each of two particles that move in respect to each other have the form:

\[ \mathbf{F}_{\text{lub}} = m \ddot{\mathbf{r}}_i = -C \frac{\dot{h}}{h} \mathbf{e}_{ij} \quad \text{with} \quad h = |\mathbf{r}_j - \mathbf{r}_i| - 2R_p = r_{ij} - 2R_p. \]  \hspace{1cm} (5.3)

Additionally, the particles feel a force due to the friction (Stokes, linear) with the homogeneous shear flow background, reading

\[ \mathbf{F}_{\text{fric}} = m \ddot{\mathbf{r}}_i = D(\dot{\mathbf{r}}_{i,y} - \dot{\gamma} \mathbf{r}_{i,y}) \mathbf{e}_x. \]  \hspace{1cm} (5.4)

Mass of a particle and the drag force coefficient form an equivalent to the mass density ratio, i.e. \( M \sim m / D \). With the dissipative action of lubrication alone, the particles would come to a halt if friction with a background at non-zero velocity would not drive them further with the shear flow. In Fig. 5.8 trajectories are shown for several combinations of the parameters \( \dot{\gamma} \) and \( m \). For small values of \( \dot{\gamma} \) and \( m \), particles continue their movement after the interaction with the other particle as if
no interaction had taken place. They arrive at the same \( y = d \) at which they had been released and traveled the same distance in \( x \)-direction resembling the perfect symmetry known for the Stokes limit. If \( \dot{\gamma} \) and/or \( m \) are increased, the symmetry of the trajectories is broken. The two particles come much closer to each other in approaching direction but are repelled so that their distance is larger when they depart. Also, their final separation in \( y \)-dimension can be found to be smaller than that at time \( t = 0 \). If \( \dot{\gamma} \) and \( m \) is increased further, the particles lose so much relative speed due to the lubrication dissipation that the Stokes friction with the fluid background is not strong enough to separate them again. Instead they begin to tumble.

### 5.2.2 Two-Particle Sticking Probability

#### Cluster Condition, Sticking Time

Along some of the trajectories particles are that close that they fulfill the clustering condition \( r_{12} < 2.2R_p \) which is derived from the maximum in the particle distribution function in a hard-sphere suspension. To define a probability that particles might be in such a state at every time step it is checked if this condition is fulfilled, and if so, a *sticking time* is increased by the time step of the integration \( \Delta t \) and normalized by the size of the time frame \( T_\dot{\gamma} \). This procedure is equivalent to an integration definition

\[
\tau = \frac{1}{T_\dot{\gamma}} \int_0^{T_\dot{\gamma}} \begin{cases} 
1, & \text{if } r_{12} < 2.2R \\
0, & \text{else}
\end{cases} dt. \quad (5.5)
\]

We can interpret \( \tau \) as the probability that the two particles will stick for a collision parameter \( d \) in dependence of the parameters \( \dot{\gamma} \) and \( m \).
Figure 5.8: Trajectories of two particles approaching each other in a shear flow with a collision parameter $d$ for a few chosen parameter sets ($\dot{\gamma}, m$).
Number Density of Collision Partners as Function of $d$

We are looking for a distribution $P(d)$ that tells us how many particles would actually follow the trajectories of Sec. 5.2.1, and could be actual collision partner, i.e. no other particle is closer at that time.

As $d$ gets larger the difference in shear velocity between the two particles increases. In an area element $\Delta d \cdot l$, where $l = 48d$ is the stretch in $x$-direction that passes by in the time frame $T_\dot{\gamma}$, we will find

$$N_l = n(\phi, R) \cdot \Delta d \cdot 48d \tag{5.6}$$

particles.

A particle collides with another if their distance is not larger than $2R$. This results in a density of particles on one trajectory that is given by

$$n_{l,\text{coll}} = 2R \cdot n(\phi, R). \tag{5.7}$$

We also could call this quantity line collision density.

The volume of a trajectory is given by its discretization in height, $\Delta d = 0.125$, and the length of an undisturbed trajectory, $48d$, i.e.

$$V_{\text{coll}} = 48d \cdot \Delta d. \tag{5.8}$$

With this contribution we take into account the probability that a particle, able to collide with the other, will actually be found on such a trajectory. For small $d$, the velocity is small, too, and hence the number of particles that touch any point of the trajectory within $T_\dot{\gamma}$ is small.

The probability that $d$ is the free path in the dimension $s = |(48d, d)| = \sqrt{d^2 + (48d)^2} \approx 48d$, meaning that this is the trajectory of an actual collision partner, is an exponential distribution

$$P(d) = 48 \ln \bar{s}_{1D} \exp\left(-\frac{48d - 2R}{\ln \bar{s}_{1D}}\right), \quad d \leq 2R, \tag{5.9}$$

with a typical mean $\bar{s}_{1D}(\phi) = 1/n$ resulting from the particle density $n$ which, in turn, is derivable from the volume ration and radius, $n(\phi, R) \sim \phi/R^2$.

Multiplying all the densities/probabilities we arrive at the final expression for the average number of particles that might be collision partners during the time frame $T_\dot{\gamma}$ in dependence on $d$,

$$N(d) = n(\phi, R) \Delta d \cdot 48d \cdot 48 \ln \bar{s}_{1D} \exp\left(-\frac{48d - 2R}{\ln \bar{s}_{1D}}\right). \tag{5.10}$$

This quantity contains all geometrical information based on $\phi$ and $R$.

In Fig. 5.9 curves for $\tau(d)$ that are measured for the same parameter sets $(\dot{\gamma}, m, \phi = 0.4)$ as in Fig. 5.8 are shown together with three realizations of (5.10).

Sticking Probability

To arrive at an overall probability of sticking in dependence of only the parameters $\dot{\gamma}$, $m$ and $\phi$ we have to integrate the product of $\tau(d)$ and $N(d)$, i.e.

$$P_0 = \int_0^{\infty} \tau(d) \cdot N(d) \, dd \tag{5.11}$$
over all collision parameter \( d \). However, the probability that two particles fulfill the cluster condition is guaranteed to be zero for all \( d > 2R \), so a numerical integration can be truncated at a finite \( d > 2R \) with no additional error.

![Graph showing sticking time \( \tau(d) \) for a range of \( \dot{\gamma} \) and fixed \( m = 5.0 \). Also shown is the number density of collision partners \( N(d) \) for \( \phi = 0.1, 0.25, 0.4 \).](image1)

Figure 5.9: Sticking time \( \tau(d) \) for a range of \( \dot{\gamma} \) and fixed \( m = 5.0 \). Also shown is the number density of collision partners \( N(d) \) for \( \phi = 0.1, 0.25, 0.4 \).

In Fig. 5.10 \( P_0(\dot{\gamma}, m, \phi = 0.4) \) is shown over a wide range of \( (\dot{\gamma}, m) \). \( P_0 \) increases as \( \sim (\dot{\gamma}m)^b \) with \( b \approx 1.25 \) for a wide range but is followed by a decrease and a subsequent increase. The later behavior might be related to the humps in \( \tau(d) \) (see Fig. 5.9) that could be, in turn, an artifact of the finite \( T_\dot{\gamma} \).

![Graph showing \( P_0 \) as a function of \( \dot{\gamma}m \). \( P_0 \) increases as \( \sim (\dot{\gamma}m)^b \) with \( b \approx 1.1 \). The origin of the decrease and the subsequent repeated increase is (probably) a consequence of the finite integration time of the two-particle system where for higher shear rates particles “touch” each other for a second time.](image2)

Figure 5.10: \( P_0 \) as a function of \( \dot{\gamma}m \). \( P_0 \) increases as \( \sim (\dot{\gamma}m)^b \) with \( b \approx 1.1 \). The origin of the decrease and the subsequent repeated increase is (probably) a consequence of the finite integration time of the two-particle system where for higher shear rates particles “touch” each other for a second time.
5.3 Statistical Clustering Model

In a sheared suspension particles might fulfill the cluster condition, forming a 2-particle-cluster. Before such a cluster of size $N_1 = 2$ breaks due to the shear stress it might collide with another cluster of size $N_2$. Defining probabilities of sticking/merging and breaking of clusters depending on the cluster sizes $N_{1,2}$ and the parameters of the sheared system, $\dot{\gamma}$, $m$, $\phi$, in this section we will set up a master equation for the probability $p_c(N)$ that a particle belongs to a cluster of size $N$ based on these probabilities.

5.3.1 Elementary Processes

The change of the probability to find a cluster of size $N$ can be caused by a number of elementary processes.

Merging of smaller clusters

A cluster of size $N$ can be formed from 2 clusters of sizes $N_1$ and $N - N_1$. The probability that such clusters of size $N_1$ and $N - N_1$ exist is $p_{c}(N_1)$ and $p_{c}(N - N_1)$, respectively. The probability that they actually collide and stick is given by $P_{\text{stick}}(N,N_1)$ which depends on the set of system parameters. A cluster of size $N$ can be formed by any possible combinations of $N_1$ and $N - N_1$, that is, the probability that a cluster of size $N$ forms through merging of two other clusters of any possible sizes can be given by

$$P_{\text{merge}} = \sum_{N_1=1}^{N-1} p_{c}(N_1)p_{c}(N - N_1)P_{\text{stick}}(N,N_1).$$ (5.12)

Breaking of larger clusters

The probability that a cluster of size $N_0 > N$ breaks into two parts one of which is of size $N$, thus the result of splitting, is modeled as

$$P_{\text{split}} = \sum_{N_0=N+1}^{\infty} \dot{\gamma}N_0^{1/D-1}p_{c}(N_0).$$ (5.13)

Several assumptions have been made here. First, we assume that the probability that a cluster breaks is proportional to the shear rate $\dot{\gamma}$, which is reasonable because $\dot{\gamma}$ defines a time scale. Second, we assume that the larger a cluster is the more sensible it is to strain. We could combine these two assumptions in another formulation of the argument: the product $\dot{\gamma}R_c$ is proportional to the strain on the cluster and breaking is linear dependent on that. As found in a previous investigation of cluster sizes, clusters in shear flow mostly are elongated in one direction (typically diagonal) leading to a dimensional relation $N = R_c^D$ with a fractal dimension of $D \approx 1.15$. The additional $-1$ in the exponent stems from the combinatoric fact that $N$ combinations of $N_1$ and $N_2 = N - N_1$ exist into which a cluster of size $N$ can break (including the case where $N_1 = 0$ or $N_2 = 0$). Hence the probability that the mother cluster breaks into a certain combination $(N_1,N_2)$ has to be divided by $N$. Every resulting combination is assumed to have the same probability.
Absorption into larger cluster

One of the processes that decrease the number of clusters of size $N$ is the formation of a larger cluster of size $N_0 > N$ by merging the cluster of size $N$ with another cluster of size $N_0 - N$. The probability of such a process is

$$P_{\text{absorb}} = \sum_{N_0 = N+1}^{\infty} p_c(N)p_c(N_0 - N)P_{\text{stick}}(N_0, N). \quad (5.14)$$

Dissolving into smaller clusters

The second process that decreases $p_c(N)$ is the breaking of the cluster of size $N$ into two smaller clusters of sizes $N_1$ and $N_2 = N - N_1$. With the assumption that all results, $(N_1, N_2)$, are equally probable, the probability of such a dissolving process is readily summed up (over all $N_1 < N$) to

$$P_{\text{dissolve}} = \sum_{N_1 = 1}^{N-1} p_c(N)\gamma N^{1/D-1} = p_c(N)(N-1)\gamma N^{1/D-1}. \quad (5.15)$$

Here again, we made the same assumptions on breaking behavior as for $P_{\text{split}}$.

5.3.2 Master Equation for $p_c(N)$

We now can add up positive and negative processes for $p_c(N)$ in a master equation for its dynamics.

$$\partial_t p_c(N) = P_{\text{merge}} + P_{\text{split}} - P_{\text{absorb}} - P_{\text{dissolve}} \quad (5.16)$$

The dynamics can be solved numerically, starting from an initial condition that fulfills the condition $\sum_{N=1}^{\infty} p_c(N) = 1$. We have chosen $p_c(1) = 1$ and $p_c(N > 1) = 0$ resembling a system with randomly distributed particles none of which is in contact with another. Solving the master equation iteratively the distribution of cluster sizes $p_c(N)$ will converge to an equilibrium solution.

Crucial for the dynamics and equilibrium form of $p_c(N)$ is the sticking probability of two clusters, i.e. $P_{\text{stick}}(N = N_1 + N_2, N_1)$. We assume that $P_{\text{stick}}$ takes a functional form

$$P_{\text{stick}} = \beta b N^{1/D} \quad \text{with} \quad \beta = C\gamma m \quad (5.17)$$

according to the findings in Sec. 5.2.2. Despite some remaining unclarity for large $\gamma m$ the exponent $b = 1.25$ has been used here. The assumption $P_{\text{stick}} \sim N^{1/D}$ stems from a spatial scaling argument, treating clusters of radius $R = N^{1/D}$ as particles with the same radius. Here, the result $D = 1.15$ from Sec. 5.1.2 was used.

Results and Discussion

Obtaining an analytical solution for $p_c(N)$ from (5.16) proves difficult. We therefore have numerically integrated the master equation (5.16) in an iterative fashion (Euler method) and found stable equilibrium solutions for $p_c(N)$ after approximately $10^4$ time steps. The distribution has initially been set to $p_c(1) = 1$ and $p_c(N) = 0$ for all $N > 1$. The summations in (5.13) and (5.14) have been truncated at $N = 500$ which has been found not to introduce significant errors when
compared to a reference integration with a upper limit of $N = 10^5$. A least-squares fit routine was coupled to the solver for (5.16) in which $\beta$ was varied so that the resulting $p_c(N)$ matches the cluster size distributions as obtained from the LBM simulations for different mass density ratios $M$ and Reynolds numbers $Re$ as described in Sec. 5.1.2. For the simulation data the KCM prediction (5.1) was used to determine a typical cluster size $N_t$.

The statistical clustering model presented in this section results in $p_c(N)$ that show a very good agreement with the KCM model, which was used as a reference here, as shown in Fig. 5.11. By varying $\beta = C^\gamma m$ in the sticking probability function (5.17) the outcome of the numerical integration can be matched to the results $p_c^{\text{LBM}}(N)$ from the LBM simulations as reported in Sec. 5.1.2.

![Figure 5.11: Equilibrium cluster size distribution $p_c(N)$ for different $\beta$ in (5.17) (data points) matching results of the KCM model (lines) for a number of typical cluster sizes $N_t$. For larger $N_t$ an increasing deviation can be observed.](image)

However, when the typical cluster size $N_t$ obtained from fits of $p_c^{\text{KCM}}(N)$ to $p_c^{\text{LBM}}(N)$ are plotted against the parameters $\beta$ obtained from fits of the statistical clustering model we see a strong deviation from the behavior $N_t \sim \beta^{1/2}$ as could be expected from the scaling $N_t \sim M Re^{1/2}$ found for the LBM results shown in Fig. 5.5. In Fig. 5.12 the relation $N_t(\beta)$ is plotted together with the expected power law. For small $\beta$ the typical cluster size $N_t(\beta)$ does indeed behave as a power law with an exponent $\approx 1/2$. For larger $\beta$ an increasing deviation from this behavior can be observed. For the largest $\beta$ studied here it even increases much faster than exponentially. The clustering model was based on clustering behavior found in a 2-particle problem. It might therefore well be that the made assumptions are reasonable as long as the typical cluster size is small. For larger clusters, and therefore direct interaction of more than two particles, corrections to this behavior will be necessary.
Figure 5.12: Typical cluster sizes $N_t$ as a function of $\beta = C\gamma m$ in (5.17) as obtained from fits of the cluster size distribution from the KCM model (5.1) to the numerical solutions of (5.16). For small $\beta$, $N_t(\beta)$ agrees with the scaling $N_t \sim St^{1/2}$ found in full simulations and shown in Fig. 5.5. An increasing deviation, with eventual divergence, from this behavior can be found for all other values of $\beta$.

5.4 Discussion

In this section we have reported measurements of the microstructure of sheared monodisperse hard-sphere suspensions in shear flow realized by means of the Lattice-Boltzmann method. We concentrated on dense suspensions with a volume fraction $\phi = 0.4$ for which a clearly visible continuous shear-thickening behavior can be expected. Unlike other works on shear-thickening suspensions, in this chapter we focused on the changes resulting from variations of the particle shear Reynolds number $Re = 0.00155 \ldots 1.5529$ for different mass density ratios $M = \rho_s/\rho_f = 1 \ldots 100$. In measurements of a cluster pair distributions for the range of considered $Re$ and $M$ we saw that the expected breaking of the symmetry in the pair distribution function occurs clearly at $Re > 1$ only for density ratios $M > 1$. With the increase of $M$ also the increase of the overall amplitude of the cluster pair distribution function with $Re$ was found much more pronounced. In measurements of cluster distributions this tendency could be recovered and quantified by the finding that the typical cluster size scales with the Stokes number $St = MRe$ as $N_t \sim St^{1/2}$. This Stokes number scaling we also could report for shear-thickening measurements for a set of $M$. These finding give rise to the assumption that particle inertia plays a dominant role in shear-thickening suspensions, a fact has not been discussed extensively in the literature.

From measurements of the distributions of the radius of gyration linked to the number of particles in clusters forming in sheared suspensions we find a fractal dimension of $D = 1.15$ for all considered combinations of $M$ and $Re$ suggesting that the geometrical properties of the clusters do not change when these system parameters are varied. Together with the asymmetry found in the pair distribution
function this fact characterizes particle clusters as preferably of chain-like shape (instead of compact globule-like clustering) and as oriented preferably in the direction of compression in the shear flow. This supports the assumption that particle clusters offer an effective mechanism through which momentum can be transferred through the system. When the apparent viscosity $\nu_{\text{app}}$ of the suspension is plotted as a function of the typical cluster size $N_t$ we indeed find that all data points collapse to one curve regardless the parameters $Re$ and $M$ which scales as $\nu_{\text{app}} \sim N_t^{0.4}$.

Aiming at investigating the process of cluster formation in this chapter we modeled the approach of two particles in a shear flow from which an integrated time was derived in which the particles fulfill the cluster criterion. The probability to find a two particles clustering in dense suspensions that could be derived from that was also found to obey Stokes number scaling but with slightly larger exponent $b = 1.25$. When this behavior of the probability for clustering particles is then used in a statistical clustering model which models probabilities of mechanisms of cluster formation and dissolution as functions of the shear rate and under usage of the spatial scaling assumption that particle cluster with a radius of gyration $R \sim N^{1/D}$ behave as single particles when they collide, a cluster size distribution function can be obtained in dependence on the shear rate and mass density ratio.

The results agree very well with the predictions of the KCM model which serves as a reference and a way to extract a typical cluster size $N_t$ from the distributions. The KCM model, however, is only valid for $Re \ll 1$. The statistical clustering model is successful in reproducing the form of $p_c(N)$ also found in the LBM simulations by varying the shear rate $\dot{\gamma}$ and mass density ratio $M$. For small $\beta$ it does reproduce the scaling $N_t \sim St^{1/2}$ observed through fully-resolved LBM simulations. For larger $\beta$ increasing deviations are present, a clear hint that higher order particle-particle interactions have to be taken into account. The scaling assumptions deduced from the two-particle model, and the assumption that cluster-cluster collisions can be modeled as particle-particle collisions might be valid for small $N_t$ but corrections are clearly necessary for larger $N_t$. 
Chapter 6

Summary and Conclusions

The research presented in this thesis was motivated by the question how nature processes information. In the introduction this question was specified to the problem of the multi-scale character of natural processes now reading: How is information processed between structures on distinct scales?

In chapter 2 we have introduced a classification of multi-scale problems based on the position of a process on the scale separation map (SSM) and the relative position of another process which is coupled to it. We have also identified coupling templates based on this relative position on the SSM and whether the two processes share the same domain or only an interface between each other. This chapter also takes the step towards and works out an execution model which was implemented in a software framework based on these findings that allows to quickly set up multi-scale simulations. Applying the scale separation concept to two examples we could show that the actual area on the SSM can be very much reduced without causing significant errors but gaining a tremendous computational speedup compared to a full-scale simulation. We have therefore successfully identified the relevant aspects of the information processing in the sub-processes and the coupling between them.

In chapter 3 we could show that the CxA idea can be successfully applied in the challenging field of modeling cardiovascular diseases. Using the CxA tools like the SSM we identified the relevant processes involved in in-stent restenosis and the coupling between them. The multi-scale model of in-stent restenosis has been implemented employing the MUSCLE coupling library and the results, presented and compared to measurements in real systems, show that the approach is valid. Furthermore, preliminary results of a three-dimensional model are presented where the same scale separation and coupling concepts were used. Also, a model for the formation of thrombus could be successfully coupled in as an initial condition to the subsequent processes.

The multi-scale idea was also applied to the macroscopic flow of two-dimensional hard-sphere suspensions. In chapter 4 we discussed the single-scale processes involved in the formation of flow and particle concentration profiles. We identified the relevant scales of the sub-processes and made use of their separation in several aspects of the modeling. The micro-macro type of scale separation of the three sub-models for macroscopic non-Newtonian flow, macroscopic advection and diffusion of the particle concentration, and the micro-model involving the dynamics of both phases in a fully resolved fashion, and the fact that they all share the
same spatial domain resulted in a multi-scale modeling approach in the sense of the Heterogeneous Multi-scale Method (HMM). We did not make use of the MUSCLE coupling environment as its current version does not support farming as a parallelization method which was intended in the initial planning of the modeling project. Instead, the HMM coupling was realized by implementing the micro-model in a subroutine fashion. However, we successfully analyzed the multi-scale model by means of the tools the CxA theory provides, discussed possible ways to implement the system as a CxA, and could estimate a very large speed-up obtained by the scale separation. The concept of CxA’s were a great help categorizing the multi-scale problem, thinking about information flow between the scales and modeling approaches. Although the models introduced in this work could have been formulated independently each with the terminology of the field of application, the CxA concept served as a common language for multi-scale problems.

In the implementation of the multi-scale suspension model we made use of scale separations also in other aspects. We successfully exploited the temporal scale separation of the macroscopic flow and diffusion to accelerate the macroscopic dynamics and reduce the run-time of the simulations. Temporal scale-splitting was also applied to the dynamics of the fluid and particle phase in the micro-model to increase the stability of the numerical integration.

The concept of Chas, even if supported by the MUSCLE library, is not a specific method. It is a framework for designing multi-scale simulation methods. For any particular problem, there is usually a considerable amount of work to develop models and realize the coupling. For the CxA model of in-stent restenosis we developed a model for thrombus formation which particularly meets the aspects of its application in the ISR model.

During the work on the HMM suspension not only a Lagrangian particle advection-diffusion method has been implemented, and the mapping of relevant quantities between Lagrangian and Eulerian representation of the particle concentration has been discussed. We have also developed and implement a database to reduce the actually necessary runs of the computationally costly micro-model (which can be considered a smart conduit in the language of CxA’s). Furthermore, the implementation of the submodels in a for the coupling optimized fashion have led to improvements of the methods used for it. To implement the micro-model by means of the Lattice-Boltzmann method for suspension flows based on the original idea of Ladd, Lees-Edwards boundary conditions (LEbc) have been developed to realize the shear flow of the micro-system in a quasi-infinite domain which corresponds well with the idea of HMM micro-models representing a sample of the micro-scale dynamics at a certain point of the macroscopic domain. Lees-Edwards boundary conditions could be shown to be able to remove the side-effects associated with the walls in a Couette type of shear flow. It could also be shown that using LEbc’s the shear-thickening behavior of hard-sphere suspensions is prolonged in comparison to Couette-flow results. When implementing and validating LEbc’s we found that the momentum exchange algorithm (MEA) used in commonly applied LBM suspension methods suffers from violation of Galileian invariance. With the help of an asymptotic analysis we could derive an appropriate correction to MEA to restore Galileian invariance. Also, a non-equilibrium refill method was found to produce more realistic particle dynamics than existing methods in situations where particles in close contact move together over the LBM lattice.
In chapter 5 the improved LB methods for suspension flow have also been successfully applied to study the behavior of the typical spatial scale in hard-sphere suspensions over a large range of particle shear Reynolds numbers $Re_p$. At higher shear rates hydrodynamically interacting particles are found to form clusters accompanied by the phenomenon of shear-thickening. We studied a cluster pair distribution function which displays anisotropy for higher $Re_p$ and geometrical aspects of the clusters. Cluster size distributions have been measured and discussed in a comparison with the predictions from the Kinetic Clustering Model. A scaling of the apparent viscosity with the typical cluster size can be found supporting the assumption that clustering is the prominent process leading to shear-thickening.

In the last chapter we also propose a statistical model for the development of cluster size distributions in dependence on shear rate $\dot{\gamma}$, $\phi$, and solid-fluid mass density ratio. Based on the findings from a two-particle collision model the formation and breaking of particle clusters are modeled and relax to cluster size distributions that are in very good agreement with earlier findings. The scaling of the typical cluster size, however, could not be derived successfully from this model. Still, we think it provides a valuable contribution to the study of the typical spatial scale in sheared systems of suspended particles.

The work presented in this thesis spans from a rather general motivation of the research interest in the multi-scale aspects of nature to a very detailed study of the formation of spatio-temporal scales in one specific system. This top-down manner of the organization of the manuscript has not necessarily been intended. It rather reflects the time-line of the author’s research over the last four years as a consequence of working out support for the observations described in the first part of the thesis. The author believes that the Complex Automata approach to the multi-scale modeling of complex systems offers a great tool to characterize sub-systems, their scales, and their coupling up to the point of an execution model. The experience in applying this concept to the multi-scale systems studied in this work, however, tells us that it is essential to understand the dynamics of the involved subprocesses and their numerical models in sufficient detail. Otherwise, any direct or indirect effect of oversimplification of the description might lead to a drastic loss of validity of the coupled model system. A detailed (error) analysis of the submodels and their (simplified) coupling is not always possible, only for very simple systems such as the reaction-diffusion coupling presented in Sec. 2.3. However, the author would like to recommend such analyses for every multi-scale system modeled. Many works on multi-scale modeling in the field of biomedicine lack a validation of the involved submodels and numerical methods although their results might have implications for the treatment of real world diseases. Validation of multi-scale models remains a big issue, also for the ISR model presented in this thesis. In order to better understand the multi-scale nature of suspension flow, the author has tried to provide detailed studies of the involved processes and the applied numerical methods.
Bibliography


Na een introductie die onderzoek naar de meerschalige aard van complexe systemen motiveert, wordt in het eerste deel van het proefschrift het abstracte concept Complex Automaten (CxA) voorgesteld als een nieuwe aanpak om zulke systemen te modelleren. In dit deel worden meerschalige koppelingen geïdentificeerd en worden passende koppelingssjablonen geopperd op basis van een klassificatie aan de hand van een schaalscheidingsafbeelding en submodel-uitvoeringslussen. Ten einde ook andere onderzoekers een eenvoudig middel te bieden om meerschalige simulaties op te zetten is er een software framework op basis van het concept CxA ontwikkeld. De resulterende meerschalige koppelingssbibliotheek en -omgeving (MUSCLE) implementeert het executiemodel van een CxA en maakt daarmee op een eenvoudige manier gedistribueerd berekenen van meerschalige modellen mogelijk. De auteur is er zeker van dat het concept CxA van grote waarde is bij het modelleren van complexe systemen waarin verschillende processen op verschillende ruimte-tijd schalen een rol spelen. In feite bewijst dit proefschrift de toepasbaarheid van dit concept door simulaties van twee verschillende meerschalige systemen te presenteren die zijn gecreëerd in deze geest.

Als eerste voorbeeld van een meerschalig model wordt er een model van het biomedische fenomeen in-stent restenose (ISR) beschreven. Het model is met behulp van MUSCLE ontwikkeld om de invloed van bloedstroom op dit proces te bestuderen. De betrokken subprocessen zijn geïdentificeerd op de schaalscheiding samen met hun type koppelingen. De gecreëerde submodellen worden met voldoende detail beschreven. Er wordt enigszins uitgeweid over de vorming van trombus in de vroege stadia van ISR en hier wordt een nieuw versimpeld trombus model voor voorgesteld. Aan het eind van dit hoofdstuk worden de resultaten van de executies van de 2D en 3D ISR CxA’s gepresenteerd en besproken en wordt er aangetoond dat het mogelijk is om de karakteristieken van in-stent restenose te reproduceren. Het model zal binnenkort verder worden ontwikkeld en de auteur is overtuigd dat het model leidt tot een uitbreiding van inzicht in het oorspronkelijke probleem.

Het tweede meerschalige systeem dat in dit proefschrift wordt beschreven is een niet-Browniaanse harde-bollensuspensie. De macroscopische emergente reologie wordt bepaald door de details van de wisselwerking tussen deeltjes en vloeistof en van deeltjes onderling. Het is een voorbeeld van systemen waarbij de typische ruimtelijke schaal op het niveau van de deeltjes veel kleiner is dan die van de macroscopische proporties. De schalen kunnen gesplitst worden met behulp van de hierarchische meerschalige methode (HMM) waarin de suspensie op de macroscchaal wordt behandeld als een niet-Newtoniaanse vloeistof terwijl lokale eigenschappen
van macroscopische velden de invoer zijn voor een volledige suspensie-simulatie. Op beide niveaus wordt de raster-Boltzmann methode gebruikt als model voor de vloeistoffase. Het naar boven en naar beneden schalen van waarden met betrekking tot viscositeit en difusiteit wordt besproken. De resultaten worden gepresenteerd in vergelijking met observaties van experimentele resultaten en analytische oplossingen voor reeds bekende niet-Newtoniaanse modellen van de stroming van suspensies. In dit hoofdstuk wordt aangetoond dat het concept CxA helpt bij het identificeren van de relevante schalen en met de formulering naar een executiemodel. Met het gestelde multischaal model wordt de berekening meerdere ordes van grootte versneld, niet alleen door de betrokken schalen te splitsen maar ook doordat er zo intrinsiek gebruik gemaakt wordt van de rotationele en Galileische symmetrien. Dit is nog uitgebreid met een op maat gemaakte implementatie van een database met een interpolatie- en extrapolatiefunctionaliteit.

Uiteraard lossen concepten niet alles op dus blijven er meerdere uitdagingen verborgen in de technische details van de gebruikte methodie bij de implementatie van een model. In dit proefschrift worden twee grote verbeteringen gegeven voor het simuleren van volledig opgeloste suspensies met behulp van de LBM. Ten eerste zijn de Lees-Edwards grensvoorwaarden (LEbc) ontwikkeld om de bijwerkingen van muren in een Couette-type schuifstroom op de microstructuur tegen te gaan. Het kan worden aangetoond dat door LEbc’s te gebruiken het afschuifverdikkende gedrag van harde-bolSuspensies wordt verlengd in vergelijking met resultaten van Couette-stroming. Ten tweede wordt er een correctie gegeven op het momentumuitwisselingsalgoritme dat wordt gebruikt in regelmatig toegepaste LBM suspensiemethodes, die de Galileische invariantie van de deeltjesdynamica herstelt.

Om de gestelde methodes te valideren alsmede ten gevolge van de auteur’s wetenschappelijke interesse in suspensie-reologie, worden in het laatste hoofdstuk de bovengenoemde verbeterde methodes gebruikt om het gedrag van de ruimtelijke schaal in harde-bolSuspensies toenemende schuiving te onderzoeken, voor verschillende verhoudingen van vaste en vloeibare inhoud. Gewoonlijk vormen zich bij een hoge afschuifsnelheid clusters van hydrodynamisch interagerende deeltjes, wat afschuifverdikkend veroorzaakt. Clustergrootte-distributies zijn gemeten en vergeleken met experimentele resultaten en theoretische modellen. Er kan een schaal-verband worden gevonden tussen de gebruikte viscositeit en de typische clustergrootte, wat de aannames bevestigt dat clustering de belangrijkste oorzaak is van afschuifverdikkening. Gegeven deze bevindingen wordt er in dit hoofdstuk een statistisch model voorgesteld voor de vorming van clustergrootte-distributies, afhankelijk van de afschuifsnelheid, de inhoudsverhouding vast/vloeibaar en de dichtheidsverhouding vast/vloeibaar. Op basis van de dynamiek van een botsingsmodel voor twee deeltjes wordt de vorming en het opbreken van deeltjesclusters gemanageerd. Het model leidt tot clustergrootte-distributies die goed in overeenstemming zijn met andere metingen.
Acknowledgements

My special thanks goes to Alfons Hoekstra for being a very helpful supervisor and for being confident in me when my scientific curiosity led me into the very details of the physics of suspensions which, besides the research carried out in the COAST project, now constitutes a larger part of this thesis. At the same time, I appreciate very much that he taught me to be pragmatic in the process and formulation of my findings in order to be able to communicate them.

I am grateful to Peter Sloot for giving me the opportunity to work in this highly interesting field of research. His quest for an answer on how nature processes information matches very well my own research motivation.

I wish to thank all the ex-COASTERs for being helpful colleagues over the time of the project and beyond. Special thanks in this regard goes to Alfonso Caiazzo.

I thank all the members of the SCS group for discussions on research as well as non-research topics. In particular, I thank Syed Murtaza and Gokhan Ertaylan in this respect. Special thanks goes to my new close colleague Joris Borgedorff for translating the summary of this thesis into Dutch.

My parents deserve my eternal gratitude for their unconditional confidence in me and their support in many ways. And, last but surely not least, I am in deep dept of gratitude to Ulli for keeping me sane during the last years.
Publications

published:


abstracts at conferences:


accepted for publication:


submitted:

• E. Lorenz, and Alfons G. Hoekstra. *Heterogeneous Multi-scale Simulations of Suspension Flow*, submitted to SIAM Multiscale Modeling and Simulation.

in preparation:

• E. Lorenz, and A. G. Hoekstra, *Particle Clustering in Shear-Thickening of Hard-Sphere Suspensions*.

• E. Lorenz, and A. G. Hoekstra, *Temporal Scale Splitting of the Dynamics of Solid and Fluid Phase in Lattice-Boltzmann Simulations of Sheared Suspensions*.

• E. Lorenz, and A. G. Hoekstra, *The Use of a Database in HMM Simulations of Suspension Flow*. 