

Hocheffiziente Präzisionsalgorithmen zur Modellierung und Analyse granularer poröser Medien

Von der Fakultät Mathematik und Physik der Universität Stuttgart
zur Erlangung der Würde eines Doktors der
Naturwissenschaften (Dr. rer. nat.) genehmigte Abhandlung

Vorgelegt von

Thomas Zauner

aus München

Hauptberichter: Prof. Dr. Dr. Rudolf Hilfer

Mitberichter: Prof. Dr. Hans-Rainer Trebin

Tag der mündlichen Prüfung: 26.06.2020

Institut für Computerphysik

Universität Stuttgart

2020

Inhaltsverzeichnis

Abbildungsverzeichnis	8
Tabellenverzeichnis	12
Symbolverzeichnis	13
1 Abstract	18
2 Kurzfassung	25
3 Motivation, Aufgabenstellung und Zusammenfassung	26
4 Begriffsbildung und elementare Größen für poröse Medien	32
4.1 Repräsentatives Elementarvolumen (REV)	33
4.2 Charakteristische Funktion	34
4.3 Porosität	34
4.4 Permeabilität	35
4.5 Spezifische Oberfläche, spezifische mittlere Krümmung und totale Krümmung	36
4.6 Lokale Porositätstheorie (LPT)	37
4.6.1 Auflösungsabhängige Beispielanwendung der LPT	40
4.7 Modellierung und Diskretisierung granularer poröser Medien	44
4.7.1 Kontinuums-Rekonstruktion eines dreidimensionalen Kontinuumsmodells . . .	44
4.7.2 Diskretisierung des Kontinuumsmodells zu binären 3D-Datensätzen	45
5 Diskrete-Elemente-Methode zur Generierung dichter Kugelpackungen	46
5.1 Definitionen, Initialisierung und Abbruchbedingungen	47
5.2 Kraftfeld und Kraftberechnung	48

5.3	Zeitintegration	51
5.4	Steuerungsoptionen	52
5.4.1	Zeitabhängige Interaktions- und Dämpfungsparameter	52
5.4.2	Stochastische Kräfte	52
5.4.3	Adaptive Kugelgrößen	53
5.4.4	Einfrieren von Freiheitsgraden	53
5.5	Effiziente parallele Implementation	54
5.5.1	Parallelisierung durch Gebietszerlegung	55
5.5.2	Effektive Kommunikation im Rahmen des MPI-Standards mit nur sechs P2P- Operationen	55
5.5.3	Nachbarschaftsstrukturen, Interaktionslisten und Mehrschrittverfahren	59
5.5.4	Komplexität und Leistungsfähigkeit der Implementation	59
5.6	Anwendungsbeispiele	61
5.6.1	Hochkorrelierte dichte polydisperse Kugelpackung	61
5.6.1.1	Simulationsparameter und Ablauf	62
5.6.1.2	Diskretisierung der Kugelpackung	65
5.6.1.3	Geometrische Charakterisierung	65
5.6.1.4	Ergebnisse aus der lokalen Porositätstheorie (LPT)	66
5.6.1.5	Permeabilitätsberechnung durch Gitter-Boltzmann-Simulationen	67
5.6.2	Kugelpackung mit rauen Porenwänden	69
5.6.2.1	Simulationsparameter und Protokolle	70
5.6.3	Große monodisperse Kugelpackung	73
5.7	Fazit	73
6	Permeabilitätsberechnung durch Gitter-Boltzmann-Simulationen	74
6.1	Boltzmann-Gleichung und Gitter-Boltzmann-Simulationen	75
6.1.1	Stoffeigenschaften bei der Anwendung in porösen Medien	79
6.1.2	Externe Kräfte bei Strömungssimulationen in porösen Medien	80
6.1.3	Behandlung der Phasengrenzen bei Simulationen in porösen Medien	81
6.1.4	Aufbau der Strömungssimulation	82
6.1.5	Kalibrierung freier Parameter und Fehlerquellen	84
6.2	Permeabilitätsberechnung eines Fontainebleau-Sandsteins	94
6.3	Fazit	96

7	Auflösungsabhängige Berechnungen aus digitalen 3D-Modellen	97
7.1	Diskretisierung des gesamten Kontinuumsmodells	98
7.2	Berechnung der Porosität aus der Grauwertverteilung	102
7.3	Berechnung der spezifischen inneren Oberfläche aus der Grauwertverteilung	106
7.4	Auflösungsabhängige LPT- und Permeabilitätsberechnung	110
7.5	Fazit	112
8	Modellierung und Berechnungen für Bentheimer Sandstein	113
8.1	Analyse und Aufbereitung der experimentellen Daten	113
8.2	Modellierung des Bentheimer Sandsteins	118
8.2.1	Generierung neuer Quarztypen	118
8.2.2	Anisotrope 'lokale' Dekoration der Kugelpackung	119
8.2.3	Final verwendete Parameter bei der Modellierung	120
8.3	Analyse und Vergleich mit den experimentellen Daten	122
8.4	Fazit	124
9	Anhang	125
9.1	Veranschaulichung eines Simulationsschrittes im DEM-Algorithmus	125
9.2	Technische Dokumentation des DEM-Programms 'PackDEM'	128
9.2.1	Einführung in das Message-Parsing-Interface und dessen Verwendung in 'Pack-DEM'	128
9.2.2	Voraussetzungen und Kompilierung des ausführbaren Programms 'PackDEM'	129
9.2.3	Programmaufruf und Beschreibung der Konfigurationsdatei	129
	Literaturverzeichnis	136

Abbildungsverzeichnis

1.1	DEM Potential	20
1.2	D3Q19 Diskretisierung	22
4.1	Fontainebleau-Rekonstruktion als 3D Bild und Querschnitte	40
4.2	Fontainebleau-Modell: Kernprobenentnahme	40
4.3	Fontainebleau-Modell: Teilgebiete und deren Querschnitte	41
4.4	Fontainebleau-Modell: Auflösungsabhängige lokale Porositätsergebnisse	42
4.5	Fontainebleau-Modell: Auflösungsabhängige Perkolationswahrscheinlichkeiten	43
4.6	Fontainebleau-Modell: Totaler Anteil der perkolierenden Zellen	43
5.1	DEM: Wechselwirkungspotentiale	50
5.2	DEM: MPI-Gebietszerlegung und Zellendefinition	56
5.3	DEM: MPI-Kommunikationsstrategie	58
5.4	DEM: Polydisperse Kugelpackung 3D-Darstellungen	61
5.5	DEM: Polydisperse Kugelpackung 2D-Querschnitte	62
5.6	DEM: Polydisperse Kugelpackung, Größenhistogramme und Überlappungen	64
5.7	DEM: Polydisperse Kugelpackung $\mu^{FP}(\phi, L, a)$ und $\mu^{PP}(\phi, L, a)$	66
5.8	DEM: Polydisperse Kugelpackung $\lambda^{FP}(\phi, L, a)$ und $\lambda^{PP}(\phi, L, a)$	67
5.9	DEM: Polydisperse Kugelpackung $p(L)$	68
5.10	DEM: Raue Porenwände, 2D-Querschnitte	69
5.11	DEM: Raue Porenwände, Radiusverteilung der primären Matrix	72
5.12	DEM: Raue Porenwände, Überlappung der primären Kugeln	72
6.1	LB: Diskretisierung der Geschwindigkeiten	76
6.2	LB: Aufbau der Simulation	82
6.3	LB: Kalibrierung am Kanal. Relative Fehler 2D	87

6.4	LB: Kalibrierung am Kanal. Relative Fehler in log-linear	88
6.5	LB: Kalibrierung am Kanal. Relative Fehler bei verschiedenen Querschnitten	89
6.6	LB: Kalibrierung am Kanal. Relative Fehler bei verschiedenen Relaxationszeiten	89
6.7	LB: Kalibrierung am Kanal. Mittlerer relativer Fehler in der Permeabilität	91
6.8	LB: Kalibrierung am Kanal. Druckverlauf und dessen Mittelungen	92
6.9	LB: Permeabilitäten aus verschiedenen LB-Methoden und Relaxationszeiten	96
7.1	HiRes: Fontainebleau-3D-MegaMol-Darstellung	98
7.2	HiRes: Fontainebleau-3D-MegaMol-Darstellung. Kleiner Ausschnitt	99
7.3	HiRes: Fontainebleau-Modell. Querschnitte bei verschiedenen Auflösungen	101
7.4	HiRes: Porosität $\phi(g_{th}, a)$ als Funktion des Schwellwerts g_{th} und der Auflösung	103
7.5	HiRes: Extrapolation von $\phi(g_{th}, a)$ bei verschiedenen Schwellwerten g_{th}	104
7.6	HiRes: Lineare Interpolation der Porosität	105
7.7	HiRes: Auflösungsabhängige LPT Berechnungen.	110
7.8	HiRes: Auflösungsabhängige Permeabilitäten aus Gitter-Boltzmann-Simulationen	111
8.1	Bentheimer Sandstein: Original 2D μCT -Querschnitt	114
8.2	Bentheimer Sandstein: Segmentierter zurechtgeschnittener μCT -Querschnitt	115
8.3	Bentheimer Sandstein: Lokale Porosität $\mu(\phi, L)$	116
8.4	Bentheimer Sandstein: Lokale Perkolationswahrscheinlichkeiten $\lambda(\phi, L)$	117
8.5	Bentheimer Sandstein: Gesamtanteil perkolierender Messzellen $p(L)$	117
8.6	Bentheimer Sandstein: 2D 10x SEM-Datensatz	118
8.7	Bentheimer Sandstein: 2D 100x SEM-Datensatz	119
8.8	Bentheimer Sandstein: Exemplarische Korntypen	119
8.9	Bentheimer Sandstein: Optischer Vergleich Modell und experimentelle Daten	121
8.10	Bentheimer Sandstein: Vergleich der lokalen Porosität $\mu(\phi, L)$	122
8.11	Bentheimer Sandstein: Vergleich lokaler Perkolationswahrscheinlichkeiten $\lambda(\phi, L)$	123
8.12	Bentheimer Sandstein: Vergleich des Gesamtanteils perkolierender Messzellen $p(L)$	123
9.1	DEM: Rand-Replika-Schritt	126
9.2	DEM: Kraft-Schritt	126
9.3	DEM: Geschwindigkeits-Schritt	126
9.4	DEM: Verschiebe-Schritt	127
9.5	DEM: Replika-Rand-Schritt	127

9.6 DEM: Replika-Leeren 127

Tabellenverzeichnis

4.1	LPT: Richtungsindex α	38
5.1	DEM: Polydisperse Kugelpackung $\phi, S_V, K_V(\mathbb{P}), T_V(\mathbb{P})$	65
5.2	DEM: Polydisperse Kugelpackung $\kappa(a)$	68
6.1	LB: Kalibrierung am Kanal. Konvergenzverhalten	86
6.2	LB: Kalibrierung am Kanal. Mittlere relative Fehler	90
6.3	LB: Kalibrierung am Kanal. Berechnete Permeabilitäten	93
6.4	LB: Berechnete Fontainebleau Permeabilitäten	95
7.1	HiRes: Fontainebleau-Modell. Übersicht der vorhandenen Diskretisierungen	100
7.2	HiRes: Interpolationsparameter bei der Porositätsbestimmung	104
7.3	HiRes: Obere und untere Schranken der spezifischen inneren Oberfläche	106
7.4	HiRes: Größen und Diskretisierungen der Proben	111
8.1	Bentheimer Sandstein: Verteilungen der definierten Volumenteile	115
8.2	Bentheimer Sandstein: Charakteristische Größen	116
8.3	Bentheimer Sandstein: Parameter der Modellierung	120
8.4	Bentheimer Sandstein: Vergleiche der berechneten Größen	122
8.5	Bentheimer Sandstein: LB-Simulation für Modell und Experiment	124

Symbolverzeichnis

Symbole	Beschreibungen
S	Poröse Probe
G	Phasen
M	Matrix (feste Phase)
P	Porenraum (fluide Phase)
K	Lokale Messzelle
$V_i(a)$	Voxel mit Seitenlänge a
$g_i(a)$	Grauwert mit $0 \leq g_i(a) \leq 216$
g_{th}	Grauwertschwelle
$1 \leq s \leq s^{\max}$	Zeitschritt im DEM Algorithmus (maximal s^{\max})
L	Seitenlänge eines Kubus
χ	Charakteristische Funktion
$\mathbf{x} \in \mathbb{R}^3$	Ortsvektor
a	Auflösung
ϕ_G, ϕ	Porosität der Phase G , des Porenraums
κ	Permeabilität
μ	Dynamische Viskosität
$\mathbf{v}(\mathbf{x})$	Geschwindigkeitsfeld eines Fluides
$p(\mathbf{x})$	Druckfeld eines Fluides
S_V	Spezifische Oberfläche
K_V	Spezifisches Integral der mittleren Krümmung
T_V	Spezifisches Integral der totalen Krümmung
$\mu(\phi, L)$	Lokale Porositätsverteilung
$\lambda(\phi, L)$	Lokale Perkolationswahrscheinlichkeit

Symbol	Beschreibung
O_i	i – te Kugel
\mathbf{x}_i^s	Ort der i – ten Kugel im Zeitschritt s
\mathbf{v}_i^s	Geschwindigkeit der i – ten Kugel im Zeitschritt s
r_i^s	Radius der i – ten Kugel im Zeitschritt s
σ_{ij}^s	Überlappung zweier Kugeln (O_i, O_j) zum Zeitschritt s
$\sigma_{t_i}^{\max}$	Maximale Überlappung für den Kugeltyp t_i
$t_i \in \{1, \dots, N_g\}$	Kugeltyp i von insgesamt N_g Typen
\mathbf{f}_i^s	Kraft auf die i – ten Kugel im Zeitschritt s
n_{ij}^s	Größe der Kraft zwischen dem Kugel paar (O_i, O_j)
$k_{t_i}^s$	Wechselwirkungskonstante für den Kugeltyp t_i
k_{ij}^s	Wechselwirkungskonstante für das Kugel paar (O_i, O_j)
dx_{cut}	Abschneidedistanz im DEM Algorithmus
$\Xi_{t_i}^s$	Dissipationskonstante des Kugeltyps t_i im Zeitschritt s
W_i^s	Wechselwirkungsmenge für die Kugel i im Zeitschritt s
$l_k^d, k = 1, 2, 3$	Seitenlängen der MPI Domänen
$l_k^c, k = 1, 2, 3$	Seitenlängen der Zellen innerhalb der MPI Domänen
S_{k,t_i}^0, S_{k,t_i}^1	Start- und Endschrirte für die Wechselwirkungsparameter

1 Abstract

Porous media are widely used materials that play an important role in a variety of problems. The investigation into their material and transport characteristics is a current research branch. Multi-scale porous rock is an omnipresent material in the recovery of oil and natural gas and many ecological problems [19, 2]. Filters are artificial porous materials that are widely used components in industrial plants [25]. Understanding their limitations and durability is key for planning efficient maintenance cycles, optimizing production and moving towards a more 'green' industry. Another rapidly growing field of application that relies on porous materials are fuel cells, wherein gas diffusion layers [23] are used. As consumer products, paper [44] and textiles are everyday porous materials. Also fascinating is their increasing medical application, for example the use of porous synthetic materials as replacements for bone structures in the medical implantology or as synthetic tissues. In all these applications it is important to have realistic digital models and adequate simulation methods to determine material properties and predict physical processes such as mechanical and thermal stresses and fluid flow through these materials.

This work focuses on natural granular porous rock as a material. It presents a new highly efficient, parallel algorithm to create realistic digital continuum models, compares simulation algorithms and discusses computation strategies that calculate media properties. The model creation process starts with experimental data samples from imaging techniques such as scanning electron microscopy (SEM), X-ray tomography or optical methods either from three-dimensional samples or two-dimensional cross sections [8]. From these data sets a realistic three-dimensional model will be reconstructed. That means that some predefined characteristics, calculated from a specific digital model statistically coincide with those of the real medium. It is important to determine which characteristic quantities are meaningful and should be used. Furthermore these quantities and the way of determining them must be representative in the digital model. Representative means that reconstructed digital models that use the same initial source parameters should give similar measured results for very large, macroscopic, sample sizes. For smaller, microscopic, sample sizes the

results should be comparable and be statistically representative. The actual size, i.e. the volume, of the digital sample at which a measured quantity becomes representative for this media is called the representative element volume (REV) of this quantity. Obviously small digital models are more viable in terms of available computational power, nevertheless the calculated properties need to be representative.

This work is divided into two parts. The first part has the objective to reconstruct a digital model of a granular porous medium from experimental data. The second part discusses methods and strategies that calculate physical properties of the medium. A central module of the digital reconstruction process used in this work is the newly developed algorithm for generating dense sphere packings for a given set of initial parameters. The previously used algorithm proved to be insufficient for very large, very dense, highly polydisperse or highly correlated packings. Many typical granular porous rocks exhibit these features of very different grain sizes, locally strongly correlated regions, large void pore spaces, narrow pore throats that might be filled with small grains, large grains whose surface is overgrown with smaller grains or complex structures accumulated by a geological process. Because of the polydisperse nature of these rocks and the various topological regions, macroscopic models are required to calculate predictive characteristic quantities and transport properties. The main module in the second part is a fluid simulation code from whose results the permeability can be calculated. After applying special boundary conditions to a discretized three dimensional data set of the porous media model the velocity field and the pressure field are calculated. Part of this work was the clarification whether this kind of simulation technique is suitable. The calibration and comparison of this new fluid simulation method to experimental data and other simulation methods was carried out with much effort.

Most calculations discussed herein use discretized data sets. These are obtained by resolving the digital continuum model at a certain resolution. Therefore all results, like geometrical characteristics or transport properties are essentially resolution dependent. If these quantities are well defined and the discrete simulation techniques are properly implemented they will asymptotically converge with increasing resolution. To save computational resources and still be able to get predictable numerical calculations, extrapolation methods and error estimates are necessary. Another objective in this work were extensive resolution-dependent analyses of the numerical calculations used.

The new construction algorithm uses the discrete element method (DEM), to generate a sphere packaging as a starting point. Special emphasis was given to have a very efficient parallelized implementation to generate large models. The goal was to be able to generate large multiscale

packings. Sphere sizes spanning two decades and packings of more than 100 million spheres were targeted. As mentioned before the available techniques were not able to meet these requests.

The approach of the discrete elements method (DEM) is to start from an initial configuration, i.e. sphere positions and sphere sizes, and then to change the position according to a set of rules until the packing complies to given constraints such as maximal overlap between each pair of spheres. The initial population includes spheres of different types that have varying size distributions. It is important to note that for this reconstruction step only the final configuration is of interest. It is not a simulation of the geological genesis of a porous rock. The rules used in this simulation are inspired by physical forces, but are justified by their capability of effectively producing a final configuration. With that said, the forces used in this work and the term 'potential' are to be understood only in the figurative sense. They have been used to help the physical intuition. The DEM method is an iterative algorithm. In every iteration the force acting on each sphere is calculated from its position, velocity, size or other special parameters. This step is called the 'force-step'. Then the spheres are 'moved' to new positions in a step called 'propagation-step'. The forces \mathbf{f}_i^s acting on the sphere O_i are defined as the sum of pairwise interaction forces

$$\mathbf{f}_i^s = \sum_{O_j \in W_i^s} \mathbf{f}_{ij}(dx_{ij}^{s-1}, r_i^{s-1}, r_j^{s-1}; k_{ij}^s) \quad (1.0.1)$$

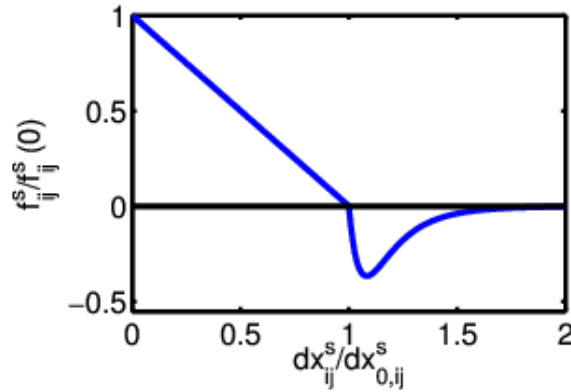


Figure 1.1: Image of the normalized force $f_{ij}^s / f_{ij}^s(0)$ acting between two spheres O_i and O_j at a time step s as a function of their normalized distance $dx_{ij}^s / dx_{0,ij}^s$.

with partner list W_i^s , radii r_i^{s-1} and r_j^{s-1} and interaction constants k_{ij}^s . For the propagation a

modified velocity-Störmer-Verlet scheme [47] is used. The sphere positions are \mathbf{x}_i^s , and the velocities are

$$\mathbf{v}_i^s = \frac{(C_{\Xi}^-)^s}{(C_{\Xi}^+)^s} \mathbf{v}_i^{s-1} \Delta t + \frac{1}{(C_{\Xi}^+)^s} (\mathbf{f}_i^{s-1} + \mathbf{f}_i^s) \Delta t / (2m_i)$$

with a discretized time step Δt and where the $(C_{\Xi}^+)^s = 1 + \frac{\Delta t}{2} \Xi_{t_i}^s$, $(C_{\Xi}^-)^s = 1 - \frac{\Delta t}{2} \Xi_{g_i}^s$ are defined by some dissipation constants $\Xi_{t_i}^s$.

The pairwise interaction forces are modeled by combining a harmonic and a generalized Lennard-Jones potential. By introducing situation-dependent parameters whose behavior is determined at the start of the simulation, complex protocols can be devised. Some examples for parameters that were changed during the simulation are the interaction and dissipation coefficients. Also the sizes of the spheres can be changed during a simulation according to the spheres condition. A locking mechanism was implemented that fixed certain spheres to their positions if certain conditions were fulfilled. This locking mechanism can be (de)-activated during the simulation.

To be able to handle several 100 millions of spheres the simulation code had to use a method of parallelization. As a parallelization paradigm the domain decomposition was chosen. In this paradigm the spatial simulation volume is decomposed into disjoint sub volumes. These sub volumes are called domains. Each computing unit, which could be a single CPU or a core in a multi core CPU or software thread (software process), only handles one such domain. This parallelization paradigm creates the need for communication between the computing units, because the information is fragmented and distributed and surface effects arise between adjacent domains. The implementation in this work uses the MPI ('Message Passing Interface') standard, as a means of inter-process communication (IPC). An 'edge-replica-cell' approach is used to treat the surface interaction between neighboring domains. Furthermore a special point-to-point (P2P) communication strategy was implemented to minimized the number and data size of necessary communications.

The final sphere packing is transformed into a continuous model of granular rock by replacing each sphere with a random polyhedron chosen from a given set and oriented randomly, as described in [10]. This digital model is then used for further calculations of geometrical characterizations and transport parameters.

The permeability is calculated from the Darcy equation

$$\kappa^{\text{LB}} = -\eta \frac{\langle \mathbf{v}_3^{\text{LB}} \rangle_{\mathcal{S}}}{\langle (\nabla p)_3^{\text{LB}} \rangle_{\mathcal{S}}}, \quad (1.0.2)$$

Where $\langle v_3^{\text{LB}} \rangle_{\mathcal{S}}$ is the flow velocity averaged over the entire sample, the mean pressure gradient in z-direction is $\langle (\nabla p)_3^{\text{LB}} \rangle_{\mathcal{S}}$ and the dynamic viscosity of the fluid is η . The velocity and pressure field within the sample is calculated using a lattice Boltzmann solver.

The Boltzmann equation is

$$\frac{\partial}{\partial t} n(\mathbf{x}, \mathbf{c}, t) + \mathbf{c} \cdot \nabla n(\mathbf{x}, \mathbf{c}, t) = \Omega(n(\mathbf{x}, \mathbf{c}, t)) \quad (1.0.3)$$

with the location $\mathbf{x} \in \mathbb{R}^3$, the velocity vector $\mathbf{c} \in \mathbb{R}^3$, the time $t \in \mathbb{R}$ and the collision operator $\Omega(n(\mathbf{x}, \mathbf{c}, t))$. It describes the time evolution of the one-particle probability density $n(\mathbf{x}, \mathbf{c}, t)$. For the collision operator a Single-Relaxation-Time (SRT) and a Multi-Relaxation-Time (MRT) implementation were compared. In the lattice Boltzmann simulations implemented in this work the volume and velocity space is discretized using the D3Q19 scheme [65].

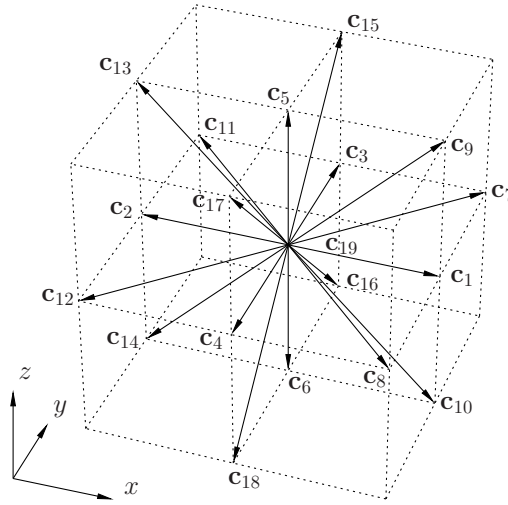


Figure 1.2: D3Q19 discretization, for details see [65].

The discretized probabilities are

$$n_i(\boldsymbol{\ell}, t) = \int_{\mathbb{W}(\boldsymbol{\ell})} \int_{\mathbb{B}(i)} n(\mathbf{x}, \mathbf{c}, t) d\mathbf{c} d\mathbf{x} \quad (1.0.4)$$

with $\mathbb{W}(\boldsymbol{\ell}) \subset \mathbb{R}^3$ the finite volume associated with the lattice point $\boldsymbol{\ell}$ and $\mathbb{B}(i) \subset \mathbb{R}^3$ a cone in velocity space with tip at the origin and direction of \mathbf{c}_i . From these the pressure and velocity field can be calculated by

$$\rho(\boldsymbol{\ell}, t) = \rho^\circ \sum_{i=1}^{N_v} n_i(\boldsymbol{\ell}, t), \quad (1.0.5)$$

$$\mathbf{v}(\boldsymbol{\ell}, t) = \sum_{i=1}^{N_v} n_i(\boldsymbol{\ell}, t) \mathbf{c}_i / \sum_{i=1}^{N_v} n_i(\boldsymbol{\ell}, t), \quad (1.0.6)$$

where ρ° is the density. The pressure is defined as

$$p(\boldsymbol{\ell}, t) = c_s^2 \rho(\boldsymbol{\ell}, t), \quad (1.0.7)$$

with

$$c_s = \frac{1}{\sqrt{3}} \left(\frac{\Delta x}{\Delta t} \right). \quad (1.0.8)$$

the speed of sound, see [73, 66].

The applicability of the lattice Boltzmann simulation to calculate permeabilities for realistic porous media was tested for several applications. The result of this investigation is that lattice Boltzmann simulations are well suited for calculating permeabilities. The two different implementations of the collision operator are both usefull. The simpler LB-BGK method must be carefully calibrated, for which a typical pore diameter must be known. If this is not possible the less sensitive, but somewhat slower, LB-MRT method should be used. The resolution used also plays a decisive role in the choice of the appropriate Boltzmann method. A promising approach is a resolution-dependent analysis with subsequent extrapolation.

The developed discrete element algorithm (DEM) with its various protocol possibilities and its parallel implementation works highly efficiently and has been successfully used in various applications. Packings with up to 100 million spheres and with highly geometric complexity can be created. For example a realistic model for a Bentheimer sandstone was created. This application was especially

challenging because of the particularly elongated and widely differing grain shapes. In another application an extremely polydisperse model was created, in which very small objects filled the pore space between the large grains and thereby inhibited the flow through these pore throats. This application made extensive use of the protocol capabilities of the implementation to create the model.

2 Kurzfassung

Diese Arbeit befasst sich mit dem Studium poröser Medien insbesondere im Anwendungsgebiet von mehrskaligen granularen porösen Medien, wie sie in natürlicher Form in geologischen Materialien wie Sandsteinen vorkommen. Konkret geht es um die Ausarbeitung und Untersuchung von theoretischen Methoden zur Generierung und Analyse von digitalen Modellen für diese Art poröser Medien. Das Fachgebiet ist physikalisch eingebettet in den größeren Komplex der stochastischen Geometrien und deren Beschreibung. Zur Modellerstellung wird ein speziell auf das Problem adaptierter Diskrete-Elemente-Algorithmus implementiert und auf ausgewählte Beispiele angewandt. Die Implementation ist äußerst effizient und kann Modelle generieren, die aus mehreren 100 Millionen Objekten bestehen. Bei der Analyse dieser Modelle werden charakteristische Größen wie die Porosität, die spezifische Oberfläche, die spezifische mittlere Krümmung und die totale Krümmung auflösungsabhängig berechnet. Ebenfalls werden die im Rahmen der Lokalen-Porosität-Theorie (LPT) definierte lokale Verteilung der Porosität, die lokale Perkolationswahrscheinlichkeit und der daraus berechnete Gesamtanteil perkolierender Zellen für verschiedene Modelle und Auflösungen betrachtet. Die Transportgröße Permeabilität der digitalen Modelle wird über das Darcy Gesetz berechnet. Dazu werden verschiedene Ausprägungen von Gitter-Boltzmann-Simulationen, wie die 'Single-Relaxation-Time'- und die 'Multi-Relaxation-Time'-Methode verglichen. Beide Simulationsmethoden sind zur Berechnung der Permeabilität anwendbar, haben ihre Berechtigung und sollten in einem iterativem Vorgehen mit zunehmend besserer Auflösung verwendet werden. Die im Rahmen dieser Dissertation erarbeiteten Präzisionsalgorithmen und Vorgehensweisen eröffnen die Möglichkeit, mehrskalige Modelle granularer poröser Medien in Laborgröße zu generieren und deren geometrische Charakteristika und Transporteigenschaften dann prädiktiv und effizient zu berechnen.

3 Motivation, Aufgabenstellung und Zusammenfassung

Poröse Medien sind weitverbreitete Materialien, die in einer Vielzahl von Problemstellungen eine wichtige Rolle spielen. Die Erweiterung des Wissens über die Beschaffenheit, Material- und Transporteigenschaften ist ein aktueller Forschungszweig. In der Erdöl- bzw. Erdgasförderung [20, 21] und bei ökologischen Problemstellungen [19, 2] sind mehrskalige poröse Gesteinsschichten allgegenwärtig. In Industrieanlagen sind Filter [25] viel verwendete Bauteile, bei denen eine möglichst genaue Kenntnis ihrer Einsatzgrenzen und Lebensdauern für die Planung der Wartungszyklen finanziell erheblich ist. In Kraftstoffzellen werden poröse Materialien z. B. als verwendet [23]. Als Endprodukte sind Papier [44] und Textilien alltägliche poröse Materialien. In der medizinischen Implantologie werden poröse, synthetische Materialien als Ersatz für Knochen verwendet [55]. Bei all diesen Anwendungen ist es wichtig, realistische digitale Modelle und leistungsstarke Simulationsmethoden zur Verfügung zu haben, um die Materialeigenschaften und das Verhalten bei physikalischen Vorgängen wie mechanischer und thermischer Belastung und bei Strömungsvorgängen exakt vorhersagen zu können. Diese Arbeit fokussiert sich in ihrer Anwendung auf granulares poröses Gestein.

Das verwendete Kontinuumsmodell, der Rekonstruktionsprozess mit Hilfe eines Packungsalgorithmus für Kugeln sowie die numerischen Berechnungen von charakteristischen Größen sind für viele Arten von porösen Medien verwendbar. Ausgangspunkt dieser Arbeit sind experimentelle bildgebende Verfahren wie Rasterelektronenmikroskopie (scanning electron microscope 'SEM'), Röntgentomographie oder optische Methoden, die zweidimensionale Schnitte oder dreidimensionale Daten liefern [8]. Aus diesen experimentellen Daten soll ein repräsentatives dreidimensionales Kontinuumsmodell rekonstruiert werden, das bedeutet ein Modell, dessen charakteristische Eigenschaften statistisch mit denen des realen Mediums übereinstimmen. Es ist daher erst festzulegen, welche charakteristischen Größen dazu verwendet werden sollen. Repräsentativ bedeutet in diesem Zusammenhang, dass verschiedene Rekonstruktionen mit denselben Ausgangs- und Zielparametern durchaus auf der mikroskopischen

Skala unterschieden werden, aber statistisch bei endlichen Systemgrößen vergleichbare Messwerte liefern.

Andere weit verbreitete Ansätze in der Modellierung von porösem Material, welche ebenfalls experimentelle 2D- bzw. 3D-Daten nützen, umgehen den Arbeitsschritt zum Erzeugen eines 3D-Kontinuumsmodells. Sie erzeugen ein a-priori diskretisiertes Modell. Die in [78] genutzte „Simulated Annealing“-Methode startet mit einer stochastischen binären Voxelveilung für Poren bzw. Matrix, die so gewählt ist, dass sie der gemessenen Porosität entspricht. Dann werden einzelne Voxel nach einem definierten Kriterium getauscht, bis eine für das Modell berechnete Zielfunktion, hier die Zwei-Punkt-Korrelationsfunktion, gut genug der aus den experimentellen Daten berechneten Funktion entspricht. Diese Modelle haben jedoch den Nachteil, dass sie ohne neue experimentelle Daten nicht „verfeinert“ werden können. Auflösungsabhängige Analysen sind also sehr viel aufwendiger.

Ein zentrales Modul des in dieser Arbeit verwendeten digitalen Rekonstruktionsprozesses ist ein Algorithmus zur Generierung von Kugelpackungen nach vorgegebenen Parametern. Die bisher genutzten Algorithmen erwiesen sich als unzureichend bei sehr großen, sehr dichten, stark polydispersen oder hochkorrelierten Packungen. Genau solche Modelle mit stark unterschiedlich großen Körnern, lokal stark unterschiedlichen Regionen wie großen Porenkammern, engen Porenzwischenräumen und Verbindungen, die mit kleinen Körnern befüllt sind, oder großen Körnern, an deren Oberfläche sich kleinere Körner angesammelt haben, sind jedoch typisch für viele poröse Gesteine. Genau wegen des erheblichen Größenunterschieds der verschiedenen Komponenten und der verschiedenen topologischen Regionen sind makroskopische Modelle notwendig, um statistisch sinnvolle charakteristische Größen zu berechnen und um in Simulationen Transportgrößen vorherzusagen.

Ein weiteres Kernmodul bei der Berechnung eines wichtigen Transportparameters, der Permeabilität, ist die Fluidsimulation. Dabei wird ein Datensatz des diskretisierten porösen Modells unter vorgegebenen Randbedingungen von einem bekannten Fluid durchströmt und es wird das Feld der Strömungsgeschwindigkeit und des Druckes berechnet. Im Rahmen dieser Arbeit soll eine neue, sehr effiziente Berechnungsmethode angewandt und validiert werden. Die Motivation zu einer neuen Berechnungsmethode ergibt sich aus dem Wunsch, größere und geometrisch komplexere Modelle verwenden zu wollen. Auch hier war die bisherige verwendete Simulationsmethode an ihre Grenzen gestoßen.

Alle Berechnungen der Materialeigenschaften und die Fluidsimulation verwenden diskretisierte Datensätze, die aus dem Kontinuumsmodell gewonnen werden, indem es zuerst mit einer bestimmten Auflösung abgerastert wird und diese primären Daten dann aufgearbeitet werden. Es stellt sich da-

her offensichtlich die Frage, bei welcher oder welchen Auflösungen die Berechnungen durchgeführt werden sollen und wie die Daten analysiert werden sollen, um verlässliche Aussagen treffen zu können. Eine weitere Aufgabenstellung in dieser Arbeit waren daher umfangreiche auflösungsabhängige Berechnungen und deren Analyse. In der Arbeit [56] wird ein komplexer Arbeitsablauf mit verschiedenen theoretischen Konzepten und Simulationstechniken vorgestellt und gezeigt, wie wichtig gut kalibrierte Methoden sind. Besonders werden in [56] auch Hochskalierungsmethoden angewandt und ihre Anwendung für mehrskalige poröse Gesteine kritisch diskutiert.

Konkrete Zielformulierung

Die Zielsetzung lässt sich wie folgend auflisten:

- Im Modellierungs- und Rekonstruktionsprozess soll ein neuer Ansatz zur Packungsgenerierung ausgearbeitet und in einer sehr effizienten parallelisierten Implementation umgesetzt werden. In Anwendungen soll dieses neue Programmpaket evaluiert und mit bekannten Ergebnissen verglichen werden. Die Anforderungen an den Packungsalgorithmus für Kugeln sind, dass er effizient Packungen aus mehreren hundert Millionen Kugeln erstellen kann. Weiterhin sollen verschiedene Kugeltypen vorgesehen sein, deren statistische Wahrscheinlichkeiten vorgegebenen Verteilungsfunktionen folgen. Es sollen Bereiche mit verschiedenen Häufigkeitsverteilungen und Kugeltypen konfiguriert werden können. Zwischen den verschiedenen Kugeltypen soll ein Überdeckungsbereich definiert werden können, der festlegt, um wie viel sich jeweils zwei Kugeln überdecken dürfen.
- Zur Berechnung der Permeabilität der diskretisierten Proben soll ein neuer Strömungslöser verwendet werden. Dieser Gitter-Boltzmann-Löser muss für die Anwendung angepasst und kalibriert werden. Die Ergebnisse müssen mit analytischen Lösungen, bisherigen Simulationsergebnissen und mit Literaturwerten verglichen werden.
- Für die im Rahmen dieser Anwendungen typischen charakteristischen Größen sollen umfangreiche auflösungsabhängige Berechnungen durchgeführt werden, um dadurch zu klären, was geeignete Auflösungen für die verschiedene Größen und Modelle sind.

Struktur der Arbeit

Die Arbeit ist in fünf Kapitel aufgeteilt. Vorangestellt ist diese Einleitung, die die Motivation und Aufgabenstellung der Arbeit darlegt. Den Abschluss bildet ein Anhang mit einigen detaillierten technischen Informationen über den Ablauf und die Bedienung der im Rahmen der Arbeit erstellten Programme und Algorithmen.

Das zweite Kapitel gibt eine Definition eines porösen Mediums und erläutert die grundlegenden Begriffe für dessen Beschreibung. Die in dieser Arbeit verwendeten Größen und die Art ihrer Berechnungen werden aufgezeigt. Weiterhin werden die Arbeitsschritte, die bei der Modellierung und anschließenden Untersuchung verwendet werden, erläutert. Auch werden Beispiele für typische Ergebnisse vorgestellt.

Im dritten Kapitel wird der Diskrete-Elemente-Algorithmus zur Erstellung dichter Kugelpackungen, unter Berücksichtigung vorgegebener Randbedingungen, vorgestellt. Dieser Algorithmus ist das Kernstück der vorliegenden Arbeit, da alle weitergehenden Untersuchungen die von ihm erstellten Modelle verwenden bzw. weiterverarbeiten. Der Grundgedanke dieses Algorithmus ist seit längerem in anderen Bereichen der Physik zur Modellierung und Simulation wohlbekannt. Seine Anpassung und Verwendung bei der Modellierung von mehrskaligen granularen porösen Medien sind jedoch neu. In diesem Kapitel werden auch zwei erste Anwendungen vorgestellt.

Die Berechnung der Permeabilität für diese Datenmodelle poröser Medien durch transiente Gitter-Boltzmann-Simulationen wird ausführlich in Kapitel vier diskutiert. Die Gültigkeit dieser physikalischen Gleichung, ihre spezielle numerische Diskretisierung und die komplexen Simulationsbedingungen sind keinesfalls offensichtlich. Die Simulationsmethode wurde an die Anforderungen in porösen Medien angepasst und es werden verschiedene Randbedingungen diskutiert.

Eine Modellierung im Rahmen der stationären Stokes- oder Reynoldsgleichung für dünne Spalte und kleine Poren und eine Finite-Differenzen- oder Finite-Volumen-Diskretisierung erscheint zunächst viel naheliegender. Dennoch hat die Gitter-Boltzmann-Simulation auch erhebliche Vorteile wie zum Beispiel ihre rechentechnische Effizienz, bei der Diskretisierung und Lösung, und eine einfache Parallelisierungsstrategie.

Die in dieser Arbeit verwendeten Berechnungen zur Charakterisierung eines porösen Mediums und zur Bestimmung effektiver Transportkoeffizienten nutzen Datenmodelle, die mit einer bestimmten Auflösung diskretisiert wurden. Im fünften Abschnitt wird ein wohlbekanntes Modell eines Fontainebleau-

Sandsteins bei sehr vielen Auflösungen diskretisiert und die erzeugten Datensätze werden für diese Berechnungen verwendet. Die in diesem Rahmen entstandenen sehr umfangreichen Datensätze wurden der Allgemeinheit zugänglich gemacht. Sie sind, nach Wissen des Autors, im Bereich der granularen porösen Medien die umfänglichsten Datensätze bezüglich des Auflösungsbereiches und der Probengröße, die es öffentlich gibt.

Im sechsten Kapitel wird eine Anwendung mit einem Diskrete-Elemente-Algorithmus diskutiert, um ein realistisches Modell eines Bentheimer Sandsteins zu erstellen. Verschiedene Modellparameter wurden variiert und die Ergebnisse mit den zur Verfügung stehenden Berechnungsmethoden der charakteristischen Größen untersucht.

Fazit der Arbeit

Der entwickelte Diskrete-Elemente-Algorithmus mit seinen diversen Protokollmöglichkeiten und seiner parallelen Implementierung arbeitet hoch effizient und wurde in einigen Anwendungen erfolgreich eingesetzt. Es wurden Packungen mit bis zu 100 Millionen Kugeln und großer geometrischer Komplexität erstellt. Des Weiteren wurde damit ein realistisches Modell für einen Bentheimer Sandstein erstellt. Diese Anwendung war wegen der besonders gestreckten und stark unterschiedlichen Kornformen anspruchsvoll in der Modellierung. Auch hier erwies sich der Algorithmus als tragfähig. In einem der Beispiele wurde ein extrem polydisperses Modell erzeugt, in dem sehr kleine Objekte die Porenzwischenräume zwischen sehr großen Objekten auffüllen und dadurch die Strömung durch das Medium stark beeinflussen. Dieses hochkorrelierte Modell nutzt extensiv die Protokollfunktionen der Implementation, um in mehreren Simulationsschritten das Modell zu erzeugen.

Die Verwendung von Gitter-Boltzmann-Simulationen zur Berechnung von Permeabilitäten kann als zuverlässig und effektiv angesehen werden. Es wurden zwei verschiedene theoretische Ansätze diskutiert und verglichen. Beide Ansätze haben ihre Berechtigung. Die einfachere LB-BGK-Methode muss sorgfältig geeicht werden, wozu typische Porendurchmesser bekannt sein müssen. Ist dies nicht möglich, zum Beispiel bei stark unterschiedlichen Strömungsquerschnitten entlang der Hauptflüsse, sollte auf die unempfindlichere, aber etwas langsamere, LB-MRT-Methode ausgewichen werden. Auch die verwendete Auflösung spielt eine entscheidende Rolle bei der Wahl der passenden Gitter-Boltzmann-Methode. Als vielversprechendes Vorgehen hat sich eine auflösungsabhängige Analyse mit anschließender Extrapolation herausgestellt.

Bei den auflösungsabhängigen Berechnungen ist es schwierig, im Vorhinein eine minimale Probengröße im Sinne des repräsentativen Elementarvolumens und eine maximal große Auflösung zu wählen, um die Datensätze möglichst klein zu halten, wenn bei einem Modell nur die Korngrößenverteilung, die Korntypenverteilung und eventuell eine geschätzte Porosität bekannt sind. Als probates Mittel erwies es sich, auf eine iterative Vorgehensweise zurückzugreifen. Durch geeignete Regression oder Extrapolation wird dabei die Auflösung erhöht, bis die gewünschte Fehlertoleranz oder die maximal handhabbare Datengröße erreicht ist.

4 Begriffsbildung und elementare Größen für poröse Medien

Zum Anfang dieses Abschnittes seien einige Definitionen für poröse Medien aus der Literatur vorangestellt. Diese Übersicht soll veranschaulichen, wie breitgefächert die Definitionen in ihrer technischen und mathematischen Ausprägung sind.

„In the most general sense, a porous material is a solid containing holes. However, a hollow metal cylinder, for example, is not usually classed as a porous material; consequently, a more precise specification of the term porous material is required. For the purpose of this study, a solid containing holes or voids, either connected or non-connected, dispersed within it in either a regular or random manner will be classed as a porous material provided that such holes occur relatively frequently within the solid.“
(„Flow of Fluids Through Porous Materials“, R.E. Collins, 1961)

In diesem Klassiker aus dem Jahr 1961 wird weitgehend der Intuition des Lesers vertraut, sich einen anschaulichen Satz an Beispielen vorzustellen. Eine deutlich präzisere topologische Definition ist in folgender Arbeit zu finden:

„An n -component porous medium in d dimensions is defined as a compact and singly connected subset \mathbb{S} of \mathbb{R}^d , which contains n closed subsets $\mathbb{P}_i \subset \mathbb{S}$ that

$$\mathbb{S} = \mathbb{P}_1 \cup \dots \cup \mathbb{P}_n \text{ and } 0 = V_d(\partial\mathbb{P}_i)$$

for all $1 \leq i \leq n$. The set \mathbb{S} is called the sample space ... The subsets \mathbb{P}_i ... represent n different phases or components, such as different materials or phases containing rock. The symbol V_d denotes the d -dimensional volume ... The sets $\partial\mathbb{P}_i$ are the phase boundaries separating the components.“

(„Transport and relaxation phenomena in porous media“, R. Hilfer, 1996)

Eine ähnliche Definition, die den Begriff poröses Medium ebenfalls aus dem Begriff des heterogenen Mediums herleitet und auch Bezug auf Größenverhältnisse nimmt, ist in folgender Arbeit zu finden:

„Eine n -phasige, heterogene Probe \mathbb{S} sei definiert als der Zusammenschluss von n Gebieten \mathbb{G}_i unterschiedlichen Materials bzw. desselben Materials in verschiedenen Aggregatzuständen (Torquato 1991). $\mathbb{S} = \mathbb{G}_1 \cup \dots \cup \mathbb{G}_n$. Die Größenskala der verschiedenen Phasen \mathbb{G}_i , die Mikroskala, sei dabei groß verglichen zur Größenskala der Moleküle und Atome. Die einzelnen Phasen können daher als Kontinua betrachtet werden. Insbesondere spielt bei einer späteren Beschreibung von Transportvorgängen das Eindringen von Atomen in molekulare Zwischenräume keine Rolle. Die porösen Medien bilden eine Untermenge der heterogenen Medien, bei der sich die Phasen \mathbb{G}_i aufteilen lassen in feste und fluide Phasen (Hilfer 2000, Adler 1992)“. ("Geometrische Modellierung und Transporteigenschaften poröser Medien", C. Manwart, PhD Thesis, 2001)

Der Autor dieser Arbeit schließt sich in seinem intuitiven Verständnis der letzten Definition an. Die Anwendungsfälle aller in dieser Arbeit betrachteten porösen Medien sind granulare poröse Gesteine oder deren rekonstruierte Modelle wie z. B. stochastische Kugelpackungen und die daraus weiterentwickelten Kornmodelle. Folgend werden einige elementare Begriffe und Definitionen präsentiert, die in dieser Arbeit verwendet werden.

4.1 Repräsentatives Elementarvolumen (REV)

Das repräsentative Elementarvolumen, kurz REV, einer stochastischen Geometrie ist ein Maß für das kleinstmögliche Gebiet, bei dem eine charakteristische Größe noch sinnvoll definiert werden kann. Bei porösen Medien sind solche charakteristischen Größen zum Beispiel die Porosität, die Pfadlängen, die spezifische innere Oberfläche oder die Permeabilität. Diese Größen sind dann sinnvoll definiert, wenn sich bei einer Vergrößerung des untersuchten Gebiets der Mittelwert nur noch unmaßgeblich ändert. Statistische Messungen bzw. Berechnungen von Größen an Proben, die mindestens dem REV für diese Größe entsprechen, führen dann zu repräsentativen Aussagen für das Medium. In [38] wird das REV-Problem detailliert diskutiert und gelöst. Welche Größen für gegebene Typen von porösen Medien weitestgehend „charakteristisch“ sind und mit welchen Methoden diese Größen berechnet werden sollten, ist seit langem ein anhaltendes Forschungsgebiet [64, 45, 56].

4.2 Charakteristische Funktion

Eine n -phasige poröse Probe $\mathbb{S} \subset \mathbb{R}^3$ sei definiert als Vereinigung $\mathbb{S} = \cup_{i=\{1,\dots,n\}} \mathbb{G}_i$ von räumlichen Gebieten $\mathbb{G}_i \subset \mathbb{R}^3$. Diese Gebiete werden von Medien mit verschiedenen physikalischen Aggregatzuständen, unterschiedlichen Materialien oder unterschiedlichen Strukturen eingenommen. Die einzelnen Medien werden Phasen genannt. Es wird davon ausgegangen, dass sich die Phasen in zwei Klassen kategorisieren lassen. Die festen Phasen die das Matrixgebiet \mathbb{M} einnehmen. Sowie fluide Phasen die das Gebiet \mathbb{P} , den Porenraum, einnehmen. Es gilt dann $\mathbb{S} = \mathbb{P} \cup \mathbb{M}$. Die charakteristische Funktion $\chi_{\mathbb{G}} : \mathbb{R}^3 \rightarrow \{0, 1\}$ sei definiert als

$$\chi_{\mathbb{G}}(\mathbf{x}) = \begin{cases} 1 & \text{wenn } \mathbf{x} \in \mathbb{G} \\ 0 & \text{wenn } \mathbf{x} \notin \mathbb{G} \end{cases} \quad (4.2.1)$$

worin $\mathbf{x} \in \mathbb{S}$ und $\mathbb{G} \in \{\mathbb{M}, \mathbb{P}\}$ also entweder die Matrix oder der Porenraum ist. In dieser Arbeit beschreibt die Matrix die feste Struktur eines porösen Gesteins. Die Daten, die diese Phase beschreiben, können aus computergenerierten Modellen oder experimentellen bildgebenden Verfahren stammen. In vielen Algorithmen wird die charakteristische Funktion $\chi_{\mathbb{G}}(\mathbf{x})$ in einer kubisch diskretisierten Form $\chi_{\mathbb{G}}^{ijk} = \chi_{\mathbb{G}}(\mathbf{x}_{ijk})$ verwendet mit

$$\mathbf{x}_{ijk} = i \cdot a\mathbf{e}_1 + j \cdot a\mathbf{e}_2 + k \cdot a\mathbf{e}_3 \quad (4.2.2)$$

und der Auflösung a sowie den kartesischen Einheitsvektoren $\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3$.

4.3 Porosität

Die Porosität $\phi_{\mathbb{P}}$ ist der Volumenanteil des Porenraums und falls nicht speziell gekennzeichnet wird die Abkürzung $\phi = \phi_{\mathbb{P}}$ verwendet. Der Volumenanteil eines Gebietes $\mathbb{G} \in \{\mathbb{M}, \mathbb{P}\}$ am Gesamtvolumen $V(\mathbb{S})$ des Systems ist

$$\phi_{\mathbb{G}} = \langle \chi_{\mathbb{G}} \rangle_{\mathbb{S}}.$$

Die Mittelwertbildung $\langle f \rangle_{\mathbb{S}}$ einer Funktion $f(\mathbf{x})$ mit $\mathbf{x} \in \mathbb{S}$ bezieht sich im Allgemeinen auf das Mittel über alle möglichen Konfigurationen des stochastischen Ensembles, wird in dieser Arbeit aber

unter Annahme von Ergodizität als räumliches Mittel verstanden

$$\langle f \rangle_{\mathbb{S}} = \frac{1}{V(\mathbb{S})} \int_{\mathbb{S}} f(\mathbf{x}) d\mathbf{x}.$$

Die Porosität ϕ kann aus dem Kontinuumsmodell approximiert werden, als statistischer Mittelwert über zufällige Positionen $\mathbf{x}_i \in \mathbb{S}$ als

$$\phi = \frac{1}{N_c} \sum_{i=1, \dots, N_c} \chi_{\mathbb{P}}(\mathbf{x}_i)$$

wobei die Auswertung von $\chi_{\mathbb{P}}(\mathbf{x}_i)$ im Kontinuum rechenintensiv sein kann. Alternativ kann ein kubisch diskretisiertes und binär segmentiertes Modell, siehe Abschnitt 4.7.2, mit Gitterpunkten \mathbf{x}_{ijk} wie in Gleichung 4.2.2 verwendet werden. Es gilt dann

$$\phi = \frac{1}{N} \sum_{i,j,k} \chi_{\mathbb{P}}(\mathbf{x}_{ijk}) = \frac{N_{\mathbb{P}}}{N}$$

also das Verhältnis der Anzahl von Porenvoxel $N_{\mathbb{P}}$ zur Gesamtanzahl der Voxel N . Die so berechnete Porosität ist dann, im Gegensatz zur Porosität aus dem Kontinuumsmodell, abhängig von der Auflösung und Segmentierung. Eine weitere Methode, die in dieser Arbeit vielfach benutzt wird, ist die Methode nach Ohser und Mücklich [61], in der die Porosität mittels eines rekonstruierten Grauerthistogramms abgeschätzt wird. Auch ist es möglich, direkt aus einem Grauerthistogramm, das aus einem diskretisierten, aber nicht segmentierten Datensatz gewonnen wurde, eine sehr gute Abschätzung der Porosität zu gewinnen. Genaueres zu diesem sehr effizienten Verfahren, das im Rahmen dieser Arbeit entwickelt wurde, findet sich in Abschnitt 7.2. Ein Medium mit hoher Porosität bedeutet nicht zwingend, dass es für Fluide leicht durchgängig ist, da die Porosität nicht die Konnektivität der Porenräume beschreibt. Es gibt eine Definition einer effektiven Porosität, die sich nur auf das Porenvolumen der verbundenen Poren beschränkt. Methoden zur experimentellen Bestimmung der Porosität sind in [14] beschrieben.

4.4 Permeabilität

Die Permeabilität $\kappa [\text{m}^2]$ eines porösen Mediums beschreibt dessen Fähigkeit für den Volumentransport. Die Materialeigenschaft des porösen Mediums ergibt sich aus der Konnektivität der Poren und

deren Größen. Daher spielt das Konzept des REV bei ihrer Interpretation eine wichtige Rolle. Erst ab einer sinnvollen Probengröße ist sie eine charakteristische Größe für das Probenmaterial. Wird ein poröses Medium der Länge L [m] und des Querschnittes A [m²] horizontal aufgrund einer Differenz im anliegenden makroskopischen Druck ΔP [Pa] von einem inkompressiblen Fluid mit der Viskosität μ [Pa · s] durchflossen und ergibt sich eine Volumenflussrate von Q [m³/s], so ist die Permeabilität

$$\kappa = \frac{Q \cdot \mu \cdot L}{A \cdot \Delta P}.$$

Die Permeabilität kann auch über eine mikroskopisch statistische Formulierung der Darcy-Gleichung

$$\langle \mathbf{v} \rangle_{\mathbb{S}} = -\frac{1}{\mu} \boldsymbol{\kappa} \langle \nabla p \rangle_{\mathbb{S}} \quad (4.4.1)$$

definiert werden. Darin ist $\langle \mathbf{v} \rangle_{\mathbb{S}}$ das gemittelte Geschwindigkeitsfeld eines Fluides mit dynamischer Viskosität μ . Der mikroskopische Druck ist p und $\langle \nabla p \rangle_{\mathbb{S}}$ ist der gemittelte Gradient dieses Druckfeldes. Diese Formulierung eröffnet den Zugang, die Permeabilität über Strömungssimulationen, welche ein Geschwindigkeits- und Druckfeld innerhalb des Porenraums liefern, zu berechnen. Im Allgemeinen ist die Permeabilität $\boldsymbol{\kappa}$, wie oben angegeben, eine richtungsabhängige tensorielle Größe (Tensor zweiter Stufe). Wird sie als Skalar κ angegeben, ist von einer isotropen Probe auszugehen. Die Permeabilität hat die Einheit einer Fläche und wird üblicherweise in mm² oder Darcy (D) angegeben, wobei gilt $1 \text{ D} = 0,9869 \mu\text{m}^2$. Es existieren mehrere empirischen Korrelationen zwischen der Porosität eines Mediums und dessen Permeabilität, siehe [14, 36], die aber nur für bestimmte Anwendungsbereiche gültig sind.

4.5 Spezifische Oberfläche, spezifische mittlere Krümmung und totale Krümmung

Die spezifische Oberfläche $S_V(\mathbb{G})$ ist das Verhältnis des Flächeninhaltes der Grenzfläche $S(\mathbb{G})$ des Gebiets \mathbb{G} zum Gesamtvolumen $V(\mathbb{S})$ der Probe

$$S_V(\mathbb{G}) = \frac{S(\mathbb{G})}{V(\mathbb{S})} \quad (4.5.1)$$

und hat damit die Dimension einer inversen Länge. In einem zweiphasigen porösen Medium mit $\mathbb{G} \in \{\mathbb{M}, \mathbb{P}\}$ gilt $S_V(\mathbb{M}) = S_V(\mathbb{P})$. Zur numerischen Berechnung der spezifischen Oberflächen wird in dieser Arbeit ein Algorithmus aus [61] verwendet.

Das spezifische Integral der mittleren Krümmung ist definiert als das Integral über die mittlere Krümmung $K(\mathbb{G})$ der Grenzfläche von \mathbb{G} dividiert durch das Gesamtvolumen der Probe,

$$K_V(\mathbb{G}) = \frac{K(\mathbb{G})}{V(\mathbb{S})} \quad (4.5.2)$$

mit $\mathbb{G} \in \{\mathbb{P}, \mathbb{M}\}$. Es gilt $K_V(\mathbb{P}) = -K_V(\mathbb{M})$. Die Berechnung der mittleren Krümmung $K(\mathbb{G})$ erfolgt mithilfe der zweidimensionalen Eulerzahl \mathcal{E}^2 .

Für die Definition des spezifischen Integrals der totalen Krümmung $T_V(\mathbb{G})$ wird die dreidimensionale Eulerzahl $\mathcal{E}^3(\mathbb{G})$ verwendet,

$$T_V(\mathbb{G}) = 4\pi \frac{\mathcal{E}^3(\mathbb{G})}{V(\mathbb{S})} \quad (4.5.3)$$

mit $\mathbb{G} \in \{\mathbb{P}, \mathbb{M}\}$ und es gilt $\mathcal{E}_V^3(\mathbb{P}) = \mathcal{E}_V^3(\mathbb{M})$ [52]. Zur Berechnung wird eine rekursive Beziehung zwischen der n-dimensionalen Euler-Zahlen \mathcal{E}^n und der (n-1)-dimensionalen Eulerzahl \mathcal{E}^{n-1} verwendet. Für eine ausführlichere Diskussion zur Interpretation und numerischen Berechnungen dieser Größen sei auf [58] und die darin aufgelistete, weiterführende Literatur hingewiesen.

4.6 Lokale Porositätstheorie (LPT)

In der lokalen Porositätstheorie (LPT) werden lokal berechnete Größen, hier im Sinne von in Teilgebieten der Gesamtprobe, statistisch ausgewertet. Typische lokale Größen können z. B. die Porosität, die spezifische Oberfläche, die spezifische mittlere Krümmung oder die spezifische totale Krümmung sein.

Als lokale Messzelle $\mathbb{K}(\mathbf{x}, L) \subset \mathbb{S}$ wird ein kubisches Teilgebiet der Seitenlänge L bezeichnet, dessen Mittelpunkt \mathbf{x} sei. Innerhalb der kubischen Gesamtprobe \mathbb{S} mit Seitenlängen $L_1 \times L_2 \times L_3$ seien nun m solcher sich gegenseitig teilweise überlappenden Messzellen platziert. In [38] wird gezeigt, dass die Form der Messzelle im Limit keinen Einfluss auf die berechneten Größen hat. In dieser Arbeit werden immer kubische Messzellen verwendet, da sie am besten der äußeren Geometrie der Probe entsprechen.

Für lokal in der Messzelle $\mathbb{K}(\mathbf{x}, L)$ berechnete reellwertige Funktionen $f_i(\mathbf{x}, L)$ mit $i = 1, \dots, n$ wird die lokale Geometrieverteilung definiert als

$$\mu(f_1, \dots, f_n, L) = \frac{1}{m} \sum_{\mathbf{x}} \prod_{i=1}^n \delta(f_i - f_i(\mathbf{x}, L)). \quad (4.6.1)$$

mit der Dirac'schen Deltafunktion δ , für Details siehe [37, 36].

Die lokale Geometrieverteilung ist eine Wahrscheinlichkeitsdichtefunktion. Daher ist die Größe $\mu(f_1, \dots, f_n, L)df_1 \dots df_n$ die Wahrscheinlichkeit dafür, dass das Ergebnis im Intervall $[f_1, f_1 + df_1], \dots, [f_n, f_n + df_n]$ liegt.

Für die lokale Verteilung der Porosität $\mu(\phi, L)$ gilt

$$\lim_{L \rightarrow 0} \mu(\phi, L) = (1 - \phi) \cdot \delta(\phi) + \phi \cdot \delta(\phi - 1) \quad , \quad (4.6.2)$$

$$\lim_{L \rightarrow \infty} \mu(\phi, L) = \delta(\phi - \bar{\phi}) \quad (4.6.3)$$

mit der mittleren Porosität $\bar{\phi}$ der Gesamtprobe \mathbb{S} . Nach 4.6.2 und 4.6.3 existiert eine mittlere Größe L^* der Messzelle, für die erstmalig gilt $\mu(0, L^*) = \mu(1, L^*) = 0$,

$$L^* = \min\{L : \mu(0, L) = \mu(1, L) = 0\}. \quad (4.6.4)$$

Das wichtigste geometrische Charakteristikum im Hinblick auf die Untersuchung von physikalischen Transportvorgängen ist die geometrische Konnektivität des Porenraumes, da es sich um eine notwendige Voraussetzung handelt.

Index α	Bedeutung
x	\mathbf{e}_1 Richtung
y	\mathbf{e}_2 Richtung
z	\mathbf{e}_3 Richtung
3	$(\mathbf{e}_1 \wedge \mathbf{e}_2 \wedge \mathbf{e}_3)$ Richtung
c	$(\mathbf{e}_1 \vee \mathbf{e}_2 \vee \mathbf{e}_3)$ Richtung

Tabelle 4.1: Definition des Richtungsindex α . Perkolierend in alle Richtungen entspricht $\alpha = 3$.

Ein Indikator für die Konnektivität einer Messzelle ist die lokale Perkolation $\Lambda_\alpha(\mathbf{x}, L)$. Es gilt $\Lambda_\alpha(\mathbf{x}, L) = 1$, falls die Messzelle $\mathbb{K}(\mathbf{x}, L)$ in Richtung α perkoliert, und $\Lambda_\alpha(\mathbf{x}, L) = 0$ sonst. Perkolation einer Messzelle in Richtung \mathbf{e}_i bedeutet, dass die senkrecht auf \mathbf{e}_i stehenden Oberflächen von $\mathbb{K}(\mathbf{x}, L)$ durch einen komplett im Porenraum liegenden Pfad verbunden sind. Damit ist die lokale Perkulationswahrscheinlichkeit

$$\lambda_\alpha(f_1, \dots, f_n, L) = \frac{\sum_{\mathbf{x}} \Lambda_\alpha(\mathbf{x}, L) \prod_{i=1}^n \delta_{f_i, f_i(\mathbf{x}, L)}}{\prod_{i=1}^n \delta_{f_i, f_i(\mathbf{x}, L)}} \quad (4.6.5)$$

mit dem Kronecker-Delta $\delta_{i,j}$. Die lokale Perkulationswahrscheinlichkeit $\lambda_\alpha(f_1, \dots, f_n, L)$ ist die Wahrscheinlichkeit, dass eine Messzelle mit Seitenlänge L und den geometrischen Messgrößen f_1, \dots, f_n in Richtung α perkoliert, siehe Tabelle 4.1.

Für den Gesamtanteil perkolierender Zellen gilt

$$p_\alpha(L) = \int \cdots \int \lambda_\alpha(f_1, \dots, f_n, L) \mu(f_1, \dots, f_n, L) df_1 \dots df_n. \quad (4.6.6)$$

Im Rest dieser Arbeit wird immer $\alpha = 3$ angenommen, falls nicht anders angegeben, und daher dieser Index nachfolgend unterdrückt. Es sei darauf hingewiesen, dass die LPT-Ergebnisse empfindlich von der verwendeten Auflösung a abhängen. Daher wird an den Stellen, wo dieser Effekt untersucht wird, dies auch explizit in der Größen vermerkt, wie $\mu(\phi, L, a)$ bzw. $\lambda_\alpha(\phi, L, a)$ und $p_\alpha(L, a)$. Auch ist es durch LPT-Messungen möglich, effektive Transportgrößen wie z. B. die Permeabilität zu bestimmen. In [64] werden dazu die Kozeny-Konstante mit einem Wert $k_K = 5$ und die Saturierungslänge L_δ , siehe [64] Seite 319 Gl. 15, als charakteristische Länge verwendet. Die Länge L_δ wird aus $p(L)$ berechnet. Eine umgekehrte Anwendung wird in [6] aufgezeigt. Darin werden aus experimentellen Messungen der dielektrischen Transportgrößen die Verteilungen der lokalen Porosität $\mu(\phi)$ und der lokalen Perkolation $\lambda_\alpha(\phi)$ berechnet. In den Arbeiten [46, 15] wird LPT genutzt, um die lokale Porositätsverteilung in Zementpaste bzw. einer Lehmphase in Schiefer zu untersuchen. In all diesen Arbeiten wird LPT jedoch nur auf Proben mit einer bestimmten Auflösung angewandt.

4.6.1 Auflösungsabhängige Beispielanwendung der LPT

Als Beispielanwendung zur Veranschaulichung typischer Ergebnisse aus der LPT wird die in [53] rekonstruierte Probe in Unterproben aufgeteilt und mit verschiedenen Auflösungen diskretisiert, siehe 4.1 und 4.2. Diese Probe wird in Kapitel 7 genauer untersucht. Eine ähnliche Analyse wird in [45] angewandt, dort allerdings auf Daten einer porösen Tonerde und nur bei einer festen Auflösung.

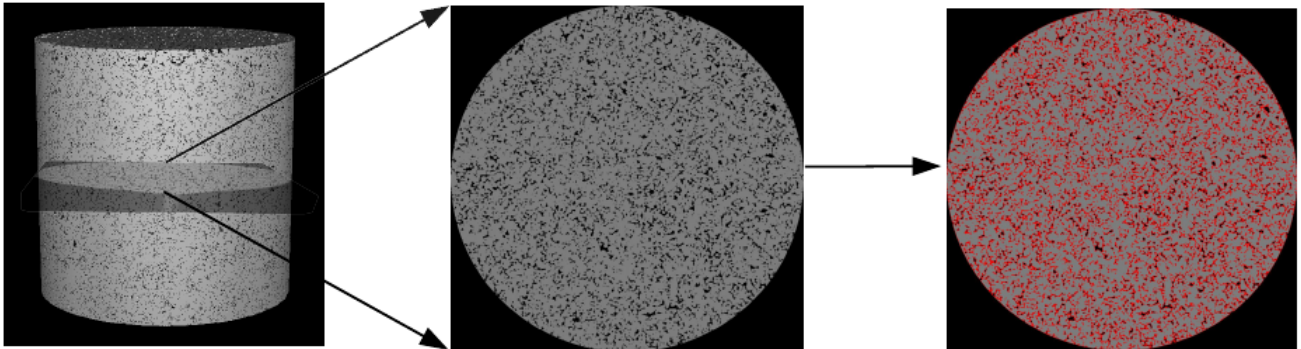


Abbildung 4.1: Links ein Bild der zylinderförmigen Rekonstruktion mit Höhe und Durchmesser 2.5 cm. In der Mitte und rechts Querschnitte durch die Probe, worin der Porenraum schwarz, die Gesteinsmatrix grau und die Ränder der Gesteinsmatrix rot sind.

Aus dem Kontinuumsmodell wird ein kubisches Stück extrahiert, diskretisiert und dann für LPT-Berechnungen verwendet. Dabei wird der Einfluss der Probengröße 1.024 cm, ..., 0.064 cm und der Auflösung $a = 180 \mu\text{m}, \dots, 2.5 \mu\text{m}$ untersucht, siehe Abb.4.3.

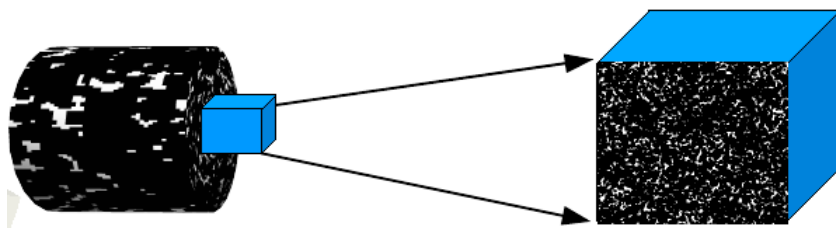


Abbildung 4.2: Darstellung, wie die kubische Probe aus dem Gesamtmodell entstanden ist.

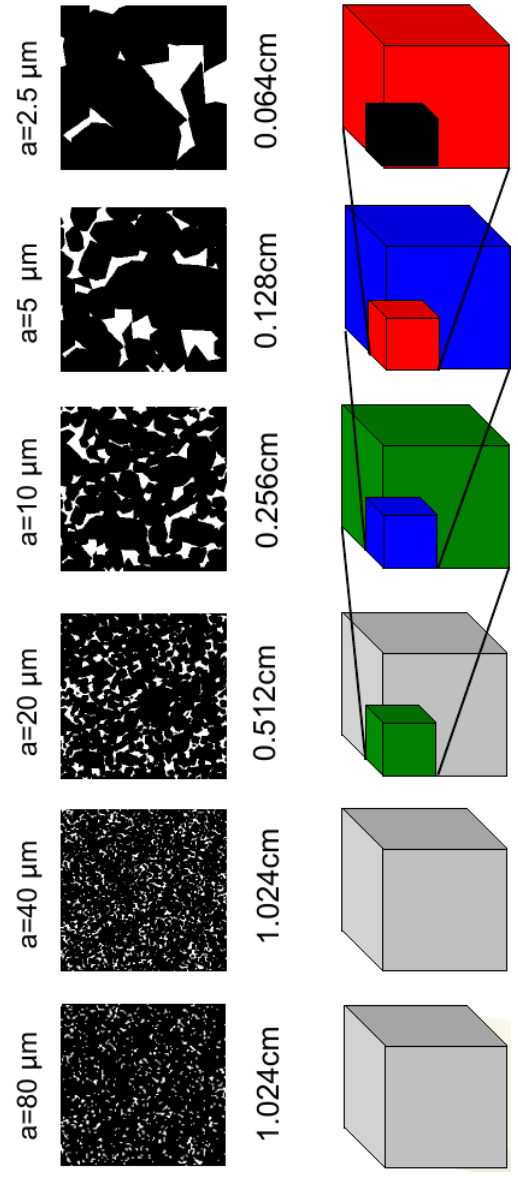


Abbildung 4.3: Oben Querschnitte mit verschiedenen Auflösungen $a = 180 \mu\text{m}, \dots, 2.5 \mu\text{m}$. Mit zunehmender Auflösung wird die kantige Struktur der einzelnen Kristalle deutlicher. Unten eine Übersicht über die verwendeten kubischen Teilproben mit Seitenlängen 1.024 cm, ..., 0.064 cm. Für die Verfeinerungen wurden jeweils Teilgebiete aus derselben Ecke verwendet.

Dadurch wird abgeschätzt was ein sinnvolles REV und eine ausreichende Auflösung zur Berechnung der lokalen Porositätsverteilung $\mu(\phi, L, a)$, der lokalen Perkolationswahrscheinlichkeit $\lambda_3(\phi, L, a)$ und dem Gesamtanteil perkolierender Zellen $p_3(L, a)$ ist. Diese Erkenntnisse werden in Kapitel 7 aufgegriffen und vertieft.

Für die Ergebnisse der lokalen Porositätsverteilung $\mu(\phi, L, a)$ scheint ein REV in der Größenordnung von $600 \mu\text{m} - 700 \mu\text{m}$ zu liegen und eine Auflösung ab $20 \mu\text{m}$ ausreichend, um ein repräsentatives Ergebnis zu beobachten.

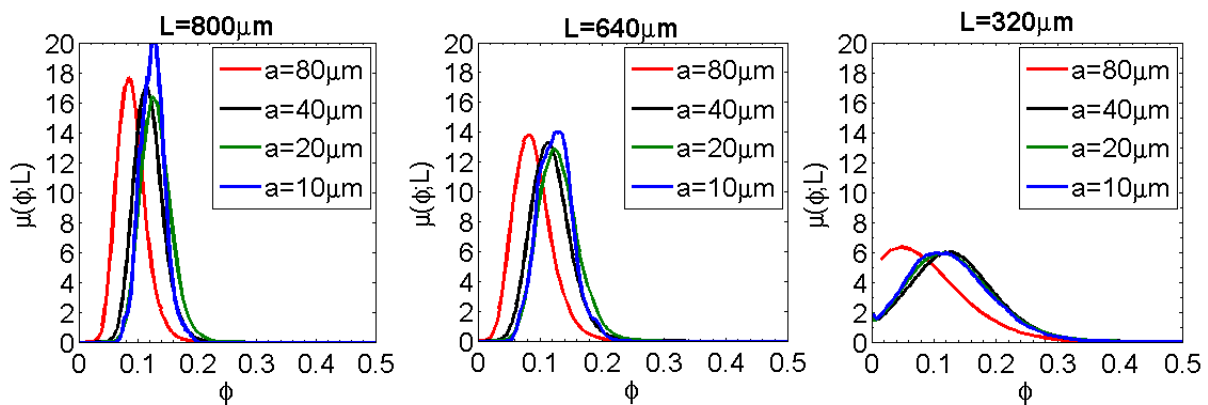


Abbildung 4.4: Auflösungsabhängige lokale Porositätsverteilung $\mu(\phi, L)$ als Funktion der lokalen Porositäten ϕ bei verschiedenen Messzellengrößen L . Die Verteilung ist bei $L = 640 \mu\text{m}$ deutlich schärfer als bei $L = 320 \mu\text{m}$. Ab einer Auflösung von $a = 40 \mu\text{m}$ ist ein repräsentatives Ergebnis zu beobachten.

Die Ergebnisse für die lokale Perkolationswahrscheinlichkeit $\lambda(\phi, L)$ und der Gesamtanteil perkolierender Zellen $p(L, a)$ scheinen deutlich sensibler im Bezug auf die Auflösung zu sein. Erst ab $a = 10 \mu\text{m} - 20 \mu\text{m}$ lässt sich von annähernd konvergierten Resultaten ausgehen, siehe Bilder 4.5, 4.6.

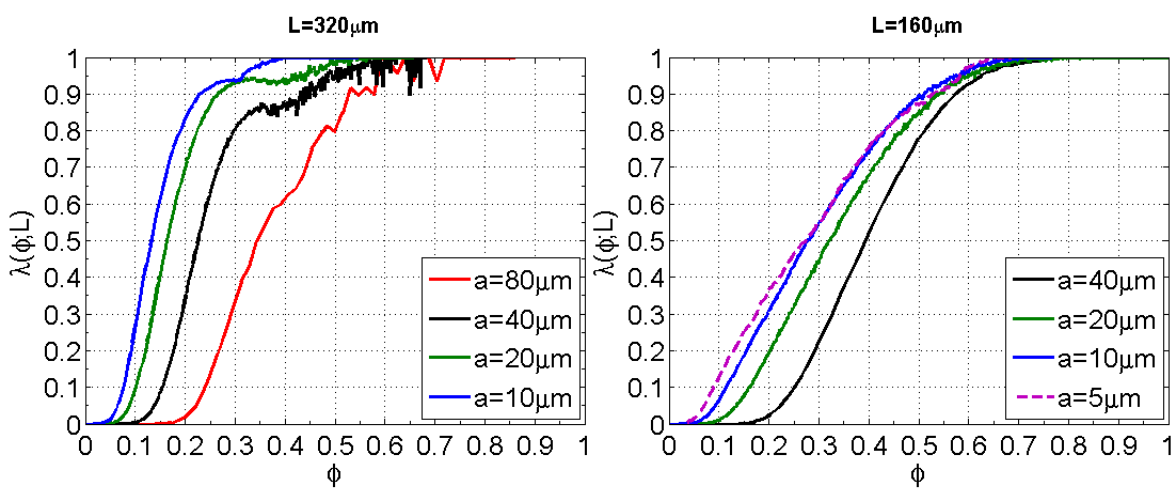


Abbildung 4.5: Auflösungsabhängige lokale Perkolationswahrscheinlichkeiten $\lambda(\phi, L)$ als Funktion der lokalen Porositäten ϕ bei verschiedenen Messzellengrößen L und Auflösungen a .

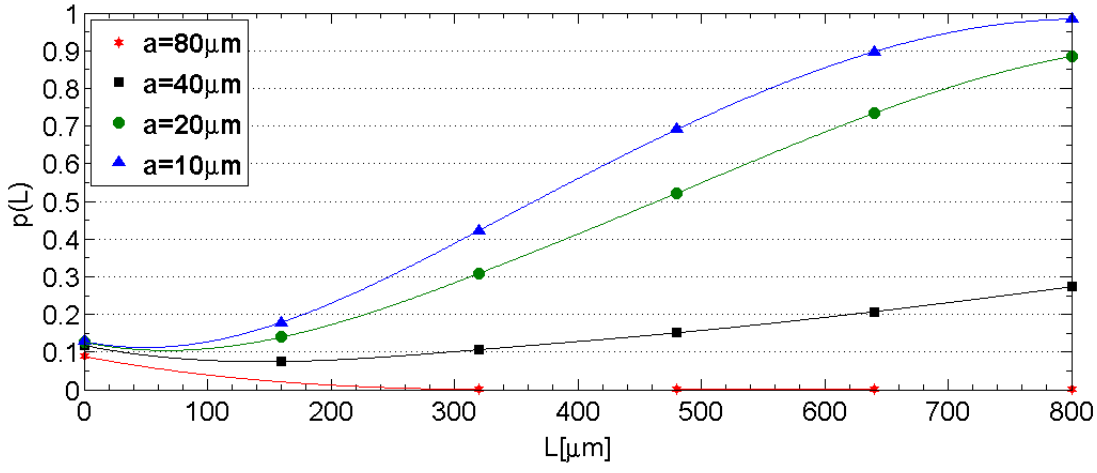


Abbildung 4.6: Totaler Anteil der perkolierenden Zellen $p(L, a)$ bei verschiedenen Probenlängen L und Auflösungen a .

4.7 Modellierung und Diskretisierung granularer poröser Medien

In dieser Arbeit werden Modelle für granulare poröse Medien in einem mehrschrittigen Verfahren erzeugt. Ausgehend von experimentellen Datensätzen wird ein Kontinuumsmodell erzeugt. Es enthält alle Informationen, um davon abgeleitete diskretisierte 3D-Datensätze mit beliebiger Auflösung zu generieren.

4.7.1 Kontinuums-Rekonstruktion eines dreidimensionalen Kontinuumsmodells

Die in dieser Arbeit verwendete Prozedur zur Erzeugung einer digitalen Rekonstruktion eines granularen porösen Mediums basiert auf folgenden drei Schritten:

1. **Datenerfassung:** Aus den experimentellen bildgebenden Daten, 2D-Schnitten oder 3D-Daten, werden beschreibende Kenngrößen wie z. B. Porosität, Korngrößenverteilung, repräsentative Korntypen, Korndichte, spezifische innere Oberfläche und Permeabilität erfasst bzw. berechnet.
2. **Packungsgenerierung:** Es wird eine Packung bestehend aus sich überlappenden Kugeln mit vorgegebener Radienverteilung und Dichte erstellt. Bereits in diesem Schritt können verschiedene Teilgebiete der Gesamtprobe mit unterschiedlichen Verteilungen und Dichten erzeugt werden. Das Resultat ist eine Datenstruktur der Mittelpunkte und Radien aller Kugeln.
3. **Dekoration:** Ausgehend von der Kugelpackung werden alle Kugeln durch statistisch ausgewählte Körner der zur Verfügung stehenden Korntypen ersetzt. Dabei sind deren Mittelpunkte durch die Mittelpunkte der Kugeln definiert. Die räumliche Orientierung wird stochastisch oder durch einen weiteren Algorithmus ausgewählt bzw. modifiziert (siehe Abschnitt 8.2.2).

Das Ergebnis ist dann ein wie in [9] beschriebenes Kontinuumsmodell eines granularen porösen Mediums basierend auf den Kornpositionen, Korntypen, Korngrößen und Kornorientierungen.

4.7.2 Diskretisierung des Kontinuumsmodells zu binären 3D-Datensätzen

Ausgehend von einem Kontinuumsmodell, wie in [9] beschrieben, werden diskretisierte Datensätze wie folgt erzeugt.

Das Probengebiet \mathbb{S} wird in N kubische Würfel, genannt Voxel, unterteilt $\mathbb{V}_i(a) \subset \mathbb{S}$ mit $i = 1, \dots, N$. Die Voxel haben eine Seitenlänge a , im Weiteren Auflösung genannt. Jedes Voxel $\mathbb{V}_i(a)$ wird nun durch $6^3 = 216$ gleichmäßig zentrierte Stützstellen abgerastert. Durch diese Stützstellen erhält jeder Voxel einen Grauwert $0 \leq g_i(a) \leq 216$ mit $1 \leq i \leq N$. Dieser Grauwert ist gleich der Anzahl an Stützstellen in jedem Voxel, welche innerhalb der Matrix \mathbb{M} und nicht im Porenraum \mathbb{P} liegen.

Bei der darauf folgenden Segmentierung wird jedem Voxel mit Grauwert $g_i(a)$ ein binärer Wert $b_i(a) \in \{0, 1\}$ zugeordnet. Dazu wird ein Schwellwert $0 \leq g_{th} \leq 215$ definiert und es gilt dann

$$b_i(a) = \begin{cases} 0 & g_i(a) \leq g_{th} \\ 1 & g_i(a) > g_{th} \end{cases}$$

Die Festlegung des Schwellwerts kann entweder trivial, z. B. $g_{th} = 128$ erfolgen oder nach der statistischen Verteilung des Grauwert-Histogramms. In dieser Arbeit wird zur Schwellwertbestimmung das Verfahren von Otsu [62] verwendet. In Kapitel 7.2 und 7.3 wird die Rolle dieses Schwellwerts erneut untersucht und es wird ein Verfahren vorgestellt, das die Berechnung der Porosität und der spezifischen Oberfläche direkt aus den Grauwerten $g_i(a)$ ermöglicht. Der Datensatz $b_i(a)$ mit $1 \leq i \leq N$ wird in dieser Arbeit als binäre 3D-Diskretisierung bei der Auflösung a verstanden.

5 Diskrete-Elemente-Methode zur Generierung dichter Kugelpackungen

Es wurden in der Vergangenheit diverse Methoden verwendet, um Packungen aus überlappenden oder berührenden Kugeln mit vorgegebenen Eigenschaften wie Packungsdichte, Überlappungsverteilung oder Kugelgrößenverteilung zu erzeugen. Die verschiedenen Algorithmen lassen sich meist in stochastische- oder Diskrete-Elemente-Ansätze unterteilen. Die stochastischen Methoden versuchen bei vorgegebenen Randbedingungen sequenziell Kugeln in das Modell einzufügen, wobei die Position und die Größe statistisch gewählt werden. Auf den ersten Blick sind diese Methoden sehr vielversprechend, da sie einfach erscheinen. Es wird jeweils eine Kugel hinzugefügt, bis ein Abbruchkriterium erfüllt ist, z. B. die Packungsdichte oder die Gesamtanzahl. Bei Packungen, die aus Kugeln stark variierender Größen bestehen sollen, und solchen bei denen die Packungsdichte sehr hoch sein soll, wird dieser Ansatz jedoch ineffizient. Zu Beginn der Simulation werden gültige Kugelpositionen und Konfigurationen noch schnell gefunden, aber zum Ende hin nimmt die Effizienz dieses Ansatzes massiv ab. Der Ansatz bei der Diskrete-Elemente-Methode (DEM) ist dann besser. Ausgehend von einer Startkonfiguration, die nicht den gewünschten Randbedingungen entspricht, werden Regeln zwischen Kugeln (allg. Elementen) angewandt, die in diskreten Simulationsschritten die Positionen und Radien ändern um die gewünschten Randbedingungen besser zu erfüllen. Eine Abbruchbedingung muss entscheiden, wann genau eine akzeptable Konfiguration erreicht ist. Natürlich wird auch dieses Verfahren bei schwierigeren Randbedingungen langsamer konvergieren. Da aber beständig die Gesamtpackung variiert werden kann, ist wesentlich besser geeignet, um nach einigen Schritten eine gültige Konfiguration zu finden. Die Schwierigkeit bei der DEM ist es, passende Regeln zu definieren, die schnell zu einer gültigen Endkonfiguration führen. Innerhalb der DEM Methoden wird unterschieden zwischen dynamischen und geometrischen. Bei geometrischen Methoden bestimmen heuristische Regelwerke unter Berücksichtigung der aktuellen Konfiguration (Geometrie) die neuen Positionen und Größen der Kugeln [70, 69]. Bei den dynamischen Methoden wird eine Bewegungsgleichung,

oftmals angelehnt an die gedämpfte newtonsche Gleichung, verwendet um die neuen Positionen zu berechnen. Eine Unterart der dynamischen DEM ist als Molekulardynamik-Simulation (MD) bekannt. Bei der MD beruhen die Regeln zwischen den Elementen, dort Moleküle oder Metamoleküle genannt, auf realistischen physikalischen Wechselwirkungskräften. Das Ziel in MD-Simulationen ist es, einen physikalischen Vorgang, wie z. B. die Faltung eines Proteins, möglichst exakt und zeitaufgelöst beschreiben zu können. Daher wurde viel Arbeit in eine möglichst gute Modellierung der Wechselwirkungskräfte verwendet. Bei der DEM, so wie sie in dieser Arbeit verwendet wird, zählt nur die Endkonfiguration. Es wird kein Modell für die physikalische Entstehungsgeschichte eines porösen Mediums erstellt. Die hier verwendeten Regeln zwischen den Kugeln sind zwar inspiriert von physikalischen Kraftgesetzen, aber eigentlich in ihrer Güte nur dadurch zu bewerten, wie effektiv sie zu einer Endkonfiguration führen. Die Bezeichnungen Kräfte, Kraftfeld und Potentiale sind hier nur im übertragenen Sinne zu verstehen, wurden aber beibehalten, weil sie der physikalischen Intuition helfen.

5.1 Definitionen, Initialisierung und Abbruchbedingungen

Verwendet wird ein zeitdiskretisierter Kraftfeldalgorithmus [80], mit Zeitintervall $\Delta t \in \mathbb{R}_{>0}$ und den Zeitschritten $1 \leq s \leq s^{\max}$. Zum Zeitschritt s enthält die Region $\Omega \subset \mathbb{R}^3$ die Kugeln O_i , $i = 1, \dots, N$ mit den Koordinaten $\mathbf{x}_i^s \in \Omega$, den Radien $r_i^s \in \mathbb{R}_{>0}$, Massen $m_i \in \mathbb{R}_{>0}$ und dem Kugeltyp $t_i \in \{1, \dots, N_g\}$. Für jeweils zwei Kugeln ist die Überlappung definiert als

$$\sigma_{ij}^s = \frac{r_i^s + r_j^s - dx_{ij}^s}{r_i^s + r_j^s - |r_i^s - r_j^s|}$$

worin $dx_{ij}^s = |\mathbf{x}_i^s - \mathbf{x}_j^s|$ der Abstand ist. Für jeden Typ ist eine maximal zulässige Überlappung festgelegt als $\sigma_{t_i}^{\max} \in \mathbb{R}$. In jedem Zeitschritt werden erst die Kräfte auf jede Kugel berechnet und dann diese entsprechend bewegt. Näheres zur Kraftberechnung und Zeitintegration sind im Kapitel 5.2 beschrieben. Eine Kugel gilt als konvergiert, wenn $0 < \max_{i \neq j}(\sigma_{ij}) \leq \sigma_{t_i}^{\max}$. Das Gesamtsystem ist konvergiert, wenn dies für alle Kugeln gilt. Der entsprechende (finale) Zeitschritt ist $s_f \in \mathbb{N}$. Ein solches System ist dann im Sinne der Anforderung eine gültige (Kugel-)Packung. Die Anforderungen beim Starten des Systems sind die Größe und Form der Region Ω , die Anzahl N der Kugeln

oder eine Punktdichte ρ sowie die Überlappungsbeschränkungen $\sigma_{t_i}^{\max} \in \mathbb{R}$. Beim Simulationsanfang wird das Gebiet stochastisch, im Einklang mit N bzw. ρ , mit Kugeln befüllt. Für jeden Kugeltyp t_i muss eine Größenverteilung angegeben werden, um die Anfangsradien r_i^0 festzulegen. Verschiedene Möglichkeiten sind implementiert.

- **Diskret**

Eine Liste gültiger Radien $r_j \in \mathbb{R}$ und zugehöriger Wahrscheinlichkeiten $P'(r_j) \in [0, 1]$, so dass $\sum_j P'(r_j) = 1$ gilt, muss vorgegeben werden. Es gilt dann für die Radienverteilung

$$P(r_i^0 = r) = \sum_j \delta(r - r_j)P'(r_j)$$

mit δ der Dirac-Delta-Funktion.

- **Konstant und kontinuierlich**

Ein oberer Grenzradius $r_{t_i}^{\max} \in \mathbb{R}$ sowie unterer Grenzradius $r_{t_i}^{\min} \in \mathbb{R}$ wird für jeden Kugeltyp vorgegeben und es gilt dann für die Radienverteilung $P(r_i^0 = r) = 1/(r_{t_i}^{\max} - r_{t_i}^{\min})$.

- **Exponential und kontinuierlich**

Ein oberer Grenzradius $r_{t_i}^{\max} \in \mathbb{R}$ sowie unterer Grenzradius $r_{t_i}^{\min} \in \mathbb{R}$ werden für jeden Kugeltyp vorgegeben. Weiterhin wird eine Konstante $\alpha \in \mathbb{R}$ für alle Kugeltypen spezifiziert.

Es gilt dann für die Radienverteilung $P(r_i^0 = r) = \frac{\alpha \exp(-\alpha r)}{(\exp(-\alpha r_{t_i}^{\min}) - \exp(-\alpha r_{t_i}^{\max}))}$.

Nachdem das System initialisiert ist, wechselt der Algorithmus zwischen einem Kraftberechnungs- und Propagationsschritt, bis eine Abbruchbedingung erfüllt ist - also entweder bis das System konvergiert ist, oder die maximal zulässige Anzahl an Zeitschritten überschritten wird. Das Endergebnis der Simulation sind die kontinuierlichen Koordinaten $\mathbf{x}_i^{s_{\max}} \in \Omega$, Radien $r_i^{s_{\max}} \in \mathbb{R}$ und der diskrete Kugeltyp $t_i \in \{1, \dots, N_g\}$. Bei allen Simulationen wurde $m_i = 1$ verwendet, trotzdem wird die Masse in den Formeln weiter verwendet.

5.2 Kraftfeld und Kraftberechnung

Die Kraft $\mathbf{f}_i^s \in \mathbb{R}^3$ auf die Kugeln O_i im Zeitschritt s wird als Summe von paarweisen Wechselwirkungskräften berechnet

$$\mathbf{f}_i^s = \sum_{s_j \in W_i^s} \mathbf{f}_{ij}(dx_{ij}^{s-1}, r_i^{s-1}, r_j^{s-1}, k_{ij}^s) \quad (5.2.1)$$

mit den wechselwirkenden Kugeln $O_j \in W_i^s$ aus der Verlet-Liste W_i^s , siehe Gleichung 5.5.1, und den Wechselwirkungskonstanten $k_{ij}^s = \frac{1}{2}(k_{t_i}^s + k_{t_j}^s)$. Die Menge an Wechselwirkungspartnern W_i^s wird in einem vorherigen, nicht notwendigerweise dem $(s-1)$ -ten Schritt, bestimmt 5.5.3. Durch die maximal zulässige Überlappung $\sigma_{t_i}^{\max}$ ist eine minimale Distanz zwischen einer Kugel des Typs t_i und aller anderen Kugeln definiert. Dieser minimale Abstand ist $dx_{\min,ij}^{s-1} = r_i^{s-1} + r_j^{s-1} - \sigma_{t_i}^{\max}(r_i^{s-1} - r_j^{s-1} - |r_i^{s-1} - r_j^{s-1}|)$. Weiterhin definiert sei der Abstand $dx_{\max,ij}^{s-1} = r_i^{s-1} + r_j^{s-1}$, also der maximale Abstand bei dem zwei Kugeln O_i und O_j noch nicht getrennt sind. Als Gleichgewichtsabstand ist definiert $dx_{eq,ij}^{s-1} = \frac{1}{2}(dx_{\max,ij}^{s-1} - dx_{\min,ij}^{s-1}) = \sigma_{t_i}^{\max}(r_i^{s-1} - r_j^{s-1} - |r_i^{s-1} - r_j^{s-1}|)$, bei dem zwischen zwei Kugeln keine Kraft wirkt. Die Paarkräfte sind als Potentialkräfte angesetzt, also $\mathbf{f}_{ij}(dx_{ij}^{s-1}, r_i^{s-1}, r_j^{s-1}) = -\nabla V_{ij}(dx_{ij}^{s-1}, r_i^{s-1}, r_j^{s-1})$, mit einem harmonischen und einem verallgemeinerten Lennard-Jones-Anteil

$$V_{ij}^{s-1} = V_{ij}(dx_{ij}^{s-1}) = \begin{cases} V^H(dx_{ij}^{s-1}) & dx_{ij}^{s-1} \leq dx_{eq,ij}^{s-1}(r_i^{s-1}, r_j^{s-1}) \\ V^{LJ}(dx_{ij}^{s-1}) & dx_{ij}^{s-1} > dx_{eq,ij}^{s-1}(r_i^{s-1}, r_j^{s-1}) \end{cases}$$

mit dem harmonischen Anteil

$$V^H(dx_{ij}^{s-1}) = \frac{(1 - V_{min}^{LJ}) \sqrt{k_{ij}^{s-1}}}{2} \cdot 1 - \frac{dx_{ij}^{s-1}{}^2}{dx_{eq,ij}^{s-1}} + V_{min}^{LJ}$$

und dem Lennard-Jones-Anteil

$$V^{LJ}(dx_{ij}^{s-1}) = k_{ij}^{s-1} \cdot \frac{b_{ij}^{s-1}{}^m}{dx_{ij}^{s-1}} - \frac{b_{ij}^{s-1}{}^n}{dx_{ij}^{s-1}}. \quad (5.2.2)$$

mit $V_{min}^{LJ} = V^{LJ}(dx_{eq,ij}^{s-1}) < 0$, $b_{ij}^{s-1} = (\frac{n}{m})^{1/(m-n)} dx_{eq,ij}^{s-1}$ und $k_{ij}^{s-1} = \frac{1}{2}(k_{t_i}^{s-1} + k_{t_j}^{s-1})$, dem Mittelwert der Kraftkonstante für die Kugeltypen t_i und t_j . Die Implementation benutzt $m = 16$ und $n = 8$. Dadurch ist die maximale abstoßende Kraft etwa drei mal so groß wie die maximale attraktive Kraft, siehe Abb. 5.1.

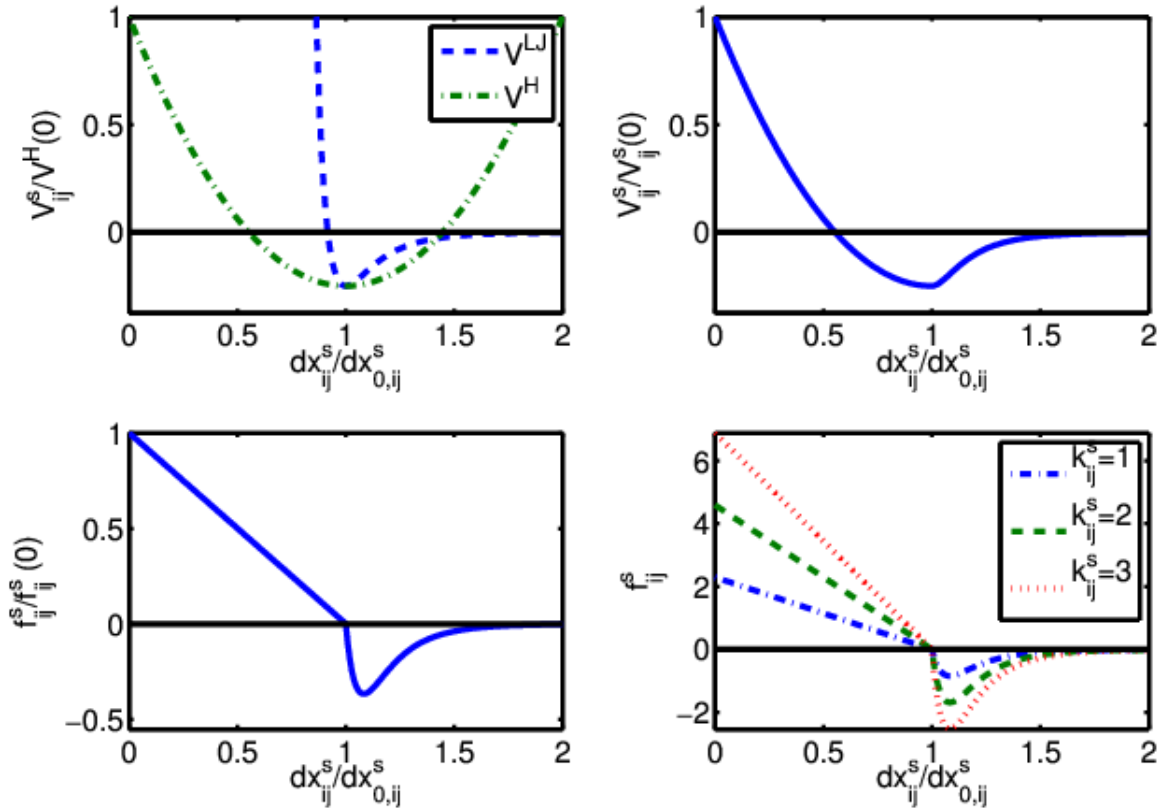


Abbildung 5.1: Die skalierten Potentiale V^H und V^{LJ} (oben links) und das skalierte Potential $V_{ij}^s(dx_{ij}^{s-1})$ (oben rechts) als Funktion des dimensionslosen Abstands $dx_{ij}^s/dx_{0,ij}^s$. Die entsprechenden skalierten Kräfte $|f_{ij}^s|/|f_{ij}^s(0)|$ für verschiedene Interaktionsparameter k_{ij}^s (unten rechts, links).

Im Vergleich mit einem rein harmonischen bzw. reinen Lennard-Jones-Potential ist das hier verwendete Potential endlich bei $dx_{ij}^{s-1} = 0$ und konstant bei großen Distanzen $\lim_{dx_{ij}^{s-1} \rightarrow \infty} V_{ij}^{s-1} = C$.

Die abgeleiteten Kräfte ergeben sich mit $\mathbf{e}_{ij}^{s-1} = \frac{\mathbf{x}_i^{s-1} - \mathbf{x}_j^{s-1}}{dx_{ij}^{s-1}}$ zu

$$\mathbf{f}_{ij}^{s-1} = \mathbf{f}_{ij}(dx_{ij}^{s-1}) = \begin{cases} f^{LJ}(dx_{ij}^{s-1}) \cdot \mathbf{e}_{ij}^{s-1} & dx_{ij}^{s-1} \leq dx_{eq,ij}^{s-1}(r_i^{s-1}, r_j^{s-1}) \\ f^H(dx_{ij}^{s-1}) \cdot \mathbf{e}_{ij}^{s-1} & dx_{ij}^{s-1} > dx_{eq,ij}^{s-1}(r_i^{s-1}, r_j^{s-1}) \end{cases}$$

worin

$$f^H(dx_{ij}^{s-1}) = \sqrt{k_{ij}^{s-1}} \cdot \frac{(1 - V_{min}^{LJ})}{dx_{eq,ij}^{s-1}} \cdot 1 - \frac{dx_{ij}^{s-1}}{dx_{eq,ij}^{s-1}}$$

und

$$f^{LJ}(dx_{ij}^{s-1}) = k_{ij}^{s-1} \cdot \frac{m(b_{ij}^{s-1})^m}{(dx_{ij}^{s-1})^{m+1}} - \frac{n(b_{ij}^{s-1})^n}{(dx_{ij}^{s-1})^{n+1}}. \quad (5.2.3)$$

Die attraktive Kraft verringert sich schnell mit zunehmender Distanz und kann daher ab einer so genannten Abschneidedistanz dx_{cut} vernachlässigt werden

$$|\mathbf{f}_{ij}(dx_{ij}^{s-1})| = 0 \text{ für } \frac{dx_{ij}^{s-1}}{dx_{eq,ij}^{s-1}} > dx_{cut}. \quad (5.2.4)$$

Die daraus resultierenden Kräfte sind Radialkräfte entlang des Vektors $(\mathbf{x}_i^s - \mathbf{x}_j^s)$. Andere Ansätze mit tangentialen Komponenten werden unter anderem in Kontaktsimulationen mit harten Kugeln verwendet oder wenn nicht sphärische Objekte in einer Packung untersucht werden sollen [16, 5].

5.3 Zeitintegration

Als Zeitpropagator wird ein geringfügig abgeändertes Geschwindigkeit-Strömer-Verlet-Schema [47] verwendet. Darin werden die Geschwindigkeiten berechnet wie

$$\mathbf{v}_i^s = \frac{(C_{\Xi}^-)_{t_i}^s}{(C_{\Xi}^+)_{t_i}^s} \mathbf{v}_i^{s-1} \Delta t + \frac{1}{(C_{\Xi}^+)_{t_i}^s} \cdot (\mathbf{f}_i^{s-1} + \mathbf{f}_i^s) \Delta t / (2m_i)$$

mit Δt der Zeitdiskretisierung und die Größen $(C_{\Xi}^+)_{t_i}^s = 1 + \frac{\Delta t}{2} \Xi_{t_i}^s$, $(C_{\Xi}^-)_{t_i}^s = 1 - \frac{\Delta t}{2} \Xi_{g_i}^s$ durch die Dissipationskonstanten $\Xi_{t_i}^s$ für jeden Kugeltyp t_i gegeben sind. Aus der Definition ist offensichtlich, dass $\Xi_{t_i}^s \leq \frac{2}{\Delta t}$ gelten muss, damit $\frac{(C_{\Xi}^-)_{t_i}^s}{(C_{\Xi}^+)_{t_i}^s} > 0$ gilt, sonst könnte die Dissipation die Bewegungsrichtung instantan umkehren. Die Positionen sind dann gegeben durch

$$\mathbf{x}_i^s = \mathbf{x}_i^{s-1} + (C_{\Xi}^-)_{t_i}^s \mathbf{v}_i^s \Delta t + \mathbf{f}_i^s \Delta t^2 / (2m_i).$$

Bei diesem Schema werden im Propagationsschritt die Kräfte aus dem aktuellen und aus dem vorherigen Schritt benötigt.

5.4 Steuerungsoptionen

Durch die Einführung situationsabhängiger Parameter, deren Verhalten beim Simulationsstart festgelegt wird, lassen sich komplexe Protokolle verwenden. Darunter wird eine Abfolge von sich ändernden Parametern je nach Zeitschritt verstanden. Dazu gehören die Interaktionsparameter der Kugeln, die Dämpfungsparameter oder spezielle Schwellwerte, welche bestimmte Berechnungen an- bzw. ausschalten.

5.4.1 Zeitabhängige Interaktions- und Dämpfungsparameter

Für jeden Kugeltyp $t_i \in \{1, \dots, N_g\}$ können Wechselwirkungsparameter $k_{t_i}^0 \in \mathbb{R}$ und $k_{t_i}^1 \in \mathbb{R}$ sowie Start- und Endschritte $S_{k,t_i}^0 \in \mathbb{N}$, $S_{k,t_i}^1 \in \mathbb{N}$ angegeben werden. Für die Wechselwirkungsparameter gilt dann:

$$k_{t_i}^s = \begin{cases} k_{t_i}^0 & s < S_{k,t_i}^0 \\ k_{t_i}^{s-1} + dk_{t_i} & S_{k,t_i}^0 \leq s < S_{k,t_i}^1 \\ k_{t_i}^1 & s \geq S_{k,t_i}^1 \end{cases} \quad (5.4.1)$$

$$\text{mit } dk_{t_i} = \frac{k_{t_i}^1 - k_{t_i}^0}{S_{k,t_i}^1 - S_{k,t_i}^0}.$$

Ebenso können, um die Dissipation zeitabhängig zu steuern, die Größen $\Xi_{t_i}^0$, $\Xi_{t_i}^1$, S_{Ξ,t_i}^0 , S_{Ξ,t_i}^1 spezifiziert werden. Es gilt dann:

$$\Xi_{t_i}^s = \begin{cases} \Xi_{t_i}^0 & s < S_{\Xi,t_i}^0 \\ \Xi_{t_i}^{s-1} + d\Xi_{t_i} & S_{\Xi,t_i}^0 \leq s < S_{\Xi,t_i}^1 \\ \Xi_{t_i}^1 & s \geq S_{\Xi,t_i}^1 \end{cases} \quad (5.4.2)$$

$$\text{mit } d\Xi_{t_i} = \frac{\Xi_{t_i}^1 - \Xi_{t_i}^0}{S_{\Xi,t_i}^1 - S_{\Xi,t_i}^0}.$$

5.4.2 Stochastische Kräfte

Kontinuierlich gleichverteilte Zufallskräfte $\mathbf{f}_i^{rnd,s} \in [0, f_{max}]^3$ mit $f_{max} \in \mathbb{R}_{>0}$ können für alle Kugeln O_i angesetzt werden, falls keine anderen Kräfte aus der paarweisen Aufsummation der Wechselwir-

kungskräfte vorhanden sind. Das kann eintreten wenn z. B. die Kugel weiter als der Abschneideradius dx_{cut} von allen anderen Kugeln entfernt ist und nicht als konvergiert gilt. Das kann genutzt werden, um aus anfänglich dünn verteilten Systemen verbundene Strukturen zu erzeugen oder um gegen Ende der Simulation einzelne noch nicht verbundene Kugeln mit bestehenden Strukturen zu verbinden. Die dadurch entstehenden Strukturen sind isotrop.

5.4.3 Adaptive Kugelgrößen

Der Algorithmus kann dahingehend konfiguriert werden, dass sich die Kugelgrößen während des Simulationsverlaufes situationsabhängig verändern. Dazu müssen eine Schranke $t_{rc} \in \mathbb{R}$ definiert werden, ein Startschritt und Endschritt $s_{rchg}^0, s_{rchg}^1 \in \mathbb{N}$, eine Anzahl von Schritten s_{nr} sowie eine Änderungsrate $\delta r \in \mathbb{R}$. Der Adaptionsmechanismus ist aktiv nur in den Schritten $s_{rchg}^0 < s < s_{rchg}^1$ und $s < s^{max} - s_{nrc}$, wenn gilt $(N_c^{s-1} - N_c^s)/N_c^s < t_{rc}$, worin N_c^s die Anzahl der konvergierten Kugeln ist. Der Parameter s_{nrc} (für 'no-radius-change') gibt eine Anzahl von Schritten am Ende der Simulation an, in denen keine Radiusänderung stattfinden darf. Wenn der Adaptionsmechanismus aktiv ist, gilt für die neuen Kugelradien

$$r_i^s = \begin{cases} r_i^{s-1} + (r_{t_i}^{max} - r_{t_i}^{min}) \cdot \delta r & \text{wenn } \max_j(\sigma_{ij}^s) < 0 \\ r_i^{s-1} - (r_{t_i}^{max} - r_{t_i}^{min}) \cdot \delta r & \text{wenn } \exists j : \sigma_{ij}^s > \sigma_{t_i}^{max} \end{cases}$$

Dadurch werden freischwebende Kugeln vergrößert, um bestenfalls im nächsten Schritt mit einer bestehenden Struktur in der Nähe verbunden zu werden. Kugeln, die mit der Umgebung eine übermäßige Überlappung haben, werden verkleinert. Der Parameter wurde eingeführt, um am Ende der Simulation, nach eventuellen Größenänderungen, dem System Zeit zum Relaxieren zu geben. Bei der Erzeugung dichter Kugelpackungen verkleinert dieser Mechanismus offensichtlich die anfangs erzeugte Radienverteilung.

5.4.4 Einfrieren von Freiheitsgraden

Wenn dieser Mechanismus aktiviert ist, durch Angabe eines Zeitschrittes s_{lock} , werden für alle Schritte $s > s_{lock}$ die Kraftberechnung und die Propagation der betroffenen Kugeln nicht ausgeführt. Betroffen sind Kugeln, die als konvergiert gelten. Diese werden dann als eingefroren markiert. Einfgefrorene Kugeln können jedoch wieder in einen nicht konvergierten Zustand wechseln, z. B. falls

eine benachbarte Kugel sich löst oder ihnen zu nahe kommt. Dieser Mechanismus sollte mit Vorsicht verwendet werden, da er unter Umständen auch die Konvergenz des Gesamtsystems verlangsamen kann. Es können lokale Situationen entstehen, die sich zu nur schwach gedämpften Schwingungen aufschaukeln. Das kann passieren, wenn einzelne Kugeln oder kleinere Kollektive als konvergiert eingestuft und dadurch eingefroren werden, sich dann jedoch die Konfiguration der Umgebung so ändert, dass im nächsten Schritt die Kugeln nicht mehr als konvergiert gelten dürfen, freigegeben werden und es durch diesen Schrittversatz zu Schwingungen eines lokalen Bereichs kommt.

5.5 Effiziente parallele Implementation

Als Parallelisierungstechnologie wurde eine Gebietszerlegung ('domain decomposition') ausgewählt. Hierbei wird das räumliche Simulationsgebiet in disjunkte Teilgebiete zerlegt. Diese Bereiche werden Domänen genannt. Jeweils eine Recheneinheit, also ein Prozessor oder Rechenkern oder CPU-Thread, behandelt nur eine solche Domäne. Jede Recheneinheit bzw. Domäne erhält eine eindeutige Identifikation (ID). Bei diesem Parallelisierungsparadigma entsteht ein Kommunikationsbedarf zwischen den Recheneinheiten, da die Information über das Gesamtsystem fragmentiert und verteilt wird und Oberflächeneffekte zwischen den Domänen entstehen. Diese Technologie ist speziell geeignet für sehr große Systeme, bei denen eventuell gar nicht alle anfallenden Daten bzw. Systeminformationen innerhalb eines lokalen Speichers verarbeitet werden können, und für Systeme bei denen der erwähnte Kommunikationsaufwand nur mäßig mit der Systemgröße zunimmt, also z. B. konstant ist oder nur linear mit der Systemgröße skaliert. Eine andere im Bereich der DEM weitverbreitete Parallelisierungstechnologie versucht die Systeminformationen nicht zu fragmentieren, sondern mehrere Berechnungskerne auf denselben Datensätzen arbeiten zu lassen. DEM-Anwendungen, die dieses Paradigma zusammen mit der OpenMP (Open Multi-Processing)-Implementation auf einer CPU-GPU-Architekturen nutzen, sind in [71] zu finden. Wie wichtig effektive Parallelisierungsstrategien und Kommunikationsalgorithmen sind, ist auch der Arbeit [54] zu entnehmen, dort allerdings angewandt auf einen Algorithmus für Strömungssimulationen in porösen Medien.

5.5.1 Parallelisierung durch Gebietszerlegung

Es sei $\Omega = [0, L_k]^3 \subset \mathbb{R}^3$ das rechteckige Simulationsgebiet mit Seitelängen L_k , mit $k = 1, 2, 3$. Es wird unterteilt in disjunkte rechteckige Domänen Ω_q^d mit $q = 1, \dots, N^d = \prod_k \frac{L_k}{l_k^d}$ der Seitenlängen l_k^d , und es gilt $\Omega = \bigcup_q \Omega_q^d$. Jede Domäne Ω_q^d wiederum wird unterteilt in rechteckige Teilgebiete Ω_{qp}^c , genannt Zellen, der Seitenlängen l_k^c mit $p = 1, \dots, N^{cd} = \prod_k \frac{l_k^d}{l_k^c}$ und $\Omega_q^d = \bigcup_p \Omega_{qp}^c$. Der größte vorkommende Kugelradius $R^{\max} = \max_{i,s} \{r_i^s\}$ beschränkt die Zellgröße nach unten, weil $l_k^c \geq 2 * R^{\max}$ gelten muss. Die Gesamtanzahl der Zellen ist $N^c = N^{cd} \cdot N^d$.

5.5.2 Effektive Kommunikation im Rahmen des MPI-Standards mit nur sechs P2P-Operationen

Die Implementation in dieser Arbeit nützt als Kommunikationsmechanismus zur 'Inter-Process-Communication' (IPC), also zur Kommunikation zwischen den für einzelne Domänen zuständigen Recheneinheiten, den MPI ('Message Passing Interface')-Standard [24]. DEM-Implementationen, die eine Gebietszerlegung als Parallelisierungstechnologie nutzen, brauchen aus zwei Gründen IPC. Die Kraftberechnung muss auch Interaktionen von Kugeln, die in verschiedenen Domänen sind, beachten und weil sich Kugeln von einem Gebiet in ein anderes bewegen können. Die Daten dieser Kugeln, also neue Positionen, Geschwindigkeiten, Typen, Radien und Wechselwirkungskonstanten, müssen aus den Datenstrukturen der einen Recheneinheit gelöscht und in die Datenstrukturen der anderen Recheneinheit transferiert werden. Die Verantwortung für das Verhalten dieser Kugeln liegt dann bei der neuen Recheneinheit. Dazu werden die Daten mittels MPI-IPC versendet. Eine Kommunikationsstrategie sind z. B. MPI-Broadcasts. Dabei sendet jede Recheneinheit alle Daten der Kugeln, die im aktuellen Zeitschritt ihren Bereich verlassen, an alle anderen Recheneinheiten. Diese Kommunikation wird 'All-to-All'- oder 'A2A'-Kommunikation genannt. Alle Recheneinheiten untersuchen wiederum die Nachrichten aller anderen Recheneinheiten nach Kugeln, die in ihr Gebiet gehören, übernehmen diese Kugeldaten und ignorieren den Rest der Nachricht. Der Vorteil dieser Kommunikationsstrategie ist ihre Einfachheit, da die Domänen selber keinerlei Informationen über ihre Lage innerhalb des Gesamtgebiets kennen müssen. Es müssen auch keine Informationen über benachbarte Domänen vorhanden sein. Der Nachteil ist der große Kommunikationsaufwand einer A2A-Kommunikation, diese skaliert mit $O((N^d)^2)$. Auch die Datengrößen jeder Nachricht an sich ist erheblich, da jede Nachricht immer alle die Domäne verlassenden Kugeldaten enthält.

Die vorliegende Implementation benutzt den 'Rand-Replika-Zellen'-Ansatz zusammen mit bekannten lokalen Nachbarschaftsinformationen und einer speziellen P2P-Kommunikationsstrategie, um die Anzahl der Nachrichten und deren Größe zu minimieren. Jede Domäne ist aufgebaut aus einem zellinneren Bereich, aus Randzellen und Replikazellen, siehe Abb. 5.2.

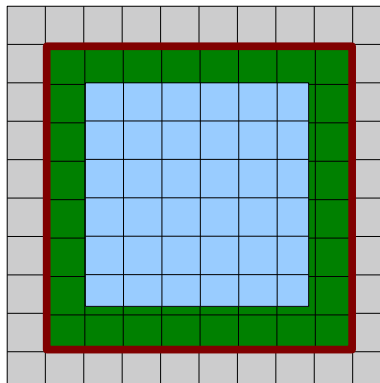


Abbildung 5.2: 2D-MPI-Domäne (rot berandet) mit inneren Zellen (blau), Randzellen (grün) und Replikazellen (grau). Graue Replikazellen entsprechen grünen Randzellen benachbarter Domänen.

Randzellen ('Rim-Cells') sind die äußersten Zellen, die noch innerhalb der Domäne liegen, deren Zelloberfläche aber auch die Oberfläche der Domäne bildet. Replikazellen ('Halo'-, 'Shadow'- oder 'Ghost'-Zellen) sind duplizierte Zellen mit den Informationen aller Randzellen der benachbarten Domänen.

Bei der hier verwendeten Kommunikationsstrategie senden Domänen nur Nachrichten an ihre benachbarten Domänen. Diese Kommunikation heißt 'Point-2-Point'- bzw. 'P2P'-Kommunikation. Dadurch erhält jede Domäne auch nur Nachrichten von ihren jeweiligen Nachbarn. Des Weiteren sind die Nachrichten an sich auch kleiner, da jede Nachricht nur Informationen über Kugeln enthält, die in diese räumliche Richtung propagiert sind, und nicht generell über alle die Domäne verlassenden Kugeln. Bei einer trivialen Implementation fallen also für jede Domäne 26 Nachrichten an. Durch eine geschickte zeitliche Abfolge von Sendeoperationen und der 'Rand-Replika'-Struktur lässt sich der Kommunikationsaufwand auf sechs P2P-Nachrichten pro Domäne beschränken, der Aufwand wird also um mehr als einen Faktor vier reduziert, siehe [47]. Bei dieser Implementation werden nur Nachrichten zu Nachbarn entlang der Hauptachsen des Systems verschickt und nicht zu den

Nachbarn, die nur über Kanten und Ecken verbunden sind, daher nur sechs Nachrichten. Es gibt in jeder Iteration zwei grundsätzlich verschiedene Schritte zum Informationsaustausch, sie heißen "Rand-Zu-Replika" und "Replika-Zu-Rand". Ein detailliertes Ablaufschema eines Iterationsschrittes ist im Anhang 9.1 zu finden.

1. "Rand-Zu-Replika" (Rim-To-Halo) Schritt: Darin sendet jede Domäne die Informationen aller Kugeln, die in ihrem Rand liegen, an die benachbarte Domäne, welche diese Kugeln in ihre Replika-Zellen einfügt. Dort können sie dann für die nächste Kraftberechnung in Betracht gezogen werden.
2. "Replika-Zu-Rand" (Halo-To-Rim) Schritt: Darin sendet jede Domäne die Informationen aller Kugeln, die in ihren Replikazellen liegen, an die benachbarte Domäne, welche diese Kugeln in ihre Randzellen einfügt. Diese neuen Kugeln sind nun in der Verantwortung dieser Domäne, d.h. die Kraftberechnung und Propagation muss für sie durchgeführt werden.

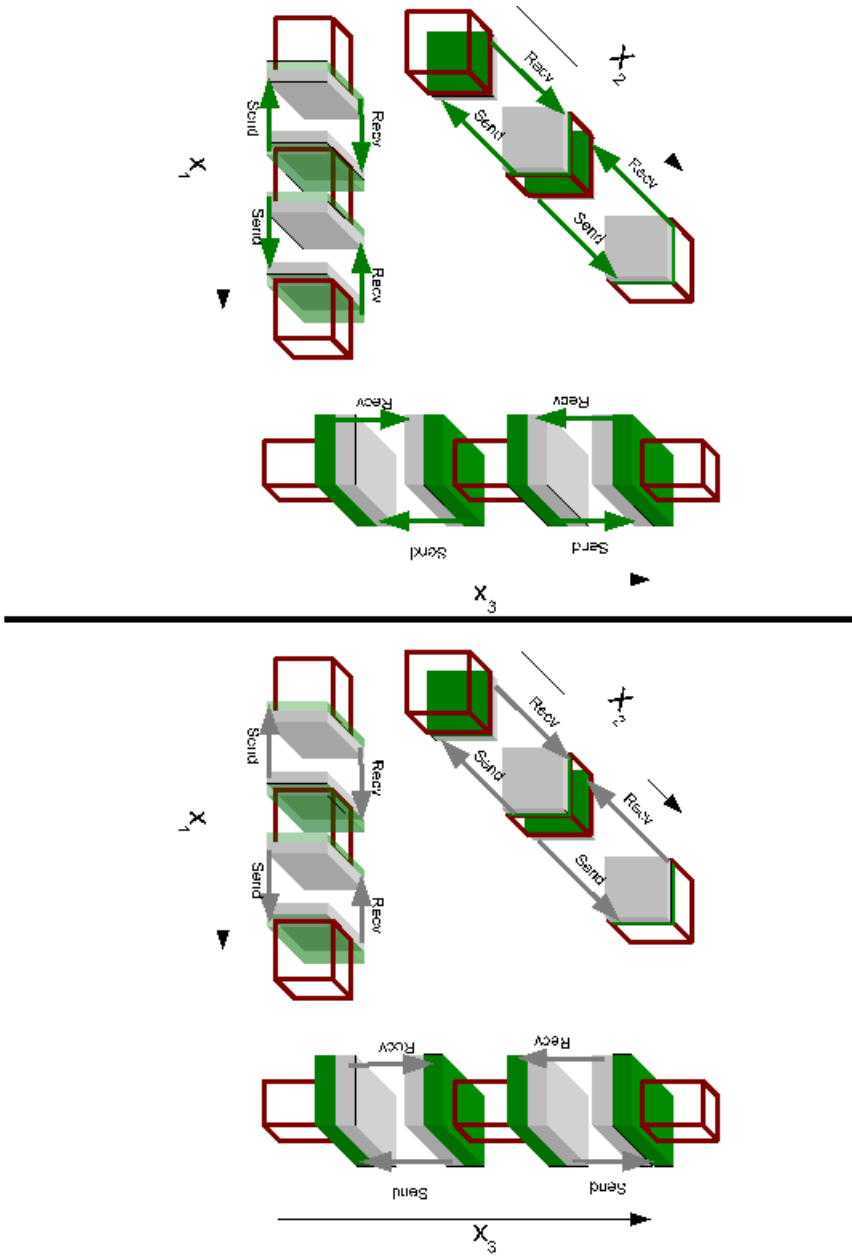


Abbildung 5.3: Effiziente P2P-MPI-Kommunikationsstrategie (Diagramm basierend auf [47] S. 132-133). Links: Die "Rand-Zu-Replica" Übertragungssequenz $x_2 \rightarrow x_1 \rightarrow x_3$. Rechts: Die "Replica-Zu-Replica" Übertragungssequenz $x_3 \rightarrow x_1 \rightarrow x_2$. Wichtig sind die verschiedenen räumlichen Anteile (grün, grau) der Rand- und Replikazellen für jede Domäne.

5.5.3 Nachbarschaftsstrukturen, Interaktionslisten und Mehrschrittverfahren

Durch die oben erwähnten lokalen Nachbarschaftsstrukturen (linked-cell-method) [4, 47] wird das Skalierungsverhalten der Kraftberechnung 5.2 von $O(N^2)$ auf $O(N)$ gedrückt. Bei dieser Implementation hat jede Zelle $\Omega_g^c \subset \mathbb{R}^3$, $g = 1, \dots, N^c$ eine Liste L_g mit den Indizes der benachbarten Zellen. Die Liste L_g ist definiert durch die Gebietszerlegung und dadurch statisch während der Simulation. Die Vereinigung einer Zelle Ω_g^c und all ihrer Nachbarzellen ist die Umgebung $U_g = \Omega_g^c \cup_{n \in L_g} \Omega_n^c \subset \mathbb{R}^3$. Bei der Berechnung der Kraft \mathbf{f}_i^s auf eine Kugeln O_i , mit $\mathbf{x}_i^s \in \Omega_g^c$, also innerhalb einer Zelle Ω_g^c werden als Wechselwirkungspartner nur Kugeln aus der Umgebung U_g betrachtet. Ob ein solcher Partner zum Potential beiträgt, hängt weiterhin von der verwendeten Abschneidedistanz dx_{cut} ab, siehe Gleichung 5.2.4. Es gilt in Gleichung 5.2.1, also

$$W_i^s = \{O_j \mid \mathbf{x}_j^{s-1} \in U_g \wedge \frac{dx_{ij}^{s-1}}{dx_{o,ij}^{s-1}} \leq dx_{cut}\} \quad (5.5.1)$$

Diese Menge wird auch 'Verlet-Liste' genannt [75]. Durch die Verlet-Listen wird die Kraftberechnung 'lokal' und hat daher eine lineare Komplexität. Eine weitere erhebliche Beschleunigung des Gesamtalgorithmus lässt sich erreichen, wenn die Verlet-Listen nicht in jedem Schritt neu bestimmt werden, sondern nur wenn $s \bmod (s_{mts}) = 0$ mit typischen Werten $s_{mts} = 2 - 10$. Diese Näherung heißt 'Multiple-Time-Step Approximation' und beruht auf der Annahme, dass sich die Wechselwirkungspartner für Kugeln in einer dichten Packung nur langsam ändern. Kugeln, die während eines Zyklus von einer Umgebung U_g in eine andere Umgebung U_{g^*} wechseln, sind für andere Kugeln nicht sichtbar. Es ist auch denkbar, eine Simulation mit $s_{mts} = 1$ zu starten, dann nach einer Phase der Konsolidierung die Simulation anzuhalten und mit der entstandenen räumlichen Konfiguration und $s_{mts} > 1$ einen neuen Simulationslauf zu starten.

5.5.4 Komplexität und Leistungsfähigkeit der Implementation

Aus den oben beschriebenen Methoden und Approximationen resultiert eine Implementierung, die bei P MPI-Prozessen Berechnungsoperationen der Komplexität $O(N/P)$ erfordert, einen Kommunika-

tionsaufwand von $O((N/P)^{2/3})$ und Speicheranforderungen von ebenfalls $O(N/P)$ hat. Das daraus resultierende Programm wurde bzgl. Stabilität und Komplexität auf einer HPC (High-Performance-Computing) Plattform mit bis zu 16342 Prozessen erfolgreich zum Generieren von Kugelpackungen getestet. In [31] wurde diese Implementierung genutzt, um Packungen von mehreren hundert Millionen Kugeln zu generieren, siehe 5.6.3. In [3] wurden ca. 80 Packungen mit jeweils um die 100.000 Kugeln generiert, um optimale Parameter für die Rekonstruktion eines Bentheimer Sandsteines zu finden, siehe 8. Es soll an dieser Stelle auch kritisch klargestellt werden, dass die angewandten Methoden ihre Grenzen finden und unzulänglich werden. Für Systeme mit eventuell zeitlich variierenden großen Dichteschwankungen auf Längenskalen der Domänengrößen wird die zeitlich konstante Gebietszerlegung ineffizient. Bei weitreichenden Kraftpotentialen werden die Nachbarschaftsstrukturen und Wechselwirkungslisten entweder sehr groß und damit ineffizient in der Kraftberechnung oder die getroffenen Approximationen zu grob und verfälschen damit die lokale Strukturbildung zu stark. Beim Auftreten hoher Geschwindigkeiten kann die Replika-Rand-Methode, bei der effektiv eine Kugel nach dem Verlassen einer Domäne erst in der nächsten Iteration in die korrekte Zelle der Nachbardomäne einsortiert wird, zu groben Fehlern führen. Diese Beschränkungen treffen aber auf den Hauptanwendungsfall von dichten, polydispersen Packungen mit nur lokalen, mäßig starken Wechselwirkungen nicht zu.

5.6 Anwendungsbeispiele

Im Folgenden werden zwei Anwendungsbeispiele diskutiert. Im Abschnitt 5.6.1 wird eine Packung mit stark unterschiedlichen Kugeln generiert und in Abschnitt 5.6.3 eine sehr große monodisperse Kugelpackung. Dabei werden verschiedene Protokolle genutzt, um ein Endergebnis zu erreichen. Eine weitere Anwendung, in der das Kugelmodell bis zu einem realistischen Modell eines Bentheimer Sandsteins weiterentwickelt wird, wird getrennt in Abschnitt 8 besprochen.

5.6.1 Hochkorrelierte dichte polydisperse Kugelpackung

In dieser Anwendung wird eine große Packung mit $N \approx 10^6$ Kugeln erzeugt. Es werden zwei verschiedene Kugeltypen mit starkem Größenunterschied verwendet. Bei der entstehenden Mikrostruktur sollen die kleinen Kugeln die Porenzwischenräume, besonders die engen Porenhäse, zwischen den großen Kugeln ausfüllen und sich an den Oberflächen der großen Kugeln ansammeln. Eine Porosität von unter 20 % soll realisiert werden. Das Ergebnis dieser Anwendung ist in Abb. 5.4 als 3D-Rendering dargestellt.

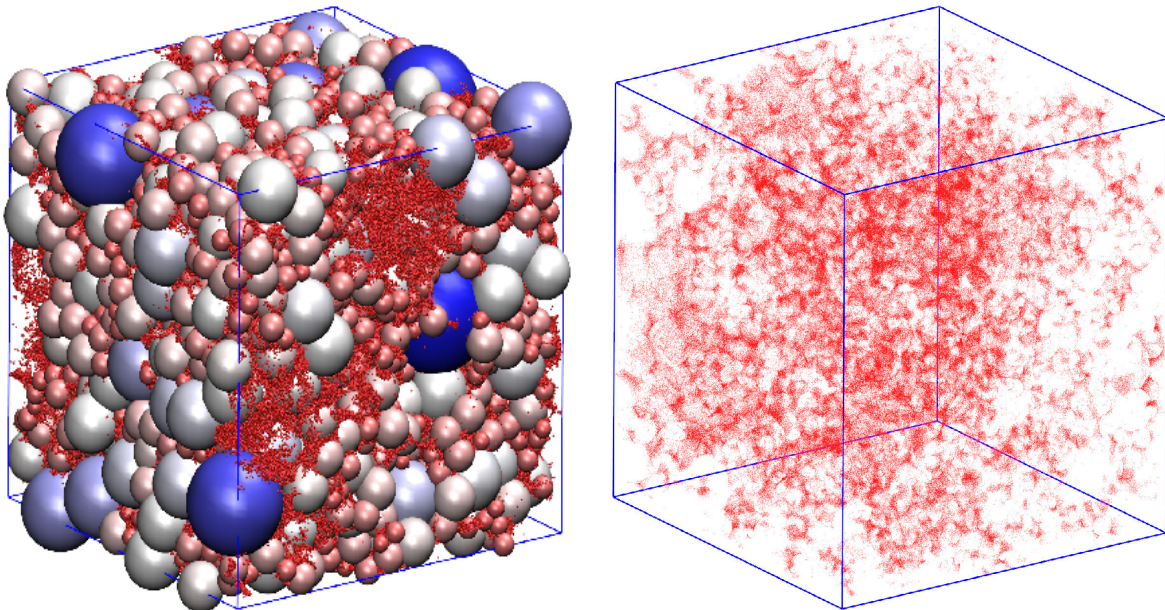


Abbildung 5.4: Links: 3D-gereordnete Darstellung der Kugelpackung, mit den großen bläulichen Kugeln und den kleinen roten Kugeln. Rechts: Die Zentren der kleinen Kugeln alleine. Es lässt sich erkennen, dass deren Mittelpunkte Cluster bilden und sich an die großen Kugeln angelagert haben.

5.6.1.1 Simulationsparameter und Ablauf

Das Simulationsgebiet mit quadratischen Seitenlängen $L_k = 300 \mu\text{m}$ enthält $N_1 = 2255$ große Kugeln des Typs $t_i = 1$ mit $i = 1, \dots, N_1$ und einer exponentiellen Radienverteilung, die durch $\alpha = 0.2$ und $r_1^{\min} = 10 \mu\text{m}$, $r_1^{\max} = 100 \mu\text{m}$ definiert ist, siehe 5.1. Diese Kugeln bilden nach der ersten Stufe der Simulation die primäre Matrix. Unter Matrix wird hier die verbundene Struktur der ausgefüllten Kugeln verstanden. In einer zweiten Simulationsphase sollen die Porenhäule und Oberflächen der primären Packung durch die kleinen Kugeln ausgefüllt werden. Diese $N_2 = N - N_1$ kleinen Kugeln vom Typ $t_i = 2$ mit $i = N_1 + 1, \dots, N$ haben Radien im Bereich von $r_2^{\min} = 0.9 \mu\text{m}$ bis $r_2^{\max} = 1.2 \mu\text{m}$.

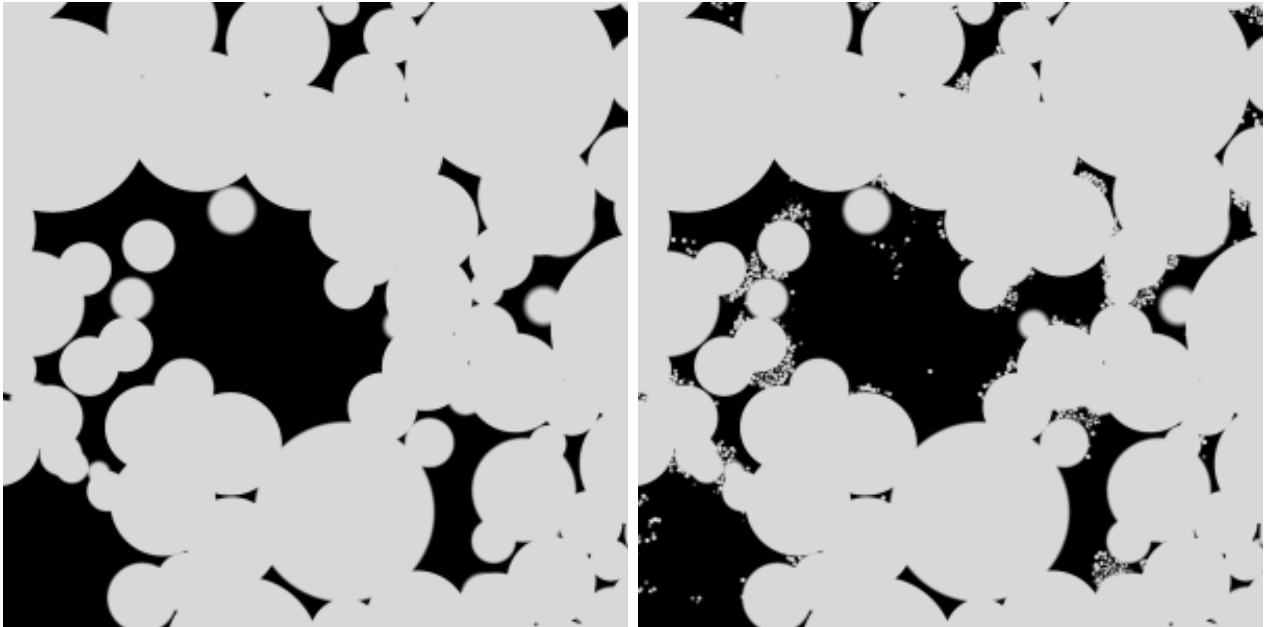


Abbildung 5.5: 2D-Schnitt durch die diskretisierte Packung mit Auflösung $1 \mu\text{m}$. Die Bilder haben eine Größe von 300×300 Voxel. Die Matrix, gebildet aus den als Festkörper betrachteten Kugeln, ist in Grau dargestellt. In Schwarz sind die Porenzwischenräume dargestellt. Das linke Bild zeigt die primäre Packung nach der ersten Simulationsphase, das rechte Bild die Packung nach der zweiten Simulationsphase. Die kleinen Kugeln füllen die Porenzwischenräume der primären Packung.

Die Gesamtsimulation besteht aus zwei hintereinander ausgeführten Phasen.

1. Phase: Die großen Kugeln vom Typ $t_i = 1$ werden stochastisch im Simulationsgebiet verteilt. Die Simulation wird gestartet mit den Interaktionskonstanten $k_i = 400$ und Dämpfungskonstanten $\Xi_i = 4$. Die maximal erlaubte Überlappung ist $\sigma_i^{\max} = 0.3$. Einfrieren von Freiheitsgraden 5.4.4 ist aktiviert mit $s_{lock} = 10$. Das System benötigt 20 Schritte bis zur Konvergenz.

2. Phase: In dieser Phase wird das Gebiet stochastisch mit einem zweiten Kugeltyp $t_i = 2$ befüllt. In dieser Phase werden die großen Kugeln eingefroren. Die Radien sind anfangs alle gleich mit $r_i = 1.2 \mu\text{m}$. Die maximal erlaubte Überlappung ist $\sigma_2^{\max} = 0.25$. Das Iterationsprotokoll ist spezifiziert durch $k_1^0 = 1200, k_1^1 = 1, s_{k,1}^0 = 20, s_{k,1}^1 = 25$ für $t_i = 1$ und $k_2^0 = 1, k_2^1 = 20, s_{k,1}^0 = 25, s_{k,1}^1 = 40$ für $t_i = 2$. Die Dissipation ist definiert durch $\Xi_2^0 = 1, \Xi_2^1 = 2$ mit $s_{\Xi,2}^0 = 30, s_{\Xi,2}^1 = 40$. Für die kleinen Kugeln sind Radiusänderungen, siehe 5.4.3, erlaubt mit den Einstellungen $\delta r = 0.02, t_{rc} = 0.1, s_{rchg}^0 = 50, s_{rchg}^1 = 80$ und $s_{nrc} = 10$. Daher sind die kleinen Kugeln am Ende der Simulation nicht mehr von gleicher Größe. Durch diese Protokolleinstellungen lagern sich die kleinen Kugeln, weil $k_1^0 = 1200$ groß ist, schnell an den Oberflächen der großen Kugeln an und rutschen dann entlang dieser Oberflächen in die Porenhäse zwischen den großen Kugeln. Während dieser Zeit sehen sich die kleinen Kugeln gegenseitig nur sehr geringfügig. Dadurch entstehen in den Porenhäsen und an den Oberflächen der großen Kugeln Gebiete mit sehr großen Dichten von kleinen Kugeln, wobei die maximal erlaubte Überlappung von $\sigma_2^{\max} = 0.25$ aber sehr oft überschritten wird. Dann mit abklingender Interaktionskonstante zwischen den großen und kleinen Kugeln, k_1^1 ist klein, und anwachsender Interaktionskonstante zwischen den kleinen Kugeln, durch $k_2^1 = 20$, schwellen diese lokalisierten dichten Packungen an, da sich die kleinen Kugeln anhand ihrer zu dichten Packung tendenziell eher abstoßen. Dabei verstopfen sie die Porenhäse und bilden dort eine neue poröse Mikrostruktur. Diese Phase dauerte ca. 100 Schritte.

Das Resultat ist als 2D-Schnitt durch die Packung am Ende der Phase 1, der primären Packung, und der finalen Packung in Abbildung 5.5 zu sehen. Die Größenverteilung beider Kugeltypen und ein Histogramm der Überlappung am Ende der Simulation sind im Bild 5.6 dargestellt. Die Größenverteilung der großen Kugeln wird im Laufe der Simulation nicht verändert. Die Größenverteilung der kleinen Kugeln hat sich dahingehend verändert, dass am Ende der Simulation mehr als die Hälfte der Kugeln geschrumpft sind, damit sich eine Überlappung im Bereich von $0 \leq \sigma_i \leq 0.25$ realisieren lässt. Die Verteilung der Überlappung am Ende der Simulation kann erklärt werden, wenn man bedenkt, dass viele am Anfang nicht verbundene Kugeln, also eine Überlappung mit $\sigma_i < 0$, sofort

einfröieren, wenn sie an den Oberflächen der großen Kugeln anlagern. Dadurch entsteht die Häufung bei $0 < \sigma_i < 0.05$. Diese Kugeln verharren im Laufe der Simulation an den Oberflächen und bilden raue Oberflächen mit nur geringer Überlappung. Die kleinen Kugeln, welche zusammen in die Porenhälsa rutschen, haben zunächst eine durchschnittliche Überlappung, die weitaus größer ist als $\sigma_2^{\max} = 0.25$. Durch das Schrumpfen und zugleich Anschwellen dieser Mikropackungen entstehen dann Strukturen mit Kugeln, deren Überlappung eher am oberen Rand des gültigen Bereiches ist. Die finale Packung zeigt eine Statistik im erlaubten Bereich von $0 < \sigma_i \leq 0.25$ für alle $i = 1, \dots, N$.

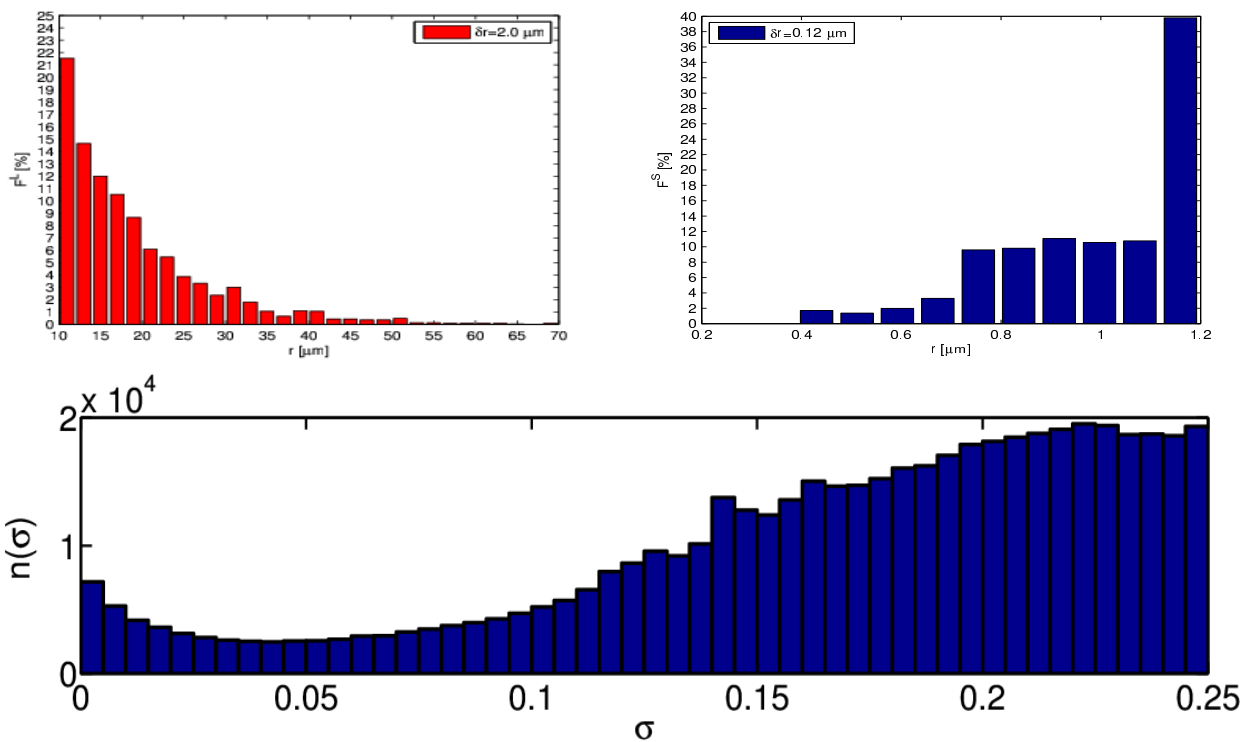


Abbildung 5.6: Oben links (rot): Histogramm der Größenverteilung F^L für die großen Kugeln mit $t_i = 1$ bei einer Auflösung von $dr = 2.0 \mu\text{m}$. Oben rechts (blau): Histogramm der Größenverteilung F^S oder kleinen Kugeln mit Auflösung von $dr = 0.12 \mu\text{m}$. Die anfänglich uniforme Größe der kleinen Kugeln von $r_i = 1.2 \mu\text{m}$ hat sich auf ein Intervall $r_i \in [0.9 \mu\text{m}, 1.2 \mu\text{m}]$ ausgeweitet. Unten (blau): Histogramm der Überlappung am Ende der Simulation. Alle Kugeln befinden sich im gültigen Bereich $0 \leq \sigma_i \leq 0.25$. Die Auflösung ist $d\sigma = 0.01$.

5.6.1.2 Diskretisierung der Kugelpackung

Die primäre Packung (PP), aus Phase 1 der Simulation, sowie die finale Packung (FP) wurden beide bei den Auflösungen $a = 0.25 \mu\text{m}$, $0.5 \mu\text{m}$, $1 \mu\text{m}$ diskretisiert. Dabei wurde das in [52] diskutierte Verfahren verwendet. Dadurch entstehen 3D-Datensätze, bei denen jeder Voxel durch einen Grauwert repräsentiert wird. Diese Grauwertdaten werden dann segmentiert, um die kubischen, binären Datensätze \mathbb{S}^{PP} und \mathbb{S}^{FP} der Seitenlänge L^V in Voxel zu erhalten, siehe Kapitel 4.7.2.

5.6.1.3 Geometrische Charakterisierung

Für die binären Datensätze \mathbb{S}^{PP} und \mathbb{S}^{FP} wurden auflösungsabhängig die Porosität, die spezifische Oberfläche S_V , das spezifische Integral der mittleren Krümmung K_V und das spezifische Integral der totalen Krümmung T_V nach dem Algorithmus in [61] berechnet. Die Ergebnisse sind in Tabelle 5.1 zu sehen. Die kleinen Kugeln haben einen Durchmesser von ca. $1.5 \mu\text{m} - 2 \mu\text{m}$ und sind daher bei der Auflösung von $a = 1 \mu\text{m}$ nur sehr schlecht als Voxel repräsentiert. Durch das Segmentieren der Grauwerte werden viele dieser Objekte räumlich verschoben oder fallen ganz weg. Erst bei der Auflösung von $a = 0.25 \mu\text{m}$ ist zu erwarten, dass diese Kugeln sinnvoll in den binären Daten repräsentiert werden. Die durch diese Objekte gebildeten Mikrostrukturen werden erst bei dieser Auflösung charakteristisch. Dadurch ist zu erklären, dass das \mathbb{S}^{FP} -Modell beim Übergang von $a = 0.5 \mu\text{m}$ auf $a = 0.25 \mu\text{m}$ erhebliche Änderungen in den Resultaten zeigt.

	$a[\mu\text{m}]$	$L^V[\text{Voxel}]$	ϕ	$S_V[\text{m}^{-1}]$	$K_V(\mathbb{P}) [\times 10^{-3} \text{m}^{-2}]$	$T_V(\mathbb{P}) [\times 10^{-4} \text{m}]$
\mathbb{S}^{PP}	1.0	300	0.182749	0.039	1.7449	-2.7909
\mathbb{S}^{PP}	0.5	600	0.182854	0.040	1.7619	-4.3873
\mathbb{S}^{PP}	0.25	1200	0.182844	0.041	1.7617	-8.8408
\mathbb{S}^{FP}	1.0	300	0.165780	0.055	2.1476	-64.932
\mathbb{S}^{FP}	0.5	600	0.166253	0.066	2.0549	-275.78
\mathbb{S}^{FP}	0.25	1200	0.166290	0.072	-0.9055	-498.52

Tabelle 5.1: Porosität ϕ , spezifische Oberfläche S_V , spezifisches Integral der mittleren Krümmung $K_V(\mathbb{P})$ und spezifisches Integral der totalen Krümmung $T_V(\mathbb{P})$ für die diskretisierten Modelle \mathbb{S}^{PP} (Primärpackung) und \mathbb{S}^{FP} (gefüllte Packung) bei verschiedenen Auflösungen a und entsprechenden Seitenlängen L^V .

5.6.1.4 Ergebnisse aus der lokalen Porositätstheorie (LPT)

Berechnungen im Rahmen der lokalen Porositätstheorie (LPT) wurden für beide binären Datensätze \mathbb{S}^{PP} und \mathbb{S}^{FP} bei den verschiedenen Auflösungen $a = 1.0r_2^{\min} = 0.9 \mu\text{m}$ bis $r_2^{\max} = 1.2 \mu\text{m}$, $0.5 \mu\text{m}$, $0.25 \mu\text{m}$ und verschiedenen Messzellengrößen $L = 10.0, 20.0, 40.0, 60.0, 80.0, 100.0, 200.0 [\mu\text{m}]$ durchgeführt. Die Ergebnisse für die lokalen Porositäten $\mu^{PP}(\phi, L, a)$ und $\mu^{FP}(\phi, L, a)$ sind in Abb. 5.7 dargestellt. Die Ergebnisse beider Packungen sind ähnlich, lediglich bei geringen Porositäten $0 < \phi < 0.15$ ist eine signifikante Änderung zu sehen. Viele dieser Messungen fallen wahrscheinlich in die Bereiche der Porenhälse, die im \mathbb{S}^{FP} -Modell durch eine Mikrostruktur aufgefüllt sind und die Statistik dementsprechend nach oben verschieben.

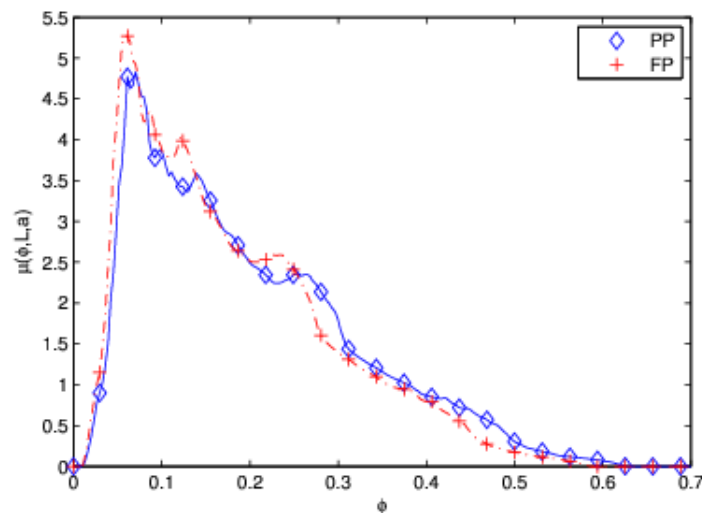


Abbildung 5.7: Lokale Porositäten $\mu^{FP}(\phi, L, a)$ (rot, Kreuze) und $\mu^{PP}(\phi, L, a)$ (blau, Quadrate) für $a = 0.25 \mu\text{m}$ und Messzellen der Seitenlänge $L = 80 \mu\text{m}$. Eine geringfügige Zunahme von Messungen mit niedriger Porosität ist im \mathbb{S}^{FP} Modell zu erkennen.

Die Berechnungen der lokalen Perkolationswahrscheinlichkeiten $\lambda^{FP}(\phi, L, a)$ und $\lambda^{PP}(\phi, L, a)$ zeigen, dass im Vergleich die Wahrscheinlichkeiten bei gleicher Porosität im \mathbb{S}^{FP} Modell erhöht sind im Vergleich zum \mathbb{S}^{PP} -Modell, siehe Abb. 5.8. Das ist verständlich, weil die eingefügten perkolierenden Mikrostrukturen mehr Möglichkeiten bereitstellen, bei einer gegebenen Porosität die Perkolation zu gewährleisten. Auch ist zu sehen, wie die Werte im \mathbb{S}^{FP} -Modell stärker von der Auflösung abhängen.

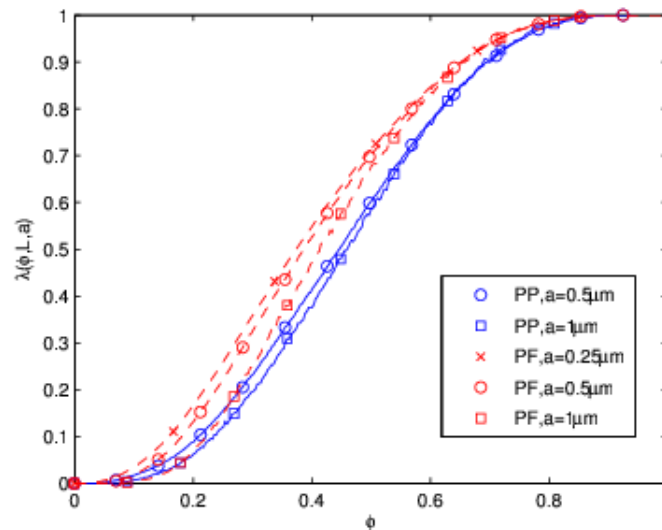


Abbildung 5.8: Lokale Perkolationswahrscheinlichkeiten $\lambda^{FP}(\phi, L, a)$ (rot, gestrichelt) und $\lambda^{PP}(\phi, L, a)$ (blau, durchgehend) für die Auflösungen $a = 0.25 \mu\text{m}$, $0.5 \mu\text{m}$, $1.0 \mu\text{m}$ und einer Messzelle der Seitenlänge $L = 10 \mu\text{m}$.

Der Gesamtanteil perkolierender Messzellen mit Seitenlänge L ist $p(L)$ und in Abb. 5.9 dargestellt. Bei der schlechteren Auflösung $a = 1 \mu\text{m}$ weichen die Resultate für die beiden Modelle \mathbb{S}^{FP} und \mathbb{S}^{PP} stark voneinander ab. Bei höherer Auflösung nähern sich die Ergebnisse des \mathbb{S}^{FP} -Modells an die des \mathbb{S}^{PP} -Modells an. Bei der höchsten Auflösung $a = 0.25 \mu\text{m}$ konnten nur Berechnungen für Messzellenlängen $L = 10 \mu\text{m}$, $20 \mu\text{m}$ durchgeführt werden.

5.6.1.5 Permeabilitätsberechnung durch Gitter-Boltzmann-Simulationen

Zur Bestimmung der Permeabilitäten der diskretisierten Modelle \mathbb{S}^{FP} und \mathbb{S}^{PP} wurden Gitter-Boltzmann-Simulationen durchgeführt, siehe [60] für Details. Die auflösungsabhängigen Resultate für die Permeabilitäten $\kappa(a)$ sind in Tabelle 5.2 zu sehen. Bei Verbesserung der Auflösung von $a = 1 \mu\text{m}$ zu $a = 0.5 \mu\text{m}$ ändert sich die Permeabilität des \mathbb{S}^{PP} -Modells um ca. 6 %, wohingegen die Permeabilität des \mathbb{S}^{FP} -Modells sich um ca. 9 % ändert.

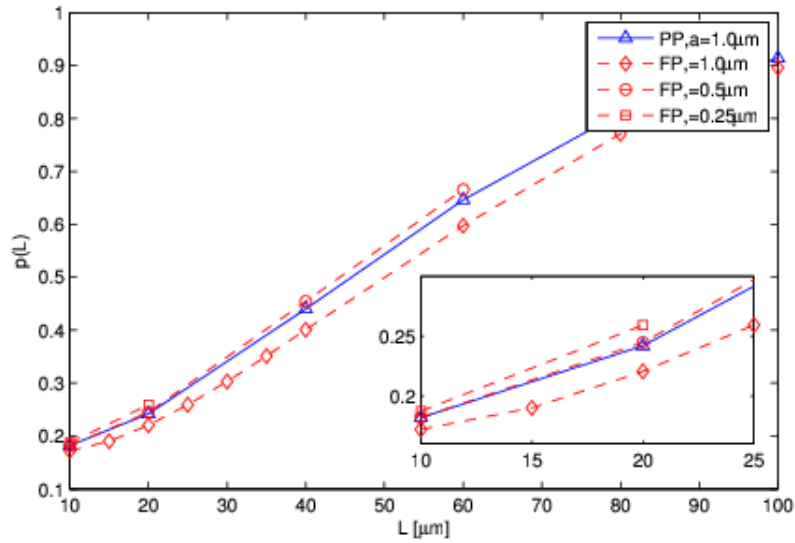


Abbildung 5.9: Gesamtanteil perkolierender Messzellen $p(L)$ mit Seitenlängen L für das \mathbb{S}^{FP} - und \mathbb{S}^{PP} -Modell mit den Auflösungen $a = 0.25 \mu\text{m}$, $0.5 \mu\text{m}$, $1.0 \mu\text{m}$.

	$a [\mu\text{m}]$	$\kappa(a) [\text{mD}]$
\mathbb{S}^{PP}	1.0	660.2
\mathbb{S}^{PP}	0.5	617.7
\mathbb{S}^{FP}	1.0	200.7
\mathbb{S}^{FP}	0.5	182.5

Tabelle 5.2: Permeabilitäten $\kappa(a)$ berechnet aus den Gitter-Boltzmann-Simulationen. Es wurden Datensätze mit verschiedenen Auflösungen a jeweils von dem Modell \mathbb{S}^{FP} bzw. \mathbb{S}^{PP} verwendet.

5.6.2 Kugelpackung mit rauen Porenwänden

Diese Anwendung zeigt, wie mit dem Algorithmus und geeigneten Protokollen in einem mehrschrittigen Verfahren komplexe Modelle erzeugt werden können. Fast 6000 große Kugeln mit Radien im Bereich $r_i \in [10.0 \mu\text{m}, 100.0 \mu\text{m}]$ mit $i = 0, \dots, 5.986$ bilden eine primäre Matrix, deren Porenwänden von fast 2 Millionen sehr viel kleineren Kugeln mit Radien $r_i \in [0.8 \mu\text{m}, 1.5 \mu\text{m}]$ mit $i = 5.987, \dots, 1.937.090$ bedeckt wird, siehe Abb. 5.10. Das finale Modell hat eine Porosität von ca. 9 %.

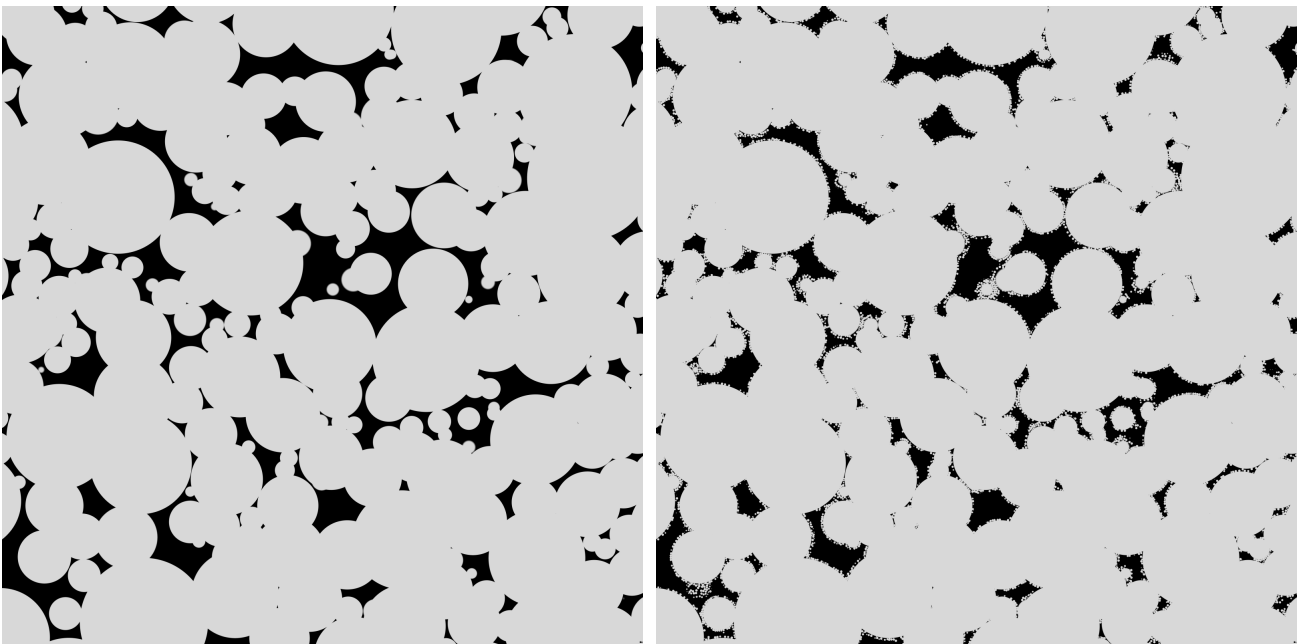


Abbildung 5.10: Querschnitt durch die Kugelpackung, die raue Oberflächen modelliert. Die Auflösung ist $a = 0.5 \mu\text{m}$ und dargestellt sind quadratische Querschnitte der Seitenlängen $L = 1600 \mu\text{m}$. Links in Grau die großen Kugeln der festen Matrix mit Radien im Bereich $[10 \mu\text{m}, 100 \mu\text{m}]$. In der rechten Abbildung bedecken zusätzlich kleine Kugeln mit Radien im Bereich $[0.9 \mu\text{m}, 1.1 \mu\text{m}]$ die Oberflächen der Porenwände. Das gesamte 3D-Modell besteht aus ca. 6000 großen Kugeln und ca. 1.9 Millionen kleinen Kugeln.

5.6.2.1 Simulationsparameter und Protokolle

Die Simulation besteht aus ca. 1.9 Mio. Kugeln in einem Gebiet der Größe $l_1^d = l_2^d = l_3^d = 1000 \mu\text{m}$. Der primäre Porenraum besteht aus Kugeln des Typs $t_i = 1$ für $i = 1, \dots, 5.985$ mit Radien $r_i \in [10 \mu\text{m}, 100 \mu\text{m}]$. Das Radiusprotokoll, welches hier bereits zur Generierung des primären Porenraums angewandt wird, trägt maßgeblich zum Erreichen der geringen primären Porosität (ca. 11 %) bei, greift jedoch stark in die Verteilung der Kugelgrößen ein, siehe Abb. 5.11. Die Konfiguration wird eingefroren, d. h. dass die Kugeln des Typs $t_i = 1$ in der weiteren Simulation ihre Position nicht mehr ändern. Im zweiten Simulationsabschnitt wird das gesamte Systemgebiet mit Kugeln des Typs $t_i = 2$ für $i = 5.987, \dots, 1.937.090$ der Größenordnung $r_i \in [0.8 \mu\text{m}, 1.5 \mu\text{m}]$ stochastisch aufgefüllt. Das verwendete Protokoll führt dazu, dass die kleinen Kugeln rasch an die Oberflächen wandern, sich dort verteilen und damit raue Porenwände erzeugen.

1. Simulationsparameter zur Generierung des primären Porenraums:

- Die Größe der Nachbarschaftszellen ist $l_1^c = l_2^c = l_3^c = 100 \mu\text{m}$.
- Der Zeitschritt ist $dt = 0.5$. Es wurden $S = 100$ Zeitschritte benötigt.
- Exponentiale Radiusverteilung mit $\alpha = 0.05$, $r_i \in [10 \mu\text{m}, 100 \mu\text{m}]$ für $i = 0, \dots, 5.985$.
- Wechselwirkungskonstanten für Kugeltypen $t_i = 1$ sind $k_1^0 = 350.0$ und $k_1^1 = 350.0$ mit Protokoll $S_1^{K^0} = S_1^{K^1} = 1$. Die Dämpfung ist $\Xi_1^0 = \Xi_1^1 = 0.10$ mit $S_1^{\Xi^0} = S_1^{\Xi^1} = 1$.
- Für die maximale zugelassene Überlappung gilt $\sigma_1^{max} = 0.35$.
- Das Protokoll der Radiusveränderung für den Kugeltyp $t_i = 1$ ist $n_{rchg} = 0.2$, $n_r = 0.01$, $t_{rchg}^0 = 55$ sowie $t_{rchg}^1 = 85$. Ab dem 55. Iterationsschritt werden die Radien sehr aggressiv geändert. Bereits wenn weniger als 20 % der Kugeln pro Zeitschritt konvergieren, beginnt die Anpassung. Aber das Protokoll erlaubt nur eine vergleichsweise geringe Radiusänderung von 1 % pro Iterationsschritt. Es wird also sehr oft die Möglichkeit einer Radiusänderung zugelassen, aber nur in geringem Maße.
- Konvergierte Punkte werden eingefroren für $t^s \geq t_{lock} = 1$, also frühestmöglich bei Erreichen einer akzeptablen Überlappung.

2. Simulationsparameter zur Bedeckung der Porenwände:

- Die anfängliche Radiusverteilung ist $r_i = r_2^{max} = 1.5 \mu\text{m}$ für $i = 5.986, \dots, 1.937.090$.
- Wechselwirkungskonstanten für den ersten Kugeltyp sind $k_1^0 = 1000.0$ und $k_1^1 = 500.0$ gesteuert durch das Protokoll $S_1^{k^0} = 1$ und $S_1^{k^1} = 10$. Die Dämpfung ist $\Xi_1^0 = \Xi_1^1 = 1.0$ mit dem Protokoll $S_1^{\Xi^0} = S_1^{\Xi^1} = 1$. Für den zweiten Kugeltyp gilt $k_2^0 = 1.0$, $k_2^1 = 40.0$ mit $S_2^{k^0} = 10$ und $S_2^{k^1} = 20$ und die Dämpfung ist $\Xi_2^0 = 1.0$ sowie $\Xi_2^1 = 2.0$ mit $S_1^{\Xi^0} = 60$ und $S_1^{\Xi^1} = 70$. Es werden anfangs vor allem die kleinen Kugeln des Typs $t_i = 2$ zu den Oberflächen der viel größeren Kugeln des Typs $t_i = 1$ hingezogen. Die gegenseitige Wechselwirkung der kleinen Kugeln und die sehr spät angeschaltete Dämpfung sorgen für einen eher gleichmäßigen Überzug der Porenwände.
- Für die maximale Überlappung gilt $\sigma_2^{max} = 0.2$, was einem sehr geringen gültigen Überlappungsbereich entspricht. Die kleinen Kugeln berühren die Oberfläche der großen Kugeln und sollen nicht darin 'einsinken'.
- Es wird ein Protokoll verwendet, das es den kleinen Kugeln des Typs $t_i = 2$ erlaubt, ihren Radius zu ändern, um eine akzeptierte Überlappung zu erreichen. Die verwendeten Parameter sind $n_{rchg} = 0.2$, $n_r = 0.1$, $t_{rchg}^0 = 60$ sowie $t_{rchg}^1 = 70$. Das Protokoll ist zusammen mit der Zunahme der Dämpfung aktiv.
- Einfrieren der konvergierten Kugeln wird verwendet für $t^s \geq t_{lock} = 20$. Das ist der Iterationsschritt, ab dem die maximale gegenseitige Wechselwirkung erreicht ist.

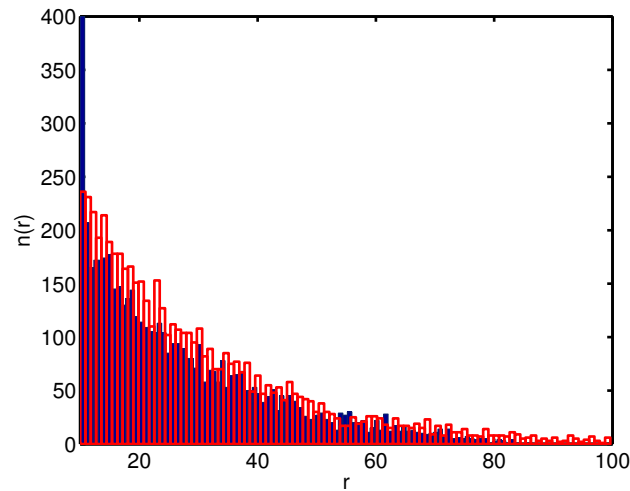


Abbildung 5.11: Histogramm der Radienverteilung für die großen Kugeln mit Typ $t_i = 1$ mit der Auflösung $dr = 1.0 \mu\text{m}$ am Anfang (unausgefüllt, rot) und am Ende der Simulation (ausgefüllt, blau). Durch das Radienprotokoll haben die roten Kugeln ihren Radius bis zum Minimalradius $r = 10.0 \mu\text{m}$ verringert.

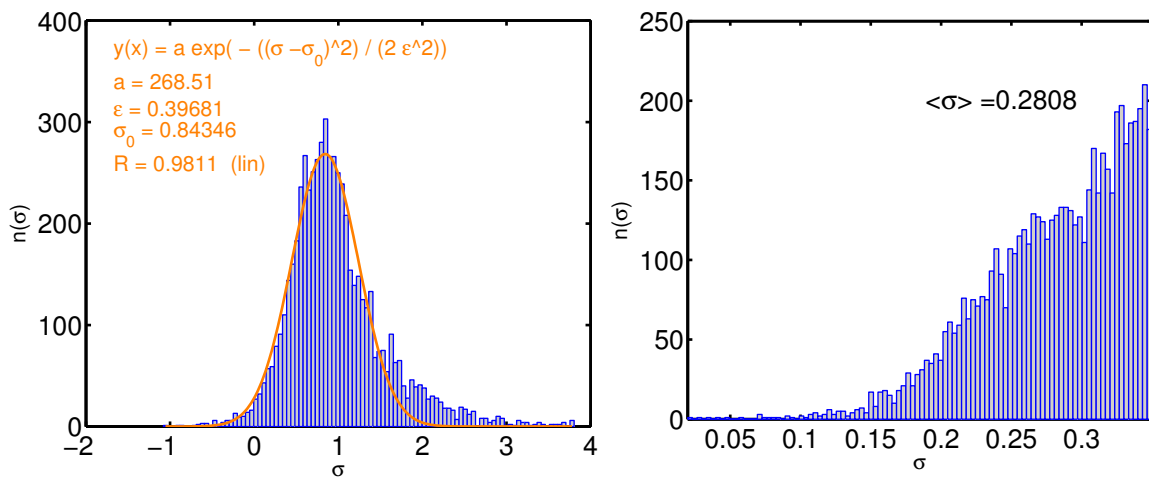


Abbildung 5.12: Histogramme der Überlappung σ für den Kugeltyp $t_i = 1$ zum Beginn der Simulation (links) und am Ende der Simulation (rechts). Für das Anfangshistogramm ist eine Normalverteilung mit den angegebenen Parametern angepasst worden. Am Ende der Simulation haben alle Kugeln eine erlaubte Überlappung $\sigma \leq 0.35$. Die mittlere Überlappung am Ende ist $\langle \sigma \rangle = 0.28$.

5.6.3 Große monodisperse Kugelpackung

In dieser Anwendung wurde die Effizienz der parallelen DEM-Implementierung validiert. Ziel war es, eine möglichst große, fast monodisperse Kugelpackung zu erzeugen. Auch wurde der Datensatz genutzt, um verschiedene bildgebende Renderverfahren zu testen und weiterzuentwickeln, siehe [31]. Dazu wurden Datensätze bestehend aus bis zu 100.000.000 Quarzkristallen verwendet. Bei der Erzeugung wurde eine relativ große maximale Überlappung akzeptiert, da es vor allem um die Evaluierung der technischen Machbarkeit mit dem DEM-Algorithmus ging. Der effektivste 3D-Renderalgorithmus wurde in Kooperation mit dem VISUS-Institut der Universität Stuttgart unter Leitung von Herrn Grottel in dem Programmpaket 'MegaMol' umgesetzt.

5.7 Fazit

Der hier vorgestellte Algorithmus mit seinen diversen Protokollmöglichkeiten ist nach bestem Wissen die derzeit einzige Implementation, die eine adaptierte Diskrete-Elemente-Simulation verwendet, um Kugelpackungen zu generieren speziell zur Weiterverwendung im Modellierungsprozess für kontinuierliche Kornmodelle eines porösen Mediums. Die parallele Implementierung ist hocheffizient und wurde auf einigen zur Verfügung stehenden Großrechnern getestet. Er übertrifft die Limitationen des bisherig verwendeten stochastischen Algorithmus und erlaubt es, größere und stark polydisperse Kugelpackungen zu generieren. Dadurch können zukünftig sehr viel realistischere Kornmodelle erzeugt werden. Ein Beispiel dafür wird weiter unten im Kapitel 8 vorgestellt.

6 Permeabilitätsberechnung durch Gitter-Boltzmann-Simulationen

Wie in dem einführenden Kapitel 4.4 zur Definition der Permeabilität beschrieben, kann aus einem gegebenen Geschwindigkeitsfeld $\mathbf{v}(\mathbf{x})$ und Druckfeld $p(\mathbf{x})$ im Porenraum mit $\mathbf{x} \in \mathbb{P}$, einer geeigneten Mittelung und der Kenntnis der dynamischen Viskosität des Fluides μ die Permeabilität des Mediums berechnet werden, siehe 4.4.1. Eine Möglichkeit, die Felder $\mathbf{v}(\mathbf{x})$ und $p(\mathbf{x})$ zu erhalten, ist es, eine Probe kubisch binär zu diskretisieren, also einen 3D-Datensatz aus kubischen Zellen zu generieren, in dem jeder Voxel entweder Porenraum oder Matrix ist. Innerhalb dieses diskretisierten Porenraums lässt sich dann eine diskretisierte Boltzmann-Gleichung lösen und erzeugt dadurch die Felder $\mathbf{v}(\mathbf{x})$ und $p(\mathbf{x})$. Diese Simulationsmethode wird Gitter-Boltzmann-Simulation genannt. Sie hat sich im Bereich poröser Medien etabliert und wird weiterentwickelt [17, 60, 43, 79, 34]. Als Alternative sei erwähnt, dass die Strömung auch durch andere Gleichungen wie z. B. die Stokes-Gleichung beschrieben werden kann und diese dann zum Beispiel mithilfe eines Finite-Differenzen-Verfahrens gelöst wird, siehe [57, 54]. Auch ist es denkbar, den gesamten Porenraum und dessen Oberfläche mittels einer unregelmäßigen Vernetzung zu diskretisieren und dann auf diesem Netz eine Strömungsgleichung zu lösen [64]. Es ist zu erwarten, dass diese Art der Vernetzung mindestens annähernd so aufwendig ist wie die eigentliche Lösung der Strömungsgleichungen. Das macht Methoden mit einer einfachen Vernetzung wie die Gitter-Boltzmann-Simulation für diese Art von Geometrie so attraktiv. Der Diskretisierungsaufwand zur Erzeugung eines binären Voxel-Datensatzes ist deutlich geringer. Näheres zur Diskretisierung und Segmentierung in Kapitel 4.7.2.

6.1 Boltzmann-Gleichung und Gitter-Boltzmann-Simulationen

Die Boltzmann-Gleichung

$$\frac{\partial}{\partial t} n(\mathbf{x}, \mathbf{c}, t) + \mathbf{c} \cdot \nabla n(\mathbf{x}, \mathbf{c}, t) = \Omega(n(\mathbf{x}, \mathbf{c}, t)) \quad (6.1.1)$$

mit dem Ort $\mathbf{x} \in \mathbb{R}^3$, dem Geschwindigkeitsvektor $\mathbf{c} \in \mathbb{R}^3$, dem Zeitpunkt $t \in \mathbb{R}$ und dem Kollisionsoperator $\Omega(n(\mathbf{x}, \mathbf{c}, t))$ beschreibt die zeitliche Entwicklung der Einteilchen-Wahrscheinlichkeitsdichte $n(\mathbf{x}, \mathbf{c}, t)$. In Gitter-Boltzmann-Simulationen wird diese Gleichung diskretisiert. Die hier verwendete Implementierung nutzt ein strukturiertes kubisches Gitter mit der Gitterkonstante Δx als Ortsdiskretisierung und dem Zeitintervall Δt . Auch der Geschwindigkeitsraum ist diskretisiert durch eine endliche Menge an möglichen Gittergeschwindigkeitsvektoren \mathbf{c}_i mit $i = 1, \dots, N_v$. Die Anzahl und Ausrichtung der möglichen Geschwindigkeiten variieren zwischen den verschiedenen Gitter-Boltzmann-Implementierungen. Die hier verwendete Implementierung ist eine 'D3Q19'-Variante, das bedeutet eine 3D-Implementierung mit $N_v = 19$ diskreten Geschwindigkeiten [65], siehe Abb. 6.1.

Ein kubisches Gitter mit Basisvektoren $\mathbf{e}_k \in \mathbb{R}^3$, $k = 1, 2, 3$ und der Gitterfunktion $\mathbf{g} : \mathbb{N}^3 \mapsto \mathbb{R}^3$, welche die Gitterknoten $\boldsymbol{\ell} \in \mathbb{N}^3$ den Ortsvektoren $\mathbf{g}(\boldsymbol{\ell}) \in \mathbb{R}^3$ zuordnet, wird verwendet. Das Simulationsgebiet ist ein rechtwinkliges Parallelepipet, siehe Abb. 6.2,

$$\mathcal{L} = \{\boldsymbol{\ell} \in \mathbb{N}^3 : 1 \leq \ell_k \leq L_k; k = 1, 2, 3\}, \quad (6.1.2)$$

mit den dimensionslosen Seitenlängen $L_k \in \mathbb{N}^3$. Physikalische Größen w , z. B. Druck und Dichte, die für Gitterknoten definiert sind, werden wie folgend abgekürzt: $w(\boldsymbol{\ell}) = w(\mathbf{g}(\boldsymbol{\ell}))$. Die Vektoren $\mathbf{n}(\boldsymbol{\ell}, t) = (n_1(\boldsymbol{\ell}, t), \dots, n_{N_v}(\boldsymbol{\ell}, t))$, mit Komponenten

$$n_i(\boldsymbol{\ell}, t) = \int_{\mathbb{W}(\boldsymbol{\ell})} \int_{\mathbb{B}(i)} n(\mathbf{x}, \mathbf{c}, t) d\mathbf{c} d\mathbf{x}. \quad (6.1.3)$$

stellen Wahrscheinlichkeiten dar.

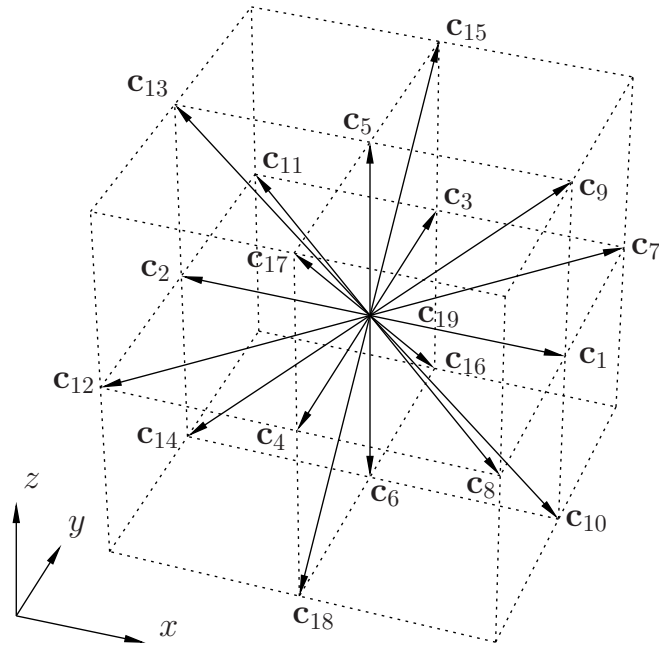


Abbildung 6.1: Diskretisierung des Geschwindigkeitsraums in die Richtungen c_i mit $i = 1, \dots, 19$ bei der D3Q19-Gitter-Boltzmann-Variante.

Darin ist $\mathbb{W}(\ell) \subset \mathbb{R}^3$ das Zellgebiet um den Punkt $\mathbf{g}(\ell)$ und $\mathbb{B}(i) \subset \mathbb{R}^3$ ein kegelförmiges Gebiet im Geschwindigkeitsraum mit Spitze im Ursprung und Richtung \mathbf{c}_i . Für die makroskopische Dichte $\rho(\ell, t)$ und Geschwindigkeit $\mathbf{v}(\ell, t)$ gilt für gegebenes \mathbf{n}

$$\rho(\ell, t) = \rho^\circ \sum_{i=1}^{N_v} n_i(\ell, t), \quad (6.1.4)$$

$$\mathbf{v}(\ell, t) = \sum_{i=1}^{N_v} n_i(\ell, t) \mathbf{c}_i / \sum_{i=1}^{N_v} n_i(\ell, t), \quad (6.1.5)$$

mit ρ° , der Referenzdichte. Der Druck ist definiert durch

$$p(\ell, t) = c_s^2 \rho(\ell, t), \quad (6.1.6)$$

worin

$$c_s = \frac{1}{\sqrt{3}} \left(\frac{\Delta x}{\Delta t} \right). \quad (6.1.7)$$

die Schallgeschwindigkeit ist, siehe [73, 66].

Durch Diskretisierung der Gleichung (6.1.1) ergibt sich das System der Differenzgleichungen

$$n_i(\boldsymbol{\ell} + \Delta \boldsymbol{\ell}_i, t + \Delta t) - n_i(\boldsymbol{\ell}, t) = \Omega_i(\boldsymbol{\ell}, t) \cdot \Delta t, \quad (6.1.8)$$

mit $\Delta \boldsymbol{\ell}_i = \mathbf{c}_i \Delta t / \Delta x$ und den Anfangsbedingungen $n_i(\boldsymbol{\ell}, 0) = 1/N_v$ (für $t = 0$). Der im allgemeinen nichtlineare Kollisionsoperator wird hier in einer Linearisierung verwendet

$$\Omega_i(\boldsymbol{\ell}, t) = \sum_{j=1}^{N_v} S_{ij} \cdot n_j(\boldsymbol{\ell}, t) - n_j^{\text{eq}}(\boldsymbol{\ell}, t), \quad (6.1.9)$$

mit lokalen Gleichgewichts-Wahrscheinlichkeitsfunktionen

$$\mathbf{n}^{\text{eq}}(\boldsymbol{\ell}, t) = (n_1^{\text{eq}}(\boldsymbol{\ell}, t), \dots, n_{N_v}^{\text{eq}}(\boldsymbol{\ell}, t))$$

und der $N_v \times N_v$ Kollisionsmatrix \mathbf{S} [74, 48].

Beim 'Single relaxation time' (SRT), auch genannt LB-BGK, Ansatz [13, 7] wird der Kollisionsoperator definiert über eine einzige Relaxationskonstante τ , wie

$$S_{ij} = -\frac{1}{\tau} \delta_{ij}, \quad (6.1.10)$$

worin δ_{ij} das Kronecker Delta ist. Bei dieser Methode wird $\mathbf{n}^{\text{eq}}(\boldsymbol{\ell}, t)$ durch eine Taylor-Entwicklung zweiter Ordnung der Maxwell-Verteilung approximiert [33],

$$n_i^{\text{eq}}(\boldsymbol{\ell}, t) = \frac{\rho \omega_i}{\rho^o} \left(1 + \frac{\mathbf{v}^{\text{eq}} \cdot \mathbf{c}_i}{c_s^2} + \frac{(\mathbf{v}^{\text{eq}} \cdot \mathbf{c}_i)^2}{2c_s^4} - \frac{\mathbf{v}^{\text{eq}} \cdot \mathbf{v}^{\text{eq}}}{2c_s^2} \right). \quad (6.1.11)$$

In Abwesenheit externer Kräfte ist die Gleichgewichtsgeschwindigkeit $\mathbf{v}^{\text{eq}}(\boldsymbol{\ell}, t) = \mathbf{v}(\boldsymbol{\ell}, t)$, siehe Gleichung (6.1.5). Wie weiter unten erklärt wird, können sich $\mathbf{v}^{\text{eq}}(\boldsymbol{\ell}, t)$ und $\mathbf{v}(\boldsymbol{\ell}, t)$ in Gleichung (6.1.5) unterscheiden, falls externe Kräfte auftreten. Die Gittergewichte ω_i sind abhängig vom Gittertyp, von der Anzahl der räumlichen Dimensionen sowie Anzahl und Orientierung der diskretisierten Geschwindigkeitsvektoren. Für eine ausführliche Darstellung der verschiedenen Gittertypen und Diskre-

tisierungen siehe [17].

Ein komplexerer Ansatz für den Kollisionsoperator wird 'multi relaxation time method' (MRT) genannt. Darin wird eine lineare Transformation \mathbf{M} derart gewählt, dass die Momente

$$m_i(\boldsymbol{\ell}, t) = \sum_j^{N_v} M_{ij} \cdot n_j(\boldsymbol{\ell}, t) \quad (6.1.12)$$

hydrodynamischen Moden des Problems entsprechen. In der hier verwendeten Implementation werden die Transformationen aus [18] genutzt. Darin ist $m_1(\boldsymbol{\ell}, t)$ die Dichte definiert in Gl. (6.1.4), $m_2(\boldsymbol{\ell}, t)$ ist die Energie, $m_i(\boldsymbol{\ell}, t)$ mit $i = 4, 6, 8$ sind die Impulsflüsse und $m_i(\boldsymbol{\ell}, t)$ mit $i = 10, 12, 14, 15, 16$ sind Komponenten des symmetrischen spurlosen Spannungstensors. Mit dem Impulsvektor $\mathbf{m}(\boldsymbol{\ell}, t) = (m_1(\boldsymbol{\ell}, t), \dots, m_{N_v}(\boldsymbol{\ell}, t))$, dem Kollisionsvektor $\boldsymbol{\Omega}(\boldsymbol{\ell}, t) = (\Omega_1(\boldsymbol{\ell}, t), \dots, \Omega_{N_v}(\boldsymbol{\ell}, t))$, der diagonalen Matrix $\check{S}_{ij} = \check{s}_i \delta_{ij}$ und dem Gleichgewichtsimpulsvektor $\mathbf{m}^{\text{eq}}(\boldsymbol{\ell}, t) = (m_1^{\text{eq}}(\boldsymbol{\ell}, t), \dots, m_{N_v}^{\text{eq}}(\boldsymbol{\ell}, t))$, gilt

$$\boldsymbol{\Omega}(\boldsymbol{\ell}, t) = -\mathbf{M}^{-1} \cdot \check{\mathbf{S}} \cdot (\mathbf{m}(\boldsymbol{\ell}, t) - \mathbf{m}^{\text{eq}}(\boldsymbol{\ell}, t)). \quad (6.1.13)$$

Während des Kollisionsschrittes werden die Dichte und der Impuls erhalten, es gilt also $m_1^{\text{eq}}(\boldsymbol{\ell}, t) = m_1(\boldsymbol{\ell}, t)$ und $m_i(\boldsymbol{\ell}, t) = m_i^{\text{eq}}(\boldsymbol{\ell}, t)$ mit $i = 4, 6, 8$. Es wird angenommen, dass die nicht erhaltenen Gleichgewichtsmomente $m_i^{\text{eq}}(\boldsymbol{\ell}, t)$, $i \neq 1, 4, 6, 8$ Funktionen dieser erhaltenen Momente sind, siehe [18]. Das Diagonalelement $\tau_i = 1/\check{s}_i$ der Kollisionsmatrix ist die Relaxationszeit des Moments $m_i(\boldsymbol{\ell}, t)$. Es gilt $\check{s}_1 = \check{s}_4 = \check{s}_6 = \check{s}_8 = 0$, weil die entsprechenden Momente erhalten werden, $\check{s}_2 = 1/\tau_{\text{bulk}}$ ist die Relaxation der Energie und $\check{s}_{10} = \check{s}_{12} = \check{s}_{14} = \check{s}_{15} = \check{s}_{16} = 1/\tau$ ist Relaxation der Elemente des Spannungstensors. Die verbleibenden Diagonalelemente $\check{\mathbf{S}}$ sind frei gewählt zu

$$\check{\mathbf{S}} = \text{diag}(0, 1/\tau_{\text{bulk}}, 1.4, 0, 1.2, 0, 1.2, 0, 1.2, 1/\tau, \\ 1.4, 1/\tau, 1.4, 1/\tau, 1/\tau, 1/\tau, 1.98, 1.98, 1.98), \quad (6.1.14)$$

und dienen der Verbesserung der Konvergenz des Algorithmus, siehe [49, 18]. In diesem Ansatz bleiben zwei freie Parameter, τ und τ_{bulk} . Diese Multirelaxationsmethode ist daher auch bekannt als 'two relaxation time' (TRT) Methode. Andere Ansätze der TRT werden in [28, 29, 22] diskutiert.

Um die Gitter-Boltzmann-Methode auf Strömungsprobleme poröser Medien anwenden zu können, ist es nötig, den Zusammenhang zwischen diesen Lösungen der Boltzmann-Gleichungen und den Lösungen der Navier-Stokes (NS) Gleichung zu untersuchen. Es existiert ein Verfahren, das zeigt, dass

unter speziellen Bedingungen, Lösungen der Boltzmann-Gleichung ebenfalls Lösungen der Navier-Stokes Gleichung sind. Diese Chapman-Enskog-Entwicklung zeigt, dass die Dichte, Geschwindigkeit und der Druck die NS-Gleichung erfüllen, wenn keine externen Kräfte auftreten und für die kinematische Viskosität gilt,

$$\nu(\tau, \Delta t) = c_s^2 \Delta t \left(\frac{\tau}{\Delta t} - \frac{1}{2} \right). \quad (6.1.15)$$

Für Details siehe [12, 77, 33, 63, 50].

Mit den Gleichungen (6.1.15) und (6.1.7) gilt

$$\frac{\tau}{\Delta t} = \frac{1}{2} + \frac{\sqrt{3}\nu}{c_s \Delta x} = \frac{1}{2} + 3 \frac{\nu \Delta t}{(\Delta x)^2} \quad (6.1.16)$$

und weil $\nu \geq 0$, $\Delta x > 0$ sowie $\Delta t > 0$, muss immer gelten $\tau/\Delta t \geq 1/2$.

6.1.1 Stoffeigenschaften bei der Anwendung in porösen Medien

Ein realistischer Porendurchmesser in porösem Gestein ist $a \approx 10^{-5}$ m. Die kinetische Viskosität für Wasser ist $\nu \approx 10^{-6} \text{ m}^2 \text{ s}^{-1}$ und die Schallgeschwindigkeit ist $c_s \approx 10^3 \text{ m s}^{-1}$. Mit typischen Geschwindigkeiten $v \approx 10^{-4} \text{ m s}^{-1}$ gilt für die Reynolds-Zahl $\text{Re} = v a / \nu \approx 10^{-3}$. Bei einer Diskretisierung von $\Delta x = 10^{-6}$ m gilt dann, nach Gl. 6.1.16, $\tau/\Delta t = 0.5017$. Es ist jedoch bekannt, dass die LB-Methode für $\tau/\Delta t \approx 1/2$ instabil ist. Eine direkte Simulation mit diesen Parametern ist also nicht sinnvoll. Man könnte festlegen, dass $\tau/\Delta t = 1$ und die Stoffeigenschaften ν und c_s wie oben beibehalten werden. Für die Diskretisierung gilt dann $\Delta t \approx 10^{-12}$ s und $\Delta x \approx 10^{-9}$ m. Mit realistischen Porendurchmessern von $a \approx 10^{-5}$ m ist diese Raumdiskretisierung nicht effizient oder gar technisch unmöglich ist. Diese Arbeit verwendet ein Argument der hydrodynamischen Ähnlichkeit. Es werden Stoffeigenschaften eines Pseudofluides angenommen, welches die kinematische Viskosität von Wasser hat, aber die Schallgeschwindigkeit $c_s = 1 \text{ m s}^{-1}$. Als Diskretisierungsparameter ergibt sich dann $\Delta x = 10^{-6}$ m und $\Delta t = 10^{-6}$ s. Eine typische Pore ist dann im Durchmesser durch 10 Knoten repräsentiert, und ein kubisches Systemgebiet mit Seitenlängen 10^{-3} m ist diskretisiert durch 1000^3 Knoten. Beim Aufbringen einer externen Kraft, wie im nächsten Abschnitt besprochen, erhöhen sich die Geschwindigkeiten auf Größenordnungen 10^{-3} m s^{-1} . Realistische Mach-Zahlen sind $\text{Ma} \approx 10^{-3}$ und Reynolds-Zahlen $\text{Re} \approx 10^{-3}$. Es handelt sich also um eine laminare Unterschallströmung. Solange $\text{Ma} \ll 1$, gilt das Argument der hydrodynamischen Ähnlichkeit.

6.1.2 Externe Kräfte bei Strömungssimulationen in porösen Medien

Um den Fluss durch das poröse Medium zu treiben, können lokale Druckgradienten verwendet werden oder es kann eine Körperkraft angelegt werden [41, 56, 81]. In den Strömungssimulationen in dieser Arbeit wird der Fluss durch eine externe Körperkraft in speziellen Vor- und Nachlaufbecken realisiert, siehe in Abschnitt 6.1.4.

Um diese externe Körperkraft $\mathbf{b}(\ell, t)$ mit einzubeziehen, wird ein Kraftterm, siehe [48],

$$\varphi_i(\ell, t) = \Delta t \cdot \frac{\rho \omega_i}{\rho^o} \left(h_0 + \frac{\mathbf{h}_1 \cdot \mathbf{c}_i}{c_s^2} + \frac{\mathbf{h}_2 : (\mathbf{c}_i \mathbf{c}_i - c_s^2 \mathbf{I})}{2c_s^4} \right), \quad (6.1.17)$$

mit $i = 1, \dots, N_v$ zur rechten Seite der Gleichung (6.1.8) addiert. Die Parameter der Ordnungen 0, 1 und 2, die in der Entwicklung vorkommen, sind h_0 , \mathbf{h}_1 und \mathbf{h}_2 . Dadurch wird die Beschleunigung im Propagationsschritt durchgeführt. Die Definitionen der Entwicklungsparameter und der Geschwindigkeiten $\mathbf{v}^{eq}(\ell, t)$ und $\mathbf{v}(\ell, t)$ unterscheiden sich in den verschiedenen Methoden, welche ausführlich in [60] besprochen werden. Immer gilt $h_0 = 0$ da sonst zusätzliche Quellterme in der Massenbilanz auftreten würden. In dieser Arbeit wird die folgende Methode verwendet. Darin ist

$$\mathbf{h}_1 = \mathbf{b}, \quad \mathbf{h}_2 = \mathbf{0}, \quad (6.1.18)$$

so dass $\varphi_i(\ell)$ nicht von $\mathbf{v}^{eq}(\ell, t)$ abhängig ist.

Weiterhin muss die Gleichungen (6.1.5) angepasst werden und eine eigene Definition für die Gleichgewichtsgeschwindigkeiten \mathbf{v}^{eq} in Gleichung (6.1.11) gelten. In abgekürzter Schreibweise gilt, wie bisher,

$$\mathbf{v}^{eq} = \sum n_i \mathbf{c}_i / \sum n_i, \quad (6.1.19)$$

aber die makroskopische Strömungsgeschwindigkeit ist definiert durch

$$\mathbf{v} = \mathbf{v}^{eq} + \Delta t \mathbf{b} / 2 \quad (6.1.20)$$

Dieser Ansatz ist effektiver in der Berechnung, darf aber nur bei der Berechnung stationärer Flüsse benutzt werden. Das ist hier keine Einschränkung, da in dieser Arbeit der Algorithmus nur zur Berechnung von Permeabilität genutzt werden soll und dabei nur das stationäre Ergebnis von Interesse ist. Die makroskopischen Ergebnisse für Dichte, Druck und Geschwindigkeit erfüllen die Massenbi-

lanz, aber es treten unphysikalische Terme in der Impulsbilanz auf [32]. Diese Terme verschwinden oder sind sehr klein in dieser Anwendung, da die räumlichen Gradienten nicht groß sind.

6.1.3 Behandlung der Phasengrenzen bei Simulationen in porösen Medien

Die am häufigsten verwendeten Randbedingungen bei Gitter-Boltzmann-Simulationen sind periodische Randbedingungen (PBC) an den Grenzen des Simulationsgebiets und 'no-slip'-Randbedingungen was verschwindenden Tangentialgeschwindigkeiten an den Phasengrenzen entspricht [73]. Diese Art von Verhalten ist beim Fluss mit kleinen Reynolds-Zahlen sicher sinnvoll. Die 'no-slip'-Randbedingung wird auch 'simple bounce back' (SBB) Randbedingung genannt. Wenn der Gitterpunkt $\ell + \Delta\ell_i$ in der Gleichung (6.1.8) in der festen Phase des porösen Mediums liegt, wird die diskrete LB-Gleichung umgeschrieben zu

$$n_i^*(\ell, t + \Delta t) - n_i(\ell, t) = \Omega_i(\ell, t) \cdot \Delta t, \quad (6.1.21)$$

worin die Wahrscheinlichkeitsfunktion n_i^* ist, welche sich auf $\mathbf{c}_i^* = -\mathbf{c}_i$ bezieht. Implementierungen die auf der SBB-Methode beruhen sind abhängig von der Viskosität und der Relaxationszeit τ [35]. Die numerische, also die von dem Algorithmus wahrgenommene Position der Phasengrenze hängt geringfügig von τ ab. Bei Strömungssimulationen in porösen Medien kann das einen starken Einfluss haben, falls die engen, aber für den Gesamtfluss im Sinne der Konnektivität wichtigen Porenhäse nur von einigen wenigen Gitterpunkten repräsentiert werden. Die berechnete Permeabilität κ , eine Materialkonstante des porösen Mediums, wäre dann effektiv von der Viskosität des Fluides abhängig. Diese unerwünschte $\kappa - \tau$ Korrelation ist wesentlich stärker bei LB-BGK-Implementierungen als bei den LB-MRT-Varianten [26, 18]. In der Literatur werden auch verbesserte 'no-slip' Randbedingungen diskutiert [63, 51, 11, 27], um diese Artefakte zu minimieren. Viele dieser Methoden beruhen auf Interpolationen an der Phasengrenze und sind daher bei Porenhäsen, die nur durch einige wenige Gitterpunkte aufgelöst sind, nicht vielversprechend - also genau in den Fällen wo sie von Nöten wären. Daher werden in den Simulationen dieser Arbeit die einfachen und bedienungstechnisch effektiven SBB verwendet.

6.1.4 Aufbau der Strömungssimulation

Das Simulationsgebiet \mathcal{L} , siehe Abb. 6.2, ist in drei Teilgebiete unterteilt. Das porösen Material \mathcal{S} , auch Probe genannt, sowie einen Vorlauf sowie Nachlauf \mathcal{I} ('Inlet') und \mathcal{O} ('Outlet'). Die Menge

$$\mathcal{C}(a) := \{\ell \in \mathcal{L} : \ell_3 = a\} \tag{6.1.22}$$

beschreibt einen horizontalen Querschnitt. Damit ist $\mathcal{C}(L_{\mathcal{I}}) \in \mathcal{I}$ der Querschnitt direkt vor der Probe, $\mathcal{C}(L_{\mathcal{I}}+1) \in \mathcal{S}$ ist der erste Querschnitt in Flussrichtung in der Probe, $\mathcal{C}(L_3 - L_{\mathcal{O}}) \in \mathcal{S}$ ist der letzte Querschnitt in der Probe und $\mathcal{C}(L_3 - L_{\mathcal{O}} + 1) \in \mathcal{O}$ ist der erste Querschnitt nach der Probe im Nachlauf. Ein Gitterknoten in \mathcal{L} ist entweder Teil des Festkörpers (Matrix) \mathcal{M} oder Teil des Porenraums \mathcal{P} , womit also gilt $\mathcal{M} \cup \mathcal{P} = \mathcal{L}$ und $\mathcal{M} \cap \mathcal{P} = \emptyset$.

Die wie oben besprochene Antriebskraft ist $\mathbf{b} = b \mathbf{e}_3$ und liegt nur in der Beschleunigungszone $\mathcal{F} \subset \mathcal{I}$ an. In allen hier untersuchten Simulationen gilt $\hat{b} = 10^{-6}$.

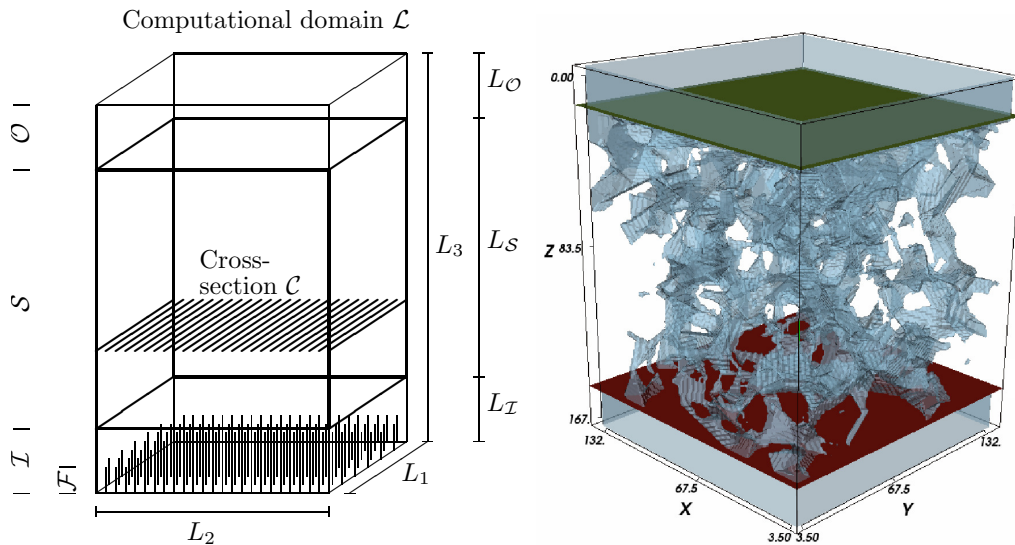


Abbildung 6.2: Aufbau der Gitter-Boltzmann-Simulation. Links als schematische Abbildung und rechts in einer dreidimensionalen Visualisierung einer konkreten Simulation mit diskretisiertem porösem Gestein. Das Simulationsgebiet wird mit \mathcal{L} bezeichnet. Das Gebiet mit porösem Material ist \mathcal{S} . Das Fluid wird im Beschleunigungsgebiet \mathcal{F} durch eine Körperkraft angetrieben. Die Vor- bzw. Nachlaufgebiete \mathcal{I} ('Inlet') und \mathcal{O} ('Outlet') sind wichtig, um Artefakte zu unterdrücken.

Die Simulationen und Resultate werden in entdimensionierten Einheiten behandelt

$$\begin{aligned}\hat{\mathbf{x}} &= \mathbf{x}/\Delta x, & \hat{t} &= t/\Delta t, & \hat{\rho} &= \rho/\rho^\circ, & \hat{p} &= p/(c_s^2 \rho^\circ), \\ \hat{\mathbf{v}} &= \mathbf{v}\Delta t/\Delta x, & \hat{\tau} &= \tau/\Delta t, & \hat{\tau}_{\text{bulk}} &= \tau_{\text{bulk}}/\Delta t, \\ \hat{\kappa} &= \kappa/(\Delta x)^2, & \hat{b} &= b(\Delta t)^2/\Delta x, & \hat{\eta} &= \eta \Delta t/\rho^\circ(\Delta x)^2, \\ \hat{q} &= q \Delta t/(\rho^\circ (\Delta x)^3), & (\hat{\nabla} p)_3 &= (\nabla p)_3 (\Delta t)^2/(\rho^\circ \Delta x),\end{aligned}$$

worin die Diskretisierungen Δx und Δt wie in Abschnitt 6.1.1 verwendet werden und η die dynamische Viskosität ist. Sofern nicht anders angegeben, ist die Relaxationskonstante $\hat{\tau} = 0.857$ bzw. in LB-MRT-Simulationen $\hat{\tau}_{\text{bulk}} = 1.0$. Die Resultate von LB-Simulationen sind durch “^{LB}” gekennzeichnet, wie z. B. die Dichte ρ^{LB} . Falls die Ergebnisse von verschiedenen Implementationen (BGK oder MRT) stammen, werden sie entsprechend gekennzeichnet, also z. B. ρ^{BGK} oder ρ^{MRT} .

Der Mittelwert einer physikalischen Größe w wird abgekürzt

$$\langle w \rangle_{\mathcal{V}} = \frac{1}{|\mathcal{V}|} \sum_{\ell \in \mathcal{V}} w(\ell), \quad (6.1.23)$$

mit $\mathcal{V} \in \{\mathcal{L}, \mathcal{S}, \mathcal{P}, \mathcal{I}, \mathcal{O}, \mathcal{F}, \mathcal{C}(a)\}$ und $|\mathcal{V}|$ der Anzahl an Gitterpunkten in diesem Gebiet. Der Massenfluss q durch einen Querschnitt $\mathcal{C}(a)$ ist

$$q(a) = \sum_{\ell \in \mathcal{C}(a) \cap \mathcal{P}} \rho(\ell) v_3(\ell) (\Delta x)^2, \quad (6.1.24)$$

mit $\rho(\ell)v_3(\ell)$, der Impulskomponente in Flussrichtung. Der Massenfluss ist:

$$Q = \frac{1}{L_3} \sum_{\ell_3=1}^{L_3} q(\ell_3). \quad (6.1.25)$$

6.1.5 Kalibrierung freier Parameter und Fehlerquellen

Zur Kalibrierung der freien Parameter wird ein laminarer Fluss durch ein Rohr mit quadratischem Querschnitt simuliert. Für diese sogenannte Poiseuille Strömung ist die analytische Lösung für den Druck und die Geschwindigkeiten bekannt. Die Simulationsparameter sind

$$\begin{aligned}
 \mathbf{g}(\boldsymbol{\ell}) &= \left(\ell_1 - \frac{L_1 + 1}{2} \right) \mathbf{e}_1 \\
 &\quad + \left(\ell_2 - \frac{L_2 + 1}{2} \right) \mathbf{e}_2 + \left(\ell_3 - \frac{1}{2} \right) \mathbf{e}_3, \\
 \mathcal{S} &= \{ \boldsymbol{\ell} \in \mathcal{L} : 2 \leq \ell_1 \leq L_1 - 1, \\
 &\quad 2 \leq \ell_2 \leq L_2 - 1, 4 \leq \ell_3 \leq L_3 - L_O \}, \\
 \mathcal{I} &= \{ \boldsymbol{\ell} \in \mathcal{L} : \ell_3 \leq L_I \}, \\
 \mathcal{O} &= \{ \boldsymbol{\ell} \in \mathcal{L} : L_3 - L_O \leq \ell_3 \leq L_3 \}, \\
 \mathcal{F} &= \{ \boldsymbol{\ell} \in \mathcal{L} : \ell_3 \leq 2 \}
 \end{aligned} \tag{6.1.26}$$

mit $L_I = 3$, $L_S = L_3 - 6$, $L_O = 3$ und \mathcal{L} dem Simulationsgebiet, siehe Abb. 6.2. Die Abmessungen sind $L_1 = L_2 = B + 2$, $L_3 = 4B$, mit $B \in \{4, 8, 16, 32, 64\}$ der dimensionslosen Kanalweite.

Die analytische Lösung, siehe [76], der Geschwindigkeitskomponente entlang des Rohres auf einem Querschnitt ist

$$\begin{aligned}
 \hat{v}^{\text{TH}}(\hat{x}_1, \hat{x}_2) &= \lim_{M \rightarrow \infty} \hat{v}(\hat{x}_1, \hat{x}_2, M) \\
 \hat{v}(\hat{x}_1, \hat{x}_2, M) &= \frac{(\hat{\nabla} p)_3}{2\hat{\eta}} \left(\frac{B^2}{4} - \hat{x}_2^2 - \frac{8B^2}{\pi^3} \sum_{n=0}^M C_n \right), \\
 C_n &= (-1)^n \frac{\cosh\left(\frac{(2n+1)\pi}{B} \hat{x}_1\right) \cos\left(\frac{(2n+1)\pi}{B} \hat{x}_2\right)}{(2n+1)^3 \cosh\left(\frac{(2n+1)\pi}{2}\right)}
 \end{aligned} \tag{6.1.27}$$

mit $\hat{x}_1 \in [-B/2, B/2]$, $\hat{x}_2 \in [-B/2, B/2]$, $(\hat{\nabla} p)_3$ dem anliegenden Druckgradienten in Flussrichtung, und $\hat{\eta}$, der dynamischen Viskosität. Der Ausdruck $v(\hat{x}_1, \hat{x}_2, M)$ ist unsymmetrisch in \hat{x}_1 und \hat{x}_2 . Entgegen der 'no-slip' Randbedingung gilt $\hat{v}(B/2, \hat{x}_2, M) \neq 0$ für ein endliches M . Als Run-

dungsfehler beim Auswerten der analytischen Formel für finites M wird definiert

$$\tilde{v}(\hat{x}, M) = \frac{2\hat{\eta}}{(\hat{\nabla}p)_3 B^2} \hat{v}(B/2, \hat{x}, M), \quad \hat{x} \in [-B/2, B/2], \quad (6.1.28)$$

und

$$\|v_{\text{wall}}(M)\|_2 := \sqrt{\int_{-B/2}^{B/2} |\tilde{v}(\hat{x}, M)|^2 d\hat{x}} \quad (6.1.29)$$

mit $\tilde{v}(\hat{x}, M)$ der normalisierten Geschwindigkeit an der Wand aus Gl. (6.1.27). In den Simulationen waren die kleinsten Geschwindigkeiten $\hat{v}^{\text{LB}}(\ell)$ von der Größenordnung $[10^{-4}, 10^{-2}]$ abhängig von der Kanalbreite B . Damit der Rundungsfehler $\|v_{\text{wall}}(M)\|_2$ mindestens eine Größenordnung unterhalb dieser kleinsten Geschwindigkeiten ist, muss $M > 50$ sein. In den folgenden Auswertungen wurde $M = 200$ verwendet.

Die Gleichung (6.1.27) beschreibt die stationäre Lösung der Geschwindigkeitskomponente entlang der Flussrichtung für den Fluss durch ein unendlich langes Rohr mit quadratischem Querschnitt und konstantem Druckgradienten. Zum Vergleich mit den Simulationsergebnissen $v^{\text{BGK}}(\ell, t)$ und $\rho^{\text{BGK}}(\ell, t)$ werden voll konvergierte Lösungen mit $t = t_{\text{end}}$ verwendet und der Druckgradient wird ausgewertet. Es sei definiert

$$\delta w(t, dt) = \max_{\ell \in (\mathcal{S} \cap \mathcal{P})} \left(\frac{w(\ell, t) - w(\ell, t - dt)}{w(\ell, t)} \right), \quad (6.1.30)$$

als die maximale relative Änderung der Größe w im Zeitraum dt im gesamten Simulationsbereich $\mathcal{S} \cap \mathcal{P}$, worin $w(\ell, t)$ entweder die Geschwindigkeit $v^{\text{BGK}}(\ell, t)$ oder die Dichte $\rho^{\text{BGK}}(\ell, t)$ ist. Da der Druck proportional zu der Dichte ist, siehe Gl. (6.1.6), ist es ausreichend, nur die Dichte zu betrachten.

Die Ergebnisse von Gleichung (6.1.30) sind in Tabelle 6.1 dargestellt. In den Simulationen kommen Geschwindigkeiten der Größenordnungen $\hat{v}^{\text{BGK}}(\ell, t) \approx 10^{-4}$ vor. Daher sind die absoluten Änderungen von der Größenordnung 10^{-12} , wenn die relativen Änderungen δv aus der Tabelle 6.1 verwendet werden. Die Dichte des Fluides ist $\hat{\rho}^{\text{BGK}}(\ell, t) \approx 1.0$, was absolute Änderungen der Größenordnung 10^{-5} bedeutet. Die Streuung der Druckgradienten kann abgeschätzt werden durch $2 \delta p / (3L) < 10^{-7}$. Für die konvergierten Felder bei $t = t_{\text{end}}$ wird die Notation $v^{\text{BGK}}(\ell)$, $\rho^{\text{BGK}}(\ell)$ und $p^{\text{BGK}}(\ell)$ für die Geschwindigkeit, die Dichte und den Druck verwendet.

Der Druck erhöht sich in der Beschleunigungszone \mathcal{F} und fällt dann entlang der Flussrichtung ab,

siehe Abb. 6.8, wo der mittlere Druck $\langle \hat{p} \rangle_{\mathcal{C}(\ell_3) \cap \mathcal{P}} - 1$ einer LB-BGK-Simulation eines Rohres mit $B = 7$ dargestellt ist. Zur Überprüfung, ob der Druckgradient $(\nabla p)_3$ konstant ist, wie oben gefordert, wird die Größe $\langle p^{\text{BGK}} \rangle_{\mathcal{C}(\ell_3) \cap \mathcal{P}}$ innerhalb der Probe linear modelliert. Bei den LB-Simulationen für Rohre mit Kanalweiten $B = 4, 8, 16, 32$, und 64 sind die Residuen von der Größenordnung 1×10^{-9} , daher kann der Druckgradient als konstant angesehen werden.

Die Geschwindigkeitskomponente in Flussrichtung $v^{\text{LB}}(\boldsymbol{\ell})$ mit $\boldsymbol{\ell} \in \mathcal{C}(L_3/2)$ wird mit der analytischen Lösung aus Gleichung (6.1.27) an den Gitterpunkten $v^{\text{TH}}(\mathbf{g}(\boldsymbol{\ell})_1, \mathbf{g}(\boldsymbol{\ell})_2)$ verglichen. Der Querschnitt $\mathcal{C}(L_3/2)$ wurde so gewählt, dass die endliche Länge des Rohres in der Simulation möglichst wenig Einfluss hat, also die Vor- und Nachlaufkammern möglichst weit weg sind. Der absolute und relative Fehler wird definiert als

$$e_v^{\text{LB}}(\boldsymbol{\ell}) := v^{\text{LB}}(\boldsymbol{\ell}) - v^{\text{TH}}(\mathbf{g}(\boldsymbol{\ell})_1, \mathbf{g}(\boldsymbol{\ell})_2), \quad (6.1.31)$$

$$\epsilon_v^{\text{LB}}(\boldsymbol{\ell}) := \frac{v^{\text{LB}}(\boldsymbol{\ell}) - v^{\text{TH}}(\mathbf{g}(\boldsymbol{\ell})_1, \mathbf{g}(\boldsymbol{\ell})_2)}{v^{\text{TH}}(\mathbf{g}(\boldsymbol{\ell})_1, \mathbf{g}(\boldsymbol{\ell})_2)}. \quad (6.1.32)$$

Der relative Fehler $|\epsilon_v^{\text{BGK}}(\boldsymbol{\ell})|$ ist in Abb. 6.3 dargestellt. Die Abb. 6.4 zeigt $|\epsilon_v^{\text{BGK}}(\boldsymbol{\ell})|$ mit $\ell_2 = L_2/2$ und $\ell_3 = L_3/2$ in einer log-linearen Darstellung für Rohre mit verschiedenen Weiten B einer LB-BGK Simulation. Die größten relativen Fehler treten am Rand der Wand auf und in den Ecken. Mit wachsender Rohrweite B fällt der Fehler schnell unter 1% in weiten Bereichen des Querschnittes.

Bei der bisherigen Untersuchung des Diskretisierungsfehlers wurden Rohre mit quadratischem Querschnitt betrachtet. Die Wände waren daher m Sinne eines möglichst kleinen Fehlers immer optimal mit dem Gitter ausgerichtet. Im Folgenden werden Rohre mit gleichseitigem dreieckigem und rundem Querschnitt untersucht, um ein Maß für den Fehler in stochastischen porösen Medien zu erhalten. Für beide Fälle existieren ebenfalls analytische Lösungen für die Permeabilität. Die Systemparameter

B	$dt/\Delta t$ [$\times 10^3$]	$t_{\text{end}}/\Delta t$ [$\times 10^3$]	$\delta v(t_{\text{end}}, dt)$ [$\times 10^{-8}$]	$\delta \rho(t_{\text{end}}, dt)$ [$\times 10^{-4}$]
4	1	20	2.34	0.174
8	1	20	2.80	0.174
16	5	30	4.27	0.869
32	5	50	7.45	0.869
64	10	120	1.33	1.74

Tabelle 6.1: Maximale relative Änderung der Geschwindigkeit $\delta v(t_{\text{end}}, dt)$ und der Dichte $\delta \rho(t_{\text{end}}, dt)$, siehe Gl. (6.1.30), im Zeitraum dt , wenn die Simulation bei t_{end} abgebrochen wurde. B ist die Kanalweite in dimensionslosen Einheiten und Δt die Zeitdiskretisierung.

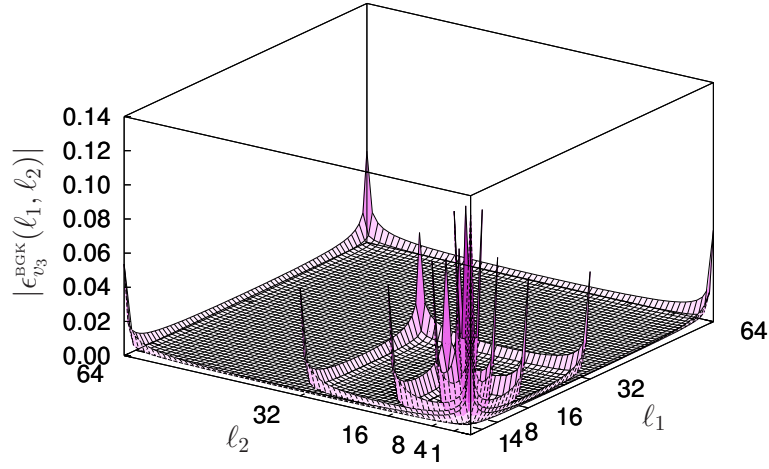


Abbildung 6.3: Übersicht über den relativen Fehler $|\epsilon_v^{\text{BGK}}(\ell)|$ mit $\ell_3 = L_3/2$, für Rohre der Weite $B \in \{4, 8, 16, 32, 64\}$. An den Ecken des Querschnittes treten die größten Fehler auf, gefolgt von den Wandpositionen. Für größere Rohre fällt der Fehler stark ab.

sind $L_1 = J$, $L_2 = J$ und, $L_3 = 4J$ mit dem Skalierungsparameter $J \in \mathbb{N}$. Die Querschnitte sind definiert über den Durchmesser B_\circ (bei Kreisrohren) oder über die Seitenlänge B_Δ (bei dreieckigen Rohren) mit $B_\circ = B_\Delta = J - 2$. Die analytischen Lösungen für das Kreis- bzw. Dreiecksrohr sind:

$$\kappa_\circ^{\text{TH}} = \frac{A_\circ^{\text{TH}}}{8\pi}, \quad A_\circ^{\text{TH}} = \frac{\pi}{4} B_\circ^2 (\Delta x)^2, \quad (6.1.33)$$

$$\kappa_\Delta^{\text{TH}} = \frac{\sqrt{3} A_\Delta^{\text{TH}}}{60}, \quad A_\Delta^{\text{TH}} = \frac{\sqrt{3}}{4} B_\Delta^2 (\Delta x)^2, \quad (6.1.34)$$

mit den Querschnittsflächen A_\circ^{TH} und A_Δ^{TH} . Bei der Diskretisierung dieser Ebenen entstehen die relativen Fehler $\epsilon_{A_\Delta}^{\text{LB}}$ und $\epsilon_{A_\circ}^{\text{LB}}$. Die Permeabilitäten aus den Simulationsergebnissen sind $\kappa_\Delta^{\text{BGK}}$ und $\kappa_\circ^{\text{BGK}}$ und die dazugehörigen relativen Fehler $\epsilon_{\kappa_\Delta}^{\text{BGK}}$ und $\epsilon_{\kappa_\circ}^{\text{BGK}}$. In Abbildung 6.5 sind die Ergebnisse dargestellt. Wie erwartet sind bei ähnlicher Auflösung die Fehler deutlich größer als bei den Rohren mit quadratischem Querschnitt bzw. bei der optimalen Ausrichtung der Gitter (Abb. 6.7). Bei den folgenden Permeabilitätsberechnungen für stochastische poröse Medien sind Fehler in der hier aufgezeigten Größenordnung zu erwarten.

Um abschätzen zu können, wie groß der Einfluss der Relaxationszeit τ auf das berechnete Geschwindigkeitsfeld von LB-BGK- und LB-MRT-Simulationen ist, werden Simulationen mit $\hat{\tau} = 0.7, 1.0, 2.0, 2.5, 3.0$ untersucht. Die Abbildung 6.6 zeigt den relativen Fehler in der Geschwindigkeit $\epsilon_v^{\text{LB}}(\ell)$,

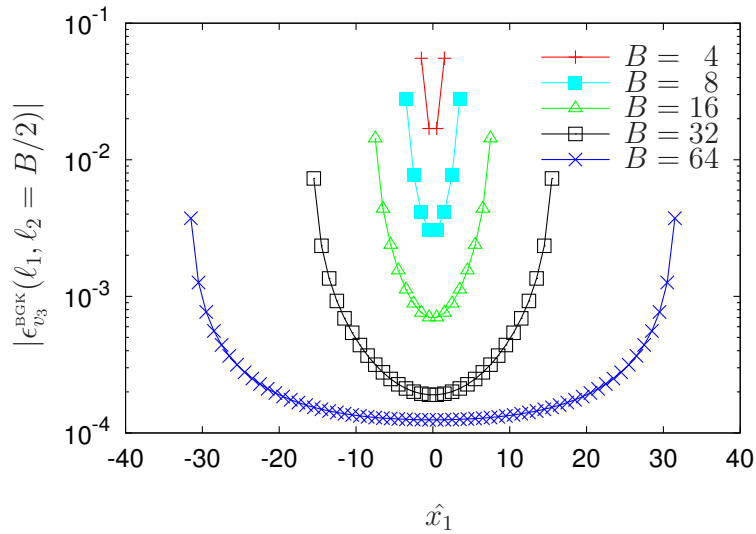


Abbildung 6.4: Ein log-lineare Darstellung des relativen Fehlers auf einer Mittellinie durch den Querschnitt $|\epsilon_v^{\text{BGK}}(\ell)|$ mit $\ell_2 = L_2/2$ und $\ell_3 = L_3/2$.

Gleichung (6.1.32), wobei $\ell_2 = B/2$, $\ell_3 = L_3/2$, und $B = 20$ gilt und beide Implementationen LB-BGK und LB-MRT verwendet wurden. In der Mitte des Rohres ist der Fehler unterhalb von 1%. Bei den LB-BGK-Simulationen tritt der größte Fehler bei großen Relaxationszeiten z. B. $\hat{\tau} = 3.0$ auf und bei den LB-MRT-Simulationen verursachen kleine Relaxationszeiten wie z. B. $\hat{\tau} = 0.7$ die größten Abweichungen. Die Geschwindigkeiten werden bei LB-BGK-Simulationen zu groß berechnet, $\epsilon_v^{\text{BGK}}(\ell) > 0$, und bei LB-MRT-Simulationen zu klein berechnet. Bei der LB-BGK-Methode ist der Fehler 100 mal so groß wie bei der LB-MRT-Methode.

Für die Permeabilitätsberechnung aus dem Darcy-Gesetz, siehe Gl. (6.1.35), wird die mittlere Geschwindigkeit $\langle v^{\text{LB}} \rangle_{\mathcal{S}}$ genutzt. Daher sind besonders der mittlere relative Fehler $\langle |\epsilon_v^{\text{BGK}}| \rangle_{\mathcal{S}}$ und der mittlere absolute Fehler $\langle |e_v^{\text{BGK}}| \rangle_{\mathcal{S}}$ von Interesse. Beide fallen bei steigender Auflösung B schnell ab, siehe Tabelle 6.2. Der mittlere relative Fehler zeigt das Verhalten eines Potenzgesetzes $\langle |\epsilon_v^{\text{BGK}}| \rangle_{\mathcal{S}} = a_1 B^{a_2}$ mit den Parametern $a_1 \approx 0.6$ und $a_2 \approx -1.6$. Dieses Modell für das Fehlergesetz kann genutzt werden, um Fehler für beliebige Weiten B abzuschätzen.

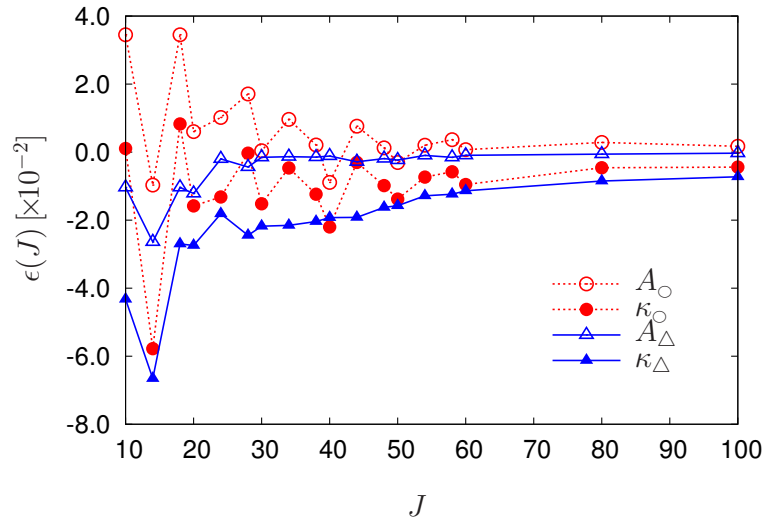


Abbildung 6.5: Relative Fehler der diskretisierten Flächen ϵ_A^{LB} und Permeabilitäten $\epsilon_\kappa^{\text{BGK}}$ für Rohre mit Kreisquerschnitt und gleichseitigem Dreiecksquerschnitt. Der Skalierungsparameter ist J .

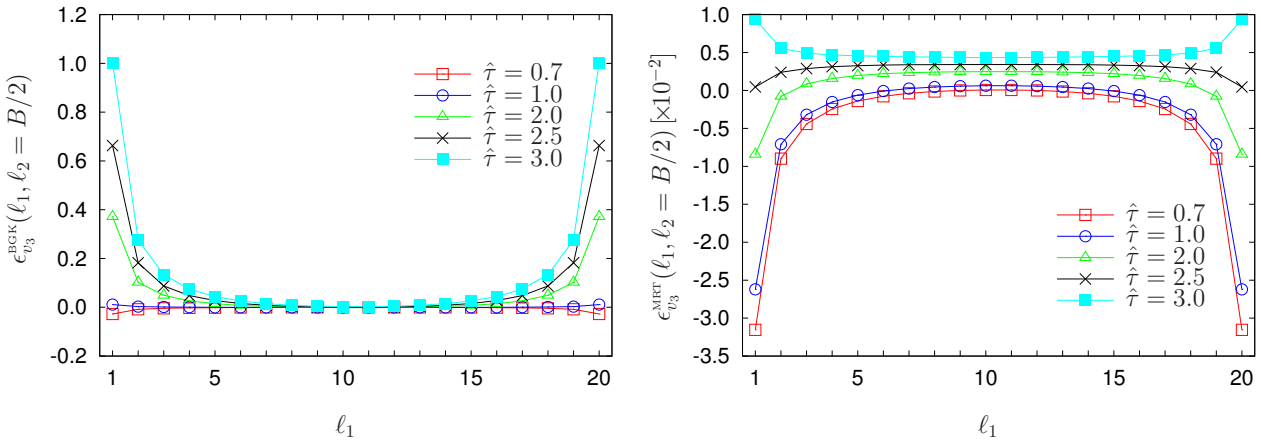


Abbildung 6.6: Relative Fehler $\epsilon_v^{\text{LB}}(\ell)$ mit $\ell_2 = B/2$ und $\ell_3 = L_3/2$ für das Flussfeld durch ein Rohr mit quadratischem Querschnitt der Weite $B = 20$ und verschiedenen Relaxationszeiten $\hat{\tau}$. Links die BGK-, rechts die MRT-Ergebnisse.

Als nächstes werden die aus den gemittelten Geschwindigkeits- und Dichtefeldern berechneten Permeabilitäten aus dem Darcy Gesetz mit analytischen Werten verglichen. Für die Permeabilität κ^{LB}

gilt laut dem Darcy-Gesetz :

$$\kappa^{\text{LB}} = -\eta \frac{\langle v^{\text{LB}} \rangle_{\mathcal{S}}}{\langle (\nabla p)_3^{\text{LB}} \rangle_{\mathcal{S}}}, \quad (6.1.35)$$

mit η der dynamischen Viskosität, $\langle v_3^{\text{LB}} \rangle_{\mathcal{S}}$ ist die durchschnittliche Geschwindigkeit in der Probe und $\langle (\nabla p)_3^{\text{LB}} \rangle_{\mathcal{S}}$ ist der gemittelte Druckgradient in Flussrichtung. Wie genau dieser Druckgradient gemittelt und aus dem direkten Simulationsergebnis $\rho^{\text{LB}}(\ell)$ berechnet wird, wird weiter unten diskutiert. Die dynamische Viskosität ist $\eta = \nu \bar{\rho}^{\text{LB}}$ mit $\bar{\rho}^{\text{LB}} = \langle \rho^{\text{LB}} \rangle_{\mathcal{P} \cap \mathcal{S}}$, was approximiert wird durch

$$\bar{\rho}^{\text{LB}} \approx \frac{\langle \rho^{\text{LB}} \rangle_{\mathcal{C}(L_{\mathcal{T}+1})} + \langle \rho^{\text{LB}} \rangle_{\mathcal{C}(L_3 - L_{\mathcal{O}})}}{2}. \quad (6.1.36)$$

Die analytisch berechnete Permeabilität ist nach [76]

$$\kappa^{\text{TH}}(B) = \lim_{M \rightarrow \infty} \frac{B^2}{4} \left(\frac{1}{3} - \frac{64}{\pi^5} \sum_{n=0}^M \frac{\tanh\left(\left(2n+1\right)\frac{\pi}{2}\right)}{(2n+1)^5} \right), \quad (6.1.37)$$

worin wieder $M = 200$ zur numerischen Evaluierung verwendet wird. Der Fehler ist definiert durch

$$\epsilon_{\kappa}^{\text{LB}}(B) := \frac{\kappa^{\text{LB}}(B) - \kappa^{\text{TH}}(B)}{\kappa^{\text{TH}}(B)}. \quad (6.1.38)$$

In Abb. 6.7 sind die relativen Fehler $\epsilon_{\kappa}^{\text{BGK}}(B)$ und $\epsilon_{\kappa}^{\text{MRT}}(B)$ dargestellt. Für Rohre mit einer Auflösung von mehr als $B = 16$ sind die Fehler kleiner als 1 %. In dieser Darstellung erscheint die LB-BGK-Simulation als die etwas bessere, aber es ist zu beachten, dass die Relaxationszeit von $\hat{\tau} = 0.857$ extra aus dieser Geometrie kalibriert wurde. Mit einem kalibrierten Relaxationsparameter kann also auch die technisch schwächere Methode durchaus gute Resultate liefern. In realistischen

B	$\langle \epsilon_v^{\text{BGK}} \rangle_{\mathcal{S}}$	$\langle \hat{\epsilon}_v^{\text{BGK}} \rangle_{\mathcal{S}} [\times 10^{-6}]$
40	0.064099	0.300
80	0.021840	0.114
160	0.007241	0.068
320	0.002331	0.034
640	0.000784	0.023

Tabelle 6.2: Mittlere relative Fehler $\langle |\epsilon_v^{\text{BGK}}| \rangle_{\mathcal{S}}$ und der gemittelte absolute Fehler $\langle |\hat{\epsilon}_v^{\text{BGK}}| \rangle_{\mathcal{S}}$ für verschiedene Rohrweiten B .

porösen Medien jedoch ist es nicht möglich, eine optimale Relaxationszeit im Vorhinein zu bestimmen, da Querschnitte der Porenhäule und Porenformen zu stark variieren. Daher ist für diese Art der Simulation die LB-MRT-Methode prinzipiell die verlässlichere, da sie weniger abhängig ist von τ .

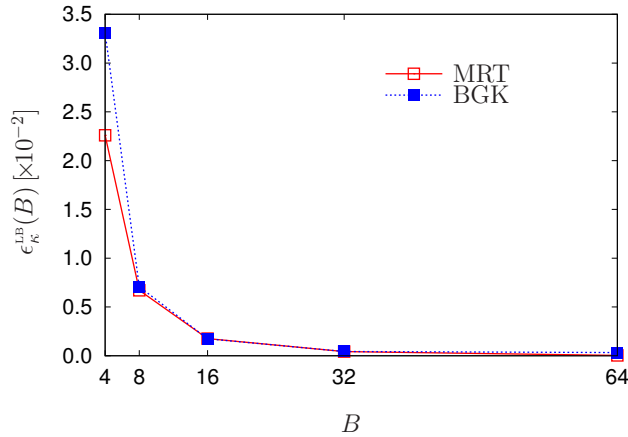


Abbildung 6.7: Relativer Fehler $\epsilon_{\kappa}^{\text{LB}}(B)$ der berechneten Permeabilitäten $\kappa^{\text{LB}}(B)$ in Abhängigkeit von der Kanalweite B .

Bei der Berechnung der Permeabilität mithilfe des Darcy-Gesetzes (6.1.35) wird der mittlere Druckgradient in Flussrichtung benötigt. Verschiedene Methoden, um diesen zu berechnen, wurden im Rahmen der Arbeit untersucht [60]. Die Ergebnisse und die für die weiteren Ergebnisse verwendete Methode werden hier wiedergegeben. Der mittlere Druckgradient kann wie folgt berechnet werden:

1. Durch die Steigung einer linearen Regression durch die Daten $\langle p \rangle_{\mathcal{C}(\ell_3) \cap \mathcal{P}}$ mit allen Querschnitten $\mathcal{C}(\ell_3)$, $L_{\mathcal{I}} + 1 \leq \ell_3 \leq L_3 - L_{\mathcal{O}}$.
2. Wie 1. aber nur mit den Querschnitten $\mathcal{C}(\ell_3)$, $L_{\mathcal{I}} + 1 + W \leq \ell_3 \leq L_3 - L_{\mathcal{O}} - W$, $W \in \mathbb{N}$, siehe Abb. 6.8. Es werden also nur innere Querschnitte verwendet, die mindestens $W \in \mathbb{N}$ Ebenen von den Rändern entfernt sind. Dadurch sollen Fehler, die durch die Randbedingungen entstehen, minimiert werden.
3. Durch das arithmetische Mittel des Druckes bei $\mathcal{C}(L_{\mathcal{I}} + 1)$ und $\mathcal{C}(L_3 - L_{\mathcal{O}})$,

$$\langle (\nabla p)_3 \rangle_S \approx \frac{\langle p \rangle_{\mathcal{C}(L_3 - L_{\mathcal{O}}) \cap \mathcal{P}} - \langle p \rangle_{\mathcal{C}(L_{\mathcal{I}} + 1) \cap \mathcal{P}}}{(L_S - 1)\Delta x}, \quad (6.1.39)$$

mit der Probenlänge L_S .

4. Ähnlich wie 3. aber mit

$$\langle (\nabla p)_3 \rangle_S \approx \frac{\langle p \rangle_{\mathcal{C}(L_3-L_{\mathcal{O}}+1) \cap \mathcal{P}} - \langle p \rangle_{\mathcal{C}(L_{\mathcal{I}}) \cap \mathcal{P}}}{\text{wie}(L_S + 1) \Delta x}. \quad (6.1.40)$$

Zum Vergleich der Methoden wurde ein System mit den folgenden Parametern verwendet:

$$\begin{aligned} L_1 &= 40, L_2 = 40, L_3 = 80, \\ \mathcal{S} &= \{\ell \in \mathcal{L} : 17 \leq \ell_1 \leq 23, 17 \leq \ell_1 \leq 23, 21 \leq \ell_3 \leq 60\}, \\ \mathcal{I} &= \{\ell \in \mathcal{L} : 1 \leq \ell_3 \leq 20\}, \mathcal{O} = \{\ell \in \mathcal{L} : 61 \leq \ell_3 \leq 80\}, \\ \mathcal{F} &= \{\ell \in \mathcal{L} : 6 \leq \ell_3 \leq 15\}, L_{\mathcal{I}} = 20, L_{\mathcal{S}} = 40, L_{\mathcal{O}} = 20. \end{aligned} \quad (6.1.41)$$

Der mittlere Druck $\langle \hat{p} \rangle_{\mathcal{C}(\ell_3) \cap \mathcal{P}}$, ist in Abb. 6.8 zu sehen.

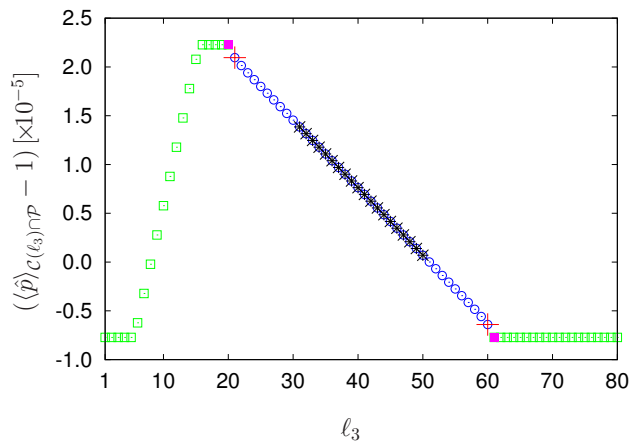


Abbildung 6.8: Mittlerer Druck $(\langle \hat{p} \rangle_{\mathcal{C}(\ell_3) \cap \mathcal{P}} - 1)$ bei einer LB-BGK-Simulation. Die für die verschiedenen Methoden der Berechnung verwendeten Bereiche sind dargestellt als: 1) Alle Probenebenen für einen linearen Fit (\odot). 2) Linearer Fit, aber mit Abstandsmaß $W = 10$ (\times). 3) Mittelung nur durch eine Ein- und Auslaufebene $\mathcal{C}(L_{\mathcal{I}}+1)$ und $\mathcal{C}(L_3-L_{\mathcal{O}})$ innerhalb der Probe ($+$). 4) Mittelung mit Ein- und Auslaufebenen $\mathcal{C}(L_{\mathcal{I}})$ und $\mathcal{C}(L_3-L_{\mathcal{O}}+1)$ nach der Probe (\blacksquare). Die grünen Knoten \square sind die Ebenen in den Kammern (\mathcal{I} und \mathcal{O}).

Das mittlere Druckfeld $\langle p \rangle_{\mathcal{C}(\ell_3) \cap \mathcal{P}}$ hat zwei Sprungstellen, am Anfang der Probe bei ($\ell_3 = 20$) und am Ende der Probe bei a ($\ell_3 = 60$). Verantwortlich dafür ist die Kompressibilität des Fluides. Wegen der periodischen Randbedingungen in Flussrichtung ist das Druckfeld konstant in den Gebieten \mathcal{I} und

\mathcal{O} und steigt dann in der Beschleunigungszone \mathcal{F} an. Die Ergebnisse der verschiedenen Methoden zur Berechnung des Druckgradienten sind in der Tabelle 6.3 zu sehen. Bei den Methoden 1, 2 und 3 ergibt sich ein Fehler von weniger als 1%. Die Methode 4 hat einen deutlich höheren Fehler und erscheint daher für stochastische poröse Medien als nicht brauchbar. Für alle weiteren Ergebnisse wird Methode 3 verwendet. Sie ist akkurat genug und einfach zu implementieren, da kein rechentechnisch teurer Fit über die Daten notwendig ist. In der letzten Zeile von Tabelle 6.3 sind Vergleichswerte für eine Simulation ohne Vor- und Nachlaufkammern und mit einer Beschleunigung über das gesamte Simulationsgebiet angegeben. Die Resultate sind zwar vergleichbar für diese Simulation, jedoch ist das Verfahren nicht auf Simulationen mit stochastischen porösen Methoden anwendbar.

$B = 7$	1	2	3	4
$\hat{\kappa}^{\text{LB}}$	1.73244	1.73689	1.71346	1.64270
$\epsilon_{\kappa} [\%]$	0.61423	0.87278	0.48787	4.59761
Ohne Vor- und Nachlaufkammern				
$\hat{\kappa}^{\text{LB}}$	1.73689			
$\epsilon_{\kappa} [\%]$	0.87277			

Tabelle 6.3: Vergleich der verschiedenen Methoden zur Berechnung des Druckgradienten $\langle(\nabla p)_3\rangle_S$. Die berechneten Permeabilitäten sind $\hat{\kappa}^{\text{LB}}$ und der relative Fehler ist ϵ_{κ} . Verwendet wurde $B = 7$ und für die Methode 2 gilt $W = 10$.

6.2 Permeabilitätsberechnung eines Fontainebleau-Sandsteins

In diesem Kapitel wird die Permeabilität eines Fontainebleau-Sandsteins mithilfe der oben besprochenen Simulationsmethode berechnet. Die diskretisierten Daten wurden aus einem diskretisierten und segmentierten μ -CT-Datensatz erzeugt. Die Ergebnisse werden mit anderen Permeabilitätsberechnungen für genau denselben Datensatz verglichen. Die Vergleichsergebnisse werden in einer Finiten-Differenzen-Simulation und in einer weiteren unabhängigen LB Simulationen gewonnen [59]. Weiterhin existieren für diese Probe auch Resultate aus einer experimentellen Bestimmung der Permeabilität [58]. Die Berechnungen wurden einmal in niedriger Auflösung (l.r.) und ein Mal in hoher Auflösung (h.r.) durchgeführt. Die l.r. Systemparameter sind:

$$\begin{aligned}L_1 &= 305, L_2 = 305, L_3 = 320, \\ \mathcal{S} &= \{\ell \in \mathcal{L} : 3 \leq \ell_1 \leq 303, 3 \leq \ell_2 \leq 303, 11 \leq \ell_3 \leq 310\}, \\ \mathcal{I} &= \{\ell \in \mathcal{L} : 1 \leq \ell_3 \leq 10\}, \mathcal{O} = \{\ell \in \mathcal{L} : 311 \leq \ell_3 \leq 320\}, \\ \mathcal{F} &= \{\ell \in \mathcal{L} : 1 \leq \ell_3 \leq 5\}, L_{\mathcal{I}} = 10, L_{\mathcal{S}} = 300, L_{\mathcal{O}} = 10, \\ \Delta x &= 7.5 \times 10^{-6} \text{ m.}\end{aligned}\tag{6.2.1}$$

Der hochaufgelöste Datensatz wurde aus dem niedrig aufgelösten erzeugt, indem jedes Voxel durch acht Voxel ersetzt wurde. Die h.r. Systemparameter sind

$$\begin{aligned}L_1 &= 605, L_2 = 605, L_3 = 620, \\ \mathcal{S} &= \{\ell \in \mathcal{L} : 3 \leq \ell_1 \leq 602, 3 \leq \ell_2 \leq 602, 11 \leq \ell_3 \leq 610\}, \\ \mathcal{I} &= \{\ell \in \mathcal{L} : 1 \leq \ell_3 \leq 10\}, \mathcal{O} = \{\ell \in \mathcal{L} : 611 \leq \ell_3 \leq 620\}, \\ \mathcal{F} &= \{\ell \in \mathcal{L} : 1 \leq \ell_3 \leq 5\}, L_{\mathcal{I}} = 10, L_{\mathcal{S}} = 600, L_{\mathcal{O}} = 10, \\ \Delta x &= 3.75 \times 10^{-6} \text{ m.}\end{aligned}\tag{6.2.2}$$

Zur Permeabilitätsberechnung wurde ein approximiertes Darcy-Gesetz verwendet, siehe dazu Gleichungen (6.1.35), (6.1.39) und (6.1.36),

$$\kappa = - \frac{\nu(L_S - 1)\Delta x \langle v_3 \rangle_S (\langle \rho \rangle_{C(L_{I+1}) \cap \mathcal{P}} + \langle \rho \rangle_{C(L_3 - L_O) \cap \mathcal{P}})}{2 (\langle p \rangle_{C(L_3 - L_O) \cap \mathcal{P}} - \langle p \rangle_{C(L_{I+1}) \cap \mathcal{P}})}. \quad (6.2.3)$$

Die kinematische Viskosität ν ist durch Gl. (6.1.15) gegeben mit $\hat{\tau}$, aus der Tabelle 6.4, $\hat{\tau}_{\text{bulk}} = 1.0$ und Δt aus Gl. (6.1.7) mit $c_s = 1 \text{ m s}^{-1}$. Es wurden LB-BGK- und LB-MRT-Simulationen durchgeführt, jeweils mit 100 000 Simulationsschritten. Die Relaxationsparameter sind in Tabelle 6.4 aufgelistet. Die Resultate κ_{extrap} sind die für eine unendlich hohe Auflösung linear extrapolierten Permeabilitäten.

Der berechnete Wert $\kappa_{\text{l.r.}} = 608 \text{ [mD]}$ für $\hat{\tau} = 0.688$, siehe Tabelle 6.4, deckt sich gut mit dem Wert $\kappa = 621 \text{ [mD]}$ aus [59]. Wie schon oben erwähnt, sind die Resultate aus dem LB-BGK, Simulationen stark abhängig von den Relaxationszeiten $\hat{\tau}$. Im Allgemeinen sollte die LB-MRT, Methode verwendet werden. Auch ist der Einfluss von $\hat{\tau}$ bei komplexen Geometrien wesentlich stärker als bei den untersuchten Rohrgeometrien. Die extrapolierten Permeabilitäten streuen im Bereich von 473–510 [mD], siehe Abb. 6.9 (unten), unabhängig von der verwendeten Simulationsmethode und der Relaxationszeit. Für gut aufgelöste Datensätze und bei Verwendung einer solchen Extrapolationsmethode aus möglichst vielen Simulationsresultaten ist daher zu erwarten, dass sich verlässliche Permeabilitätsberechnungen gewinnen lassen, auch aus LB-BGK-Simulationen ohne besonders kalibrierte Relaxationszeiten, also z. B. mit $\hat{\tau} = 1$.

LB-Methode	$\hat{\tau}$	$\kappa_{\text{l.r.}}$ [mD]	$\kappa_{\text{h.r.}}$ [mD]	κ_{extrap} [mD]
BGK	0.688	621	—	—
BGK	0.688	608	559	510
BGK	0.857	773	634	495
MRT	0.688	505	489	473
MRT	0.857	558	518	478
MRT	1.000	601	541	481

Tabelle 6.4: Simulationsresultate für die Permeabilitäten, $\kappa_{\text{l.r.}}$ mit niedriger Auflösung und $\kappa_{\text{h.r.}}$ hoher Auflösung, bei verschiedenen Simulationsmethoden LB-BGK und LB-MRT und Relaxationszeiten $\hat{\tau}$. Extrapolierte Ergebnisse sind κ_{extrap} .

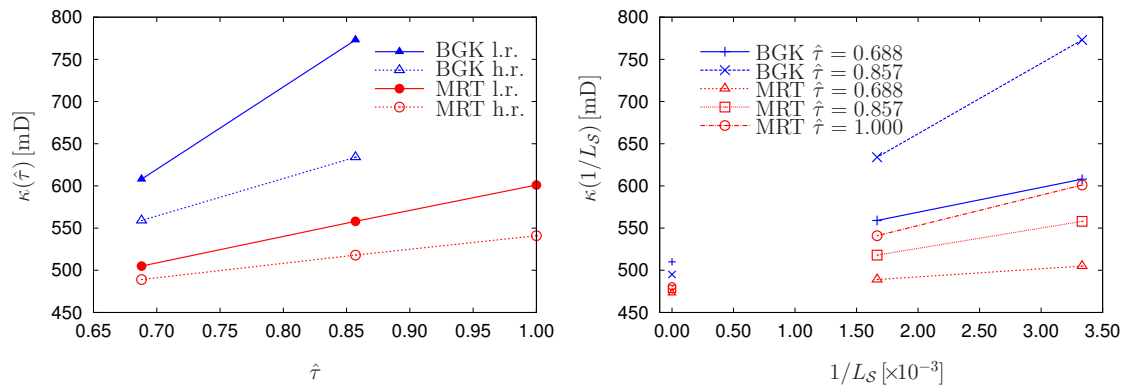


Abbildung 6.9: Oben die Ergebnisse für die Permeabilität berechnet mit verschiedenen $\hat{\tau}$ und LB-Methoden, jeweils für die l.r. und h.r. Probe. Der Einfluss der Relaxationszeit $\hat{\tau}$ ist bei den BGK -Implementation stärker. Unten die Resultate der Permeabilität für die l.r. Probe mit $L_S = 300$ und die h.r. Probe mit $L_S = 600$. Bei $1/L_S = 0$ sind die extrapolierten Resultate κ_{extrap} zu sehen.

6.3 Fazit

Gitter-Boltzmann-Simulationen eignen sich gut zur Berechnung der Permeabilitäten. Die Resultate der Permeabilitätsberechnungen selbst aus einfachen LB-BGK-Simulationen sind prädiktiv und liefern gute Resultate. Wichtig sind allerdings eine Eichung von $\hat{\tau}$ und eine gute Auflösung der Probe. Kann diese Eichung nicht sinnvoll vorgenommen werden, zum Beispiel wenn die Porenraumdurchmesser innerhalb der Geometrie um Größenordnungen schwanken, sollte auf die LB-MRT zurückgegriffen werden. Eine gute Auflösung ist erst zu erwarten, wenn auch die kleinsten Porenhälse, die zur Konnektivität beitragen, mit einigen Voxel aufgelöst sind. Allgemein empfiehlt sich eine auflösungsabhängige Analyse mit anschließender Extrapolation.

7 Auflösungsabhängige Berechnungen aus digitalen 3D-Modellen

Die Verfügbarkeit von hochaufgelösten digitalen Datensätzen aus kalibrierten realistischen Rekonstruktionsmodellen eines porösen Mediums ist für die Berechnung von physikalischen Transportparametern und geometrischen Charakteristika des porösen Mediums von entscheidender Bedeutung - besonders wenn, wie besprochen in Abschnitt 6, die Berechnung auf einer Extrapolation über die Auflösung beruht. Um dabei aussagekräftige Ergebnisse zu liefern, muss für dasselbe Modell eine Bandbreite an Auflösungen zur Verfügung stehen. Im Rahmen dieser Arbeit wurden digitale 3D-Datensätze in vielen Auflösungen von grob bis extrem hochaufgelöst für einen Fontainebleau-Sandstein erstellt [40]. Dieser - nach eigenem Kenntnisstand - größte der Allgemeinheit zur Verfügung stehende Datensatz ist eine Referenz in diesem Forschungsfeld [42, 68, 67]. In dieser Arbeit wurde der Datensatz genutzt, um geometrische Eigenschaften wie Porosität und spezifische innere Oberfläche direkt aus der, bei der Diskretisierung erzeugten, Grauwertverteilung zu berechnen. Weiterhin wurden Teilgebiete mit verschiedenen Auflösungen diskretisiert um auflösungsabhängige Berechnung im Rahmen der LPT und für die Permeabilität zu erhalten.

7.1 Diskretisierung des gesamten Kontinuumsmodells

Das mehrskalige Kontinuumsmodell eines Fontainebleau-Sandsteines aus den Arbeiten [9, 10, 53] wurde verwendet und mit Auflösungen von $a = 117.12 \mu\text{m}, \dots, 0.4576 \mu\text{m}$ digitalisiert. Das Gebiet für das Kontinuumsmodell ist ein Zylinder der Höhe 2.5 cm und Durchmesser 2.5 cm . Untersucht wird ein kubischer Ausschnitt der Seitenlänge 1.5 cm , welcher aus ungefähr 1.02 Millionen Quarz-Kristallen besteht. Es existieren 99 unterschiedliche Quarztypen und jeder dieser konvexen Quarztypen ist analytisch beschrieben durch 18 Ebenen. Diese Quarztypen wurden stochastisch im Gebiet deponiert, rotiert und skaliert und gegen experimentelle Daten kalibriert, siehe Abb. 7.2 und Abb. 7.1.

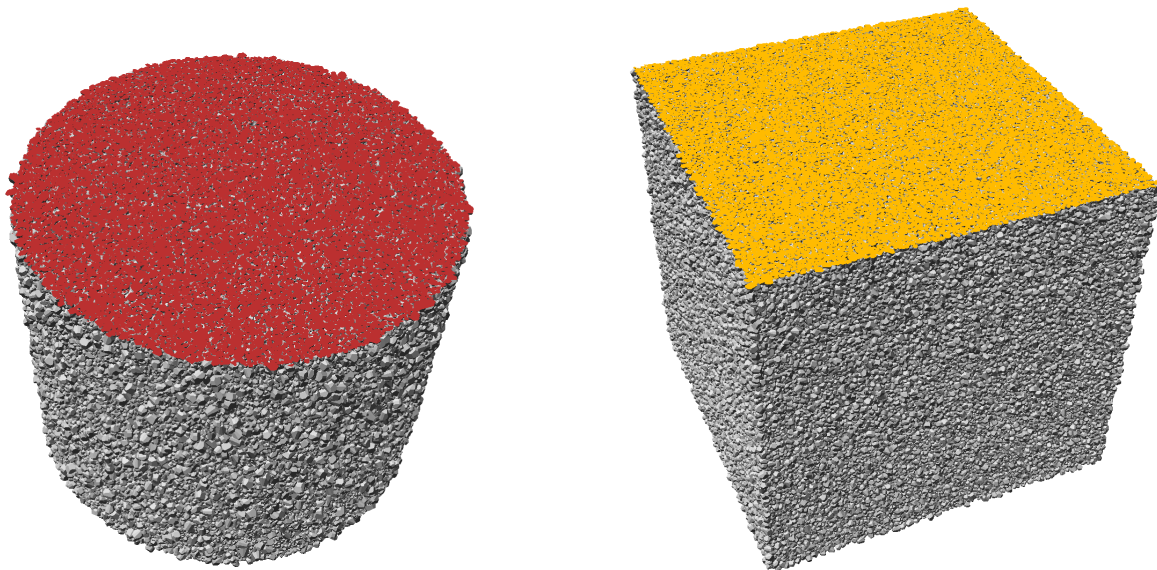


Abbildung 7.1: Links die 3D-Darstellung des Kontinuumsmodells in seiner Zylinderform mit einer Höhe und einem Durchmesser von 2.5 cm . Rechts ein Würfel mit Seitenlänge $L = 1.5 \text{ cm}$, der aus ca. 1.02 Millionen Einzelkristallen besteht. Er wurde aus dem Innern des Zylinders 'herausgeschnitten'. Die 3D Bilder wurden mit einer speziell für diese Kontinuumsmodelle optimierten Bildverarbeitungssoftware 'MegaMol' [31] erzeugt.

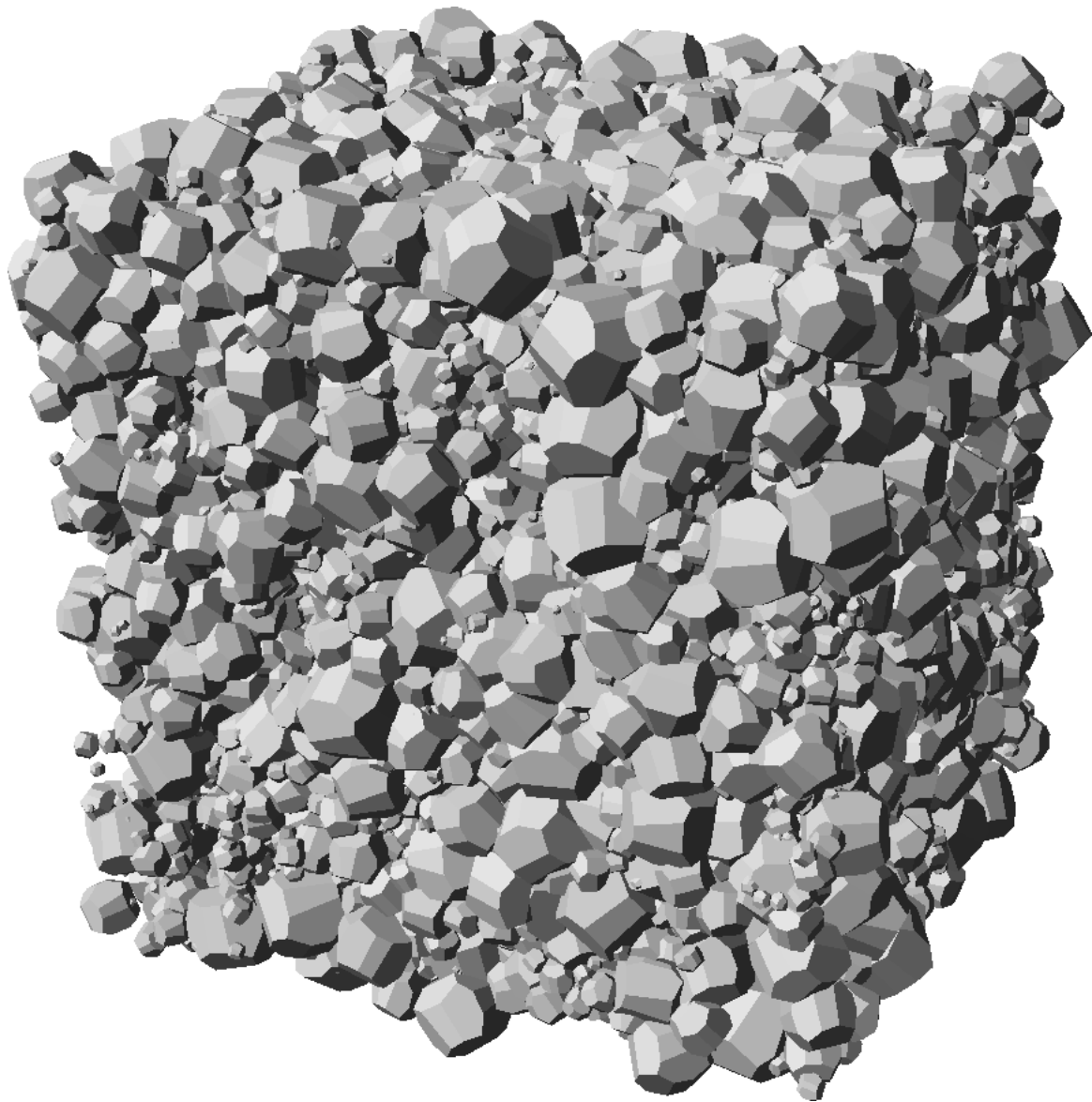


Abbildung 7.2: Kleiner Ausschnitt aus dem Kontinuumsmodell. Zu erkennen sind die unterschiedlich großen und stochastisch orientierten sich überlappenden Einzelkristalle. Dieser Ausschnitt besteht aus ca. 100.000 Kristallen.

Die Probe \mathbb{S} wurde in N kubische Würfel (Voxel) unterteilt, $\mathbb{V}_i(a) \subset \mathbb{S}$, mit $i = 1, \dots, N$. Die Seitenlänge $a = l \cdot a_0$ der Voxel, mit $l \in \mathbb{N}$, sind Vielfache der Basislänge $a_0 = 1.5 \text{ cm}/16384 = 0.9155273 \mu\text{m}$. Jedes Voxel $\mathbb{V}_i(a)$ enthält $6^3 = 216$ gleichmäßig zentrierte Stützstellen. Durch diese Stützstellen erhält jeder Voxel einen Grauwert $0 \leq g_i(a) \leq 216$ mit $1 \leq i \leq N$. Dieser Grauwert ist gleich der Anzahl an Stützstellen in jedem Voxel, die innerhalb der Matrix \mathbb{M} und nicht im Porenraum \mathbb{P} liegen, siehe [53]. Die Grauwerte approximieren das Integral

$$\psi_i(a) = \frac{1}{a^3} \int_{\mathbb{V}_i \cap \mathbb{P}} d^3x \approx 1 - \frac{g_i(a)}{216} \quad (7.1.1)$$

welches der Porosität des Voxel \mathbb{V}_i entspricht. Es gilt also: $g_i(a) = 0$ approximiert $\mathbb{V}_i \subset \mathbb{P}$ und $g_i(a) = 216$ approximiert $\mathbb{V}_i \subset \mathbb{M}$. Tabelle 7.1 ist eine Übersicht über alle erzeugten Datensätze.

a	$128a_0$	$64a_0$	$32a_0$	$16a_0$	$8a_0$	$2a_0$	$4a_0$	a_0	$a_0/2$
L	128	256	512	1024	2048	4096	8192	16384	32768
N	2^{21}	2^{24}	2^{27}	2^{30}	2^{33}	2^{36}	2^{39}	2^{42}	2^{45}

Tabelle 7.1: Übersicht über die erstellten Datensätze. Die Auflösung a ist als Vielfache der Basisauflösung $a_0 = 0.91552734 \mu\text{m}$ gegeben. Die Gesamtanzahl an Voxeln ist N . Jeder Voxel wird als 1 Byte repräsentiert. Dadurch ergibt sich ein Gesamtspeicherplatz von über 4 Terabytes.

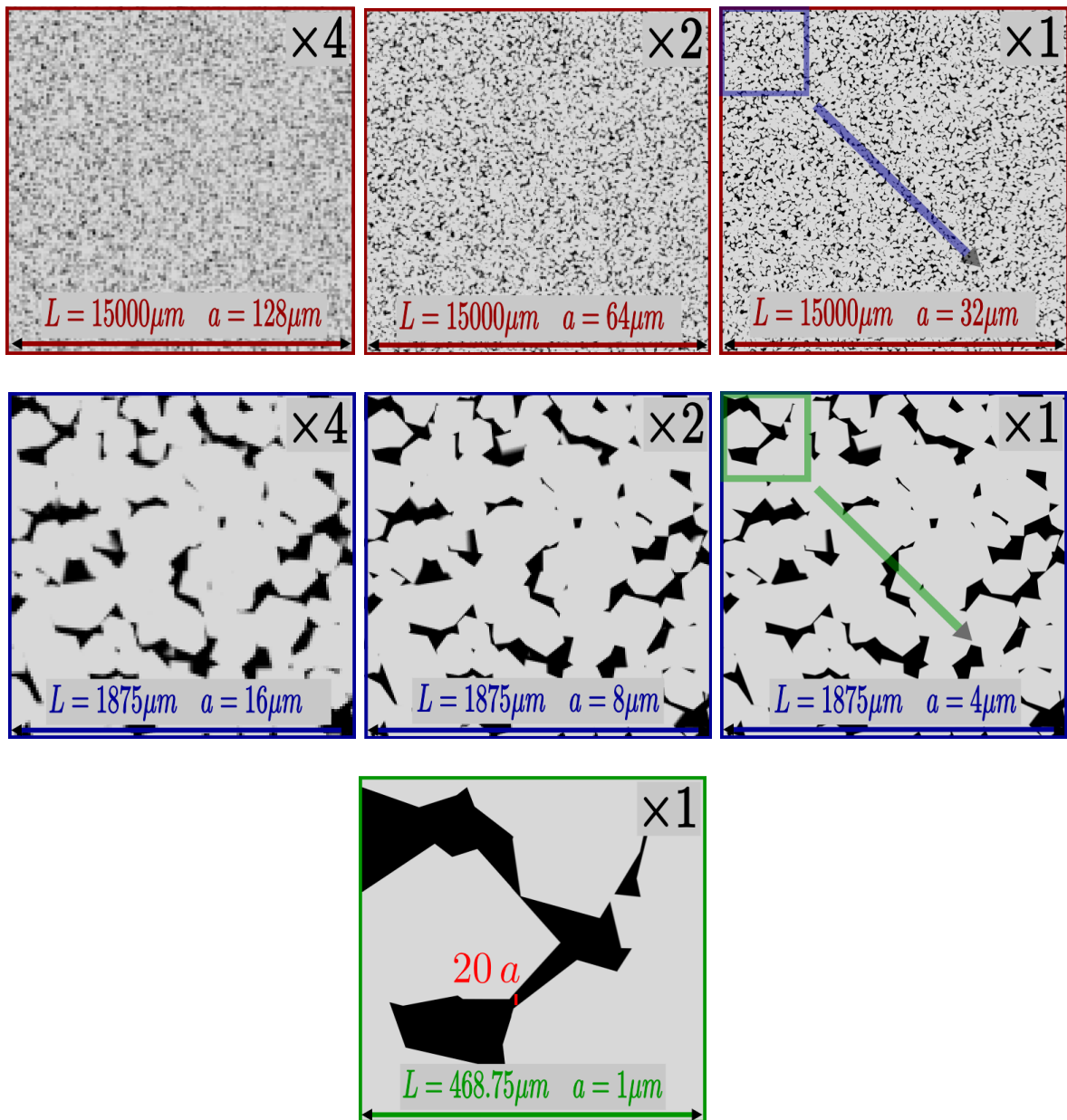


Abbildung 7.3: Querschnitte des diskretisierten und segmentierten Kontinuumsmodells bei verschiedenen Auflösungen $a = 1 \mu\text{m} - 128 \mu\text{m}$. Der Porenraum \mathbb{P} ist schwarz und die Matrix \mathbb{M} grau dargestellt. Als Porenraum gelten alle Voxel $\mathbb{V}_i(a)$ mit $g_i(a) < 128$, alle anderen Voxel sind Matrix. In der ersten Reihe ist ein quadratischer Querschnitt der Seitenlänge $L = 15 \text{ mm}$ zu sehen, in der zweiten Reihe ein Querschnitt mit $L = 1.875 \text{ mm}$ und in der letzten Reihe mit $L = 0.46875 \text{ mm}$. Im jeweils letzten Bild der ersten und zweiten Reihe ist der Ausschnitt der nächsten Vergrößerung der nächsten Zeile dargestellt.

7.2 Berechnung der Porosität aus der Grauwertverteilung

In diesem Abschnitt werden die Porosität und die spezifische innere Oberfläche berechnet. Vorge stellt wird eine Berechnung, die nur die aus der Diskretisierung erhaltenen Grauwert histogramme benötigt. Die Ergebnisse dieser neuen Berechnungsmethode werden mit anderen Standardmethoden verglichen.

Die diskrete Grauwertverteilung, also die Anzahl der Voxel mit Grauwert g bei einer Auflösung a , sei

$$M(g, a) = \sum_{i=1}^N \delta_{gg_i(a)}$$

mit $\delta_{ij} = 1$ für $i = j$ und $\delta_{ij} = 0$ für $i \neq j$.

Eine Abschätzung der Porosität ist dann

$$\phi(g_{th}, a) = \frac{1}{N} \sum_{g=0}^{g_{th}} M(g, a)$$

als Funktion der Auflösung a und des bei der Diskretisierung verwendeten Schwellwertes $0 \leq g_{th} \leq 215$. Der Schwellwert gibt an, über welchem Wert ein Voxel als Matrix erachtet wird, siehe Kapitel 4.7.2.

Die exakte Porosität des Kontinuumsmodells, dargestellt als unendliche feine Diskretisierung, ist

$$\phi = \lim_{a \rightarrow 0} \phi(g_{th}, a)$$

und es gilt $\phi(0, a) \leq \phi \leq \phi(215, a)$ für alle g_{th} und a .

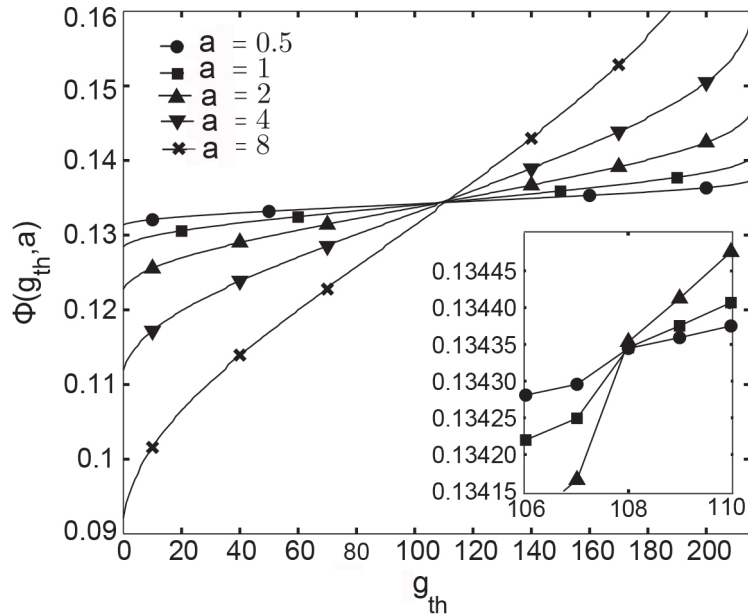


Abbildung 7.4: Die berechnete Porosität $\phi(g_{th}, a)$ als Funktion des Schwellwerts g_{th} für verschiedene Auflösungen $a = 8a_0, 4a_0, 2a_0, a_0, a_0/2$.

In Abbildung 7.4 ist die Porosität $\phi(g_{th}, a)$ als Funktion des Schwellwerts g_{th} für verschiedene Auflösungen $a = 8a_0, 4a_0, 2a_0, a_0, a_0/2$ dargestellt. Abbildung 7.5 zeigt die Schranken $\phi(0, a)$ bzw. $\phi(215, a)$ für $a = 8a_0, 4a_0, 2a_0, a_0, a_0/2$ und die acht linearen Extrapolationen

$$\phi_i(a; c) = \varphi_i(c) + k_i(c)a \quad (7.2.1)$$

mit $i \in \{0, 215\}$. Die Steigungen $k_i(c)$ und Achsendurchgänge $\varphi_i(c)$ werden so bestimmt, dass gilt $\phi_i(a; c) = \phi(i, a)$ für $a = c$ und $a = c/2$. Weiterhin werden Porositäten, als Datenpunkte durch das Symbol \times dargestellt, gezeigt, die mit Schwellwerten berechnet werden, welche die Varianz der Grauwertistogramme maximieren, siehe [62]. Die berechneten Werte sind in Tabelle 7.2 angegeben.

c	$k_0(c)$	$\varphi_0(c)$	$k_{215}(c)$	$\varphi_{215}(c)$
$8a_0$	-0.004944104133756	0.131677153054625	0.006128869055829	0.133806445926894
$4a_0$	-0.005437994277600	0.133652713629999	0.006038194581379	0.134169143824690
$2a_0$	-0.005680456774371	0.134137638623542	0.005981708264244	0.134282116458962
a_0	-0.005843877976029	0.134301059825200	0.005942813391024	0.134321011332180

Tabelle 7.2: Steigungen und Achsenabschnitte der Extrapolationen für die Gleichung 7.2.1

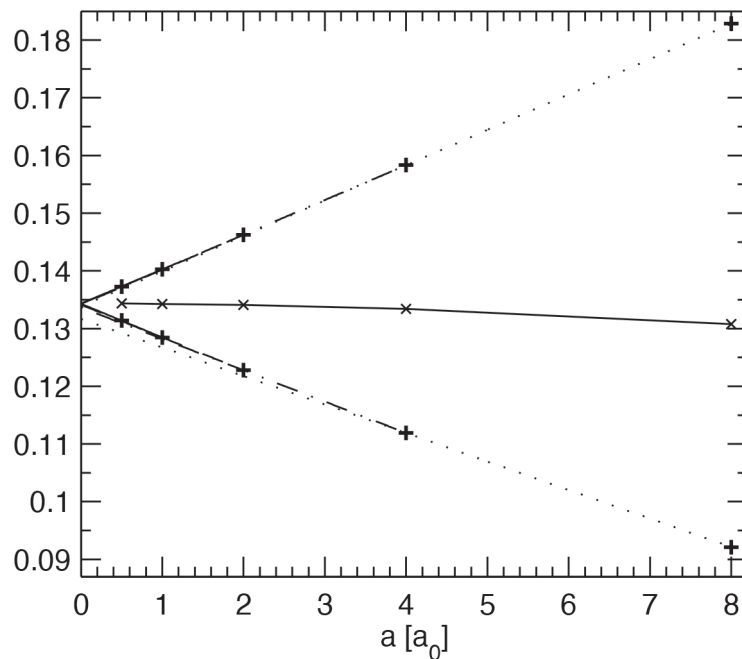


Abbildung 7.5: Die unteren Grenzwerte $\phi(0, a)$ und oberen Grenzwerte $\phi(215, a)$ der Porosität ϕ bei den Auflösungen a gekennzeichnet mit Kreuzen '+'. Die Extrapolationen, jeweils vier für die untere bzw. obere Schranke, wurden auf eine unendlich feine Auflösung $a = 0$ extrapoliert. Die innerhalb der Schranke liegenden Datenpunkte ('x') stammen von einer Wahl des Schwellwerts g_{th} , der die Varianz des Grauerthistogramms maximiert.

In Abbildung 7.6 sind die Mittelwerte

$$\phi^*(c) = \frac{\varphi_0(c) + \varphi_{215}(c)}{2}$$

gegen deren Differenz

$$\phi^{**}(c) = \varphi_{215}(c) - \varphi_0(c)$$

aufgetragen sowie eine lineare Ausgleichsgerade für die Daten. Die Ausgleichsgerade ergibt $\lim_{c \rightarrow 0} \phi^*(c) = 0.134313 \pm 0.00001$.

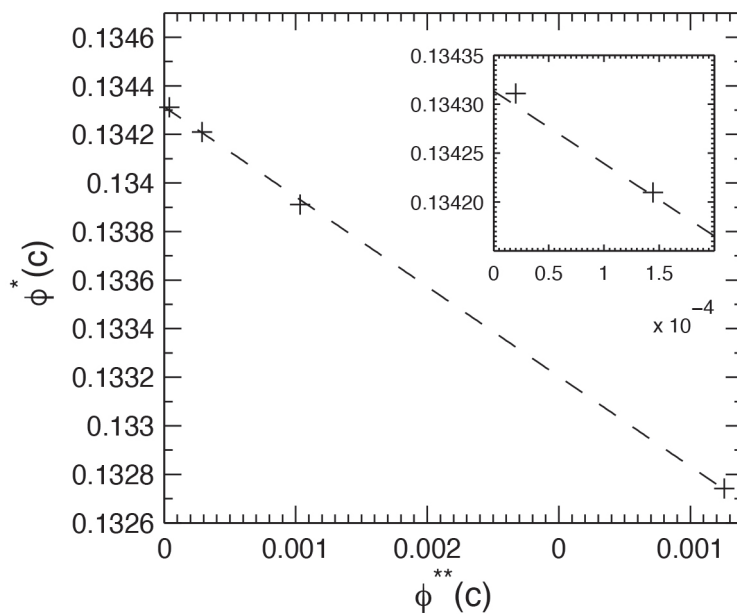


Abbildung 7.6: Lineare Interpolation der Mittelwerte $\phi^*(c)$ aufgetragen gegen die Differenzen $\phi^{**}(c)$ für $c = 8a_0, 4a_0, 2a_0, a_0$.

7.3 Berechnung der spezifischen inneren Oberfläche aus der Grauwertverteilung

Im Folgenden wird beschrieben, wie aus der diskreten Grauwertverteilung $M(g, a)$ und einigen Annahmen die spezifische innere Oberfläche S_V abgeschätzt werden kann. Es wird davon ausgegangen, dass jeder Voxel $\mathbb{V}_i(a)$, bei der Auflösung a , entweder komplett im Poren- bzw. Phasenraum liegt oder aber nur von einer Phasengrenze geteilt wird, die als Ebene approximiert werden kann. Der Grauwert der Voxel ist $0 < g_i(a) < 216$.

Es sei $\mathbb{B}_i(R)$ eine Kugel mit Radius R und dem Mittelpunkt im Zentrum des Voxel $\mathbb{V}_i(a)$. Dann ist $\mathbb{B}_i(a/2)$ die einbeschriebene Kugel mit Radius $a/2$ und $\mathbb{B}_i(\sqrt{3}a/2)$ die umschreibende Kugel mit Radius $\sqrt{3}a/2$. Die Fläche einer den Voxel schneidenden Ebene ist dann nach oben beschränkt durch die größtmögliche Grundfläche, die bei dem Schnitt durch die umschriebene Kugel entstehen kann. Es sei b der Radius des Kreises der beim Schnitt einer Ebene mit einer Kugel mit Radius R entsteht. Die Höhe der größeren Kugelschale sei h .

$a[a_0]$	8	4	2	1	0.5
$S_{max}(a) [\text{mm}^{-1}]$	20.039	20.530	20.774	20.904	20.946
$S_{mf}(a) [\text{mm}^{-1}]$	9.6371	9.8774	9.9978	10.061	10.065
$S_{min}(a) [\text{mm}^{-1}]$	6.0407	6.1915	6.2673	6.3068	6.3096
$S_{ex}(a) [\text{mm}^{-1}]$	9.168 ± 0.02	9.825 ± 0.06	10.159 ± 0.16	$10.307(\pm 0.43)$	$10.355(\pm 1.34)$

Tabelle 7.3: Obere und untere Schranken $S_{max}(a)$ und $S_{min}(a)$ aus den Gleichungen 7.3.1, 7.3.2. $S_{mf}(a)$ ist die nach den Gleichungen 7.3.3-7.3.8 berechnete 'mean-field'-Abschätzung. $S_{ex}(a)$ wurde durch den in [52] beschriebenen Algorithmus berechnet.

Es gilt dann $b^2 = h(2R - h)$ und $V = \frac{\pi}{3}h^2(3R - h)$ ist das Volumen der Kugelschale. Mit der Porosität des Voxel

$$\psi = \frac{3V}{4\pi R^3}$$

kann aufgelöst werden nach h

$$h_m(\psi) = R \left\{ 1 + 2 \cos \left[\frac{\arccos(1 - 2\psi) + 2m\pi}{3} \right] \right\}$$

mit $m = 0, 1, 2$. Durch die geometrisch gegebene Randbedingung $0 \leq h \leq R$ ist nur die Lösung

$m = 2$ sinnvoll. Die Bodenfläche der Kugelschale ist $A = \pi b^2$ und damit gilt für die kreisförmige Bodenfläche als Funktion der Porosität und des Kugelradius R

$$A(\psi, R) = \pi h_2(\psi) [2R - h_2(\psi)].$$

Mit der allgemeinen Formulierung der Porosität aus Gl. 7.1.1 ergibt sich die obere Grenze der spezifischen inneren Oberfläche zu

$$S_{max}(g, a) = \frac{1}{a^3} A \left(1 - \frac{g}{216}, \frac{\sqrt{3}a}{2} \right) \quad (7.3.1)$$

für einen Voxel mit Grauwert g bei einer Auflösung a .

Ebenso lässt sich eine untere Grenze finden, wenn man die Bodenfläche der einbeschriebenen Kugel $\mathbb{B}_i(a/2)$ mit Radius $a/2$ betrachtet. Die kleinstmögliche Fläche ergibt sich für Kugelschnitte mit einer Ebene, die senkrecht zu einer der Raumdiagonalen des Voxel ist. Es ist zu beachten, dass die einbeschriebene Kugel nur von der Ebene geschnitten wird, wenn der Größenabstand der Ebene zu einer Voxelcke größer ist als $a(\sqrt{3} - 1)/2$, wenn also die Porosität mindestens $\psi_{min} = \sqrt{3}(\sqrt{3} - 1)^3/16 \approx 0.042468$ ist oder maximal $\psi_{max} = 1 - [\sqrt{3}(\sqrt{3} - 1)^3/16] \approx 0.95753$. Es gibt also eine untere Grenze von $S_{min}(g, a) = 0$ für $g \leq 9$ und $g \geq 206$. Für alle anderen Grauwerte $10 \leq g \leq 205$ gilt

$$S_{min}(g, a) = \frac{1}{a^3} A \left(1 - \frac{g}{216}, \frac{a}{2} \right). \quad (7.3.2)$$

Die Grenzwerte können zusammengefasst werden als

$$S_b(a) = \frac{1}{N(a)} \sum_{g=1}^{215} M(g, a) S_b(g, a) \quad (7.3.3)$$

mit $b = min$ oder $b = max$. In der Tabelle 7.3 stehen die Ergebnisse für $S_{min}(a)$ und $S_{max}(a)$ für $a = 8a_0, 4a_0, 2a_0, a_0, a_0/2$.

Da die berechneten Schranken recht weit auseinanderliegen und daher für eine direkte Abschätzung der spezifischen inneren Oberfläche ungeeignet sind, wird eine Mittelung mit weiteren realistischen Annahmen verwendet.

Wie bereits erwähnt, wird bei $a \rightarrow 0$ jedes Voxel von höchstens einer Ebene geschnitten. Diese Ebe-

ne kann jedoch verschiedene Orientierungen haben. Bei einer sehr großen Anzahl von Voxel kann argumentiert werden, dass sich die Orientierungen heraus mitteln. Daraus entsteht die im weiteren verwendete Annahme, dass die Schnittebene eines Voxel immer senkrecht zu einer Raumdiagonalen verläuft. Es kann im Weiteren eine der drei Raumdiagonalen mit der dazugehörigen Voxelecke betrachtet werden, ohne an Allgemeinheit zu verlieren.

Es sei $0 \leq d \leq \sqrt{3}a/2$ der kleinere Abstand der Ebene von der Voxelecke. Für $0 \leq d \leq a/\sqrt{3}$ ist die Schnittfläche ein gleichschenkliges Dreieck und in diesem Fall gilt der Zusammenhang zwischen d und der Porosität ψ :

$$\frac{d(\psi, a)}{a} = \left(\frac{2(1 - \psi)}{\sqrt{3}} \right)^{1/3} \quad (7.3.4)$$

und die Fläche des Dreiecks ist

$$A(d, a) = \frac{3\sqrt{3}}{2}d(\psi, a)^2 \quad (7.3.5)$$

Für den Bereich $a/\sqrt{3} \leq d \leq \sqrt{3}a/2$ ist die Schnittfläche ein Hexagon und für die entsprechenden Größen gilt

$$\frac{d(\psi, a)}{a} = \frac{\sqrt{3}}{2} - \cos \left[\frac{\arccos(2(2\psi - 1)/\sqrt{3}) + \pi}{3} \right] \quad (7.3.6)$$

und

$$A(d, a) = \frac{\sqrt{3}}{2} \left[3d^2 - 3(\sqrt{3}d - a)^2 \right] \quad (7.3.7)$$

Daraus ergibt sich die spezifische innere Oberfläche eines Voxel mit Grauwert g bei einer Auflösung a zu

$$S_{mf}(g, a) = \frac{A(d(\psi, a), a)}{a^3} \quad (7.3.8)$$

wobei für $1 \leq g \leq 36$ die Gleichungen 7.3.4,7.3.5 verwendet werden, für $1 \leq g \leq 36$ die Gleichungen 7.3.6,7.3.7 mit $\psi = 1 - (g/216)$, für $37 \leq g \leq 108$ mit $\psi = 1 - (g/216)$ ebenfalls die Gleichungen 7.3.6,7.3.7, für $109 \leq g \leq 179$ auch die Gleichung 7.3.6, aber mit $\psi = g/216$ und für $180 \leq g \leq 215$ die Gleichungen 7.3.4,7.3.5.

Des endgültige Ergebnis $S_{mf}(a)$ wird dann aus Gl. 7.3.8 und Gl. 7.3.3 berechnet. Es ist der Tabelle 7.3 in der zweiten Zeile angegeben. Als Vergleich zu der hier angegebenen Abschätzung aus den Grauerthistogrammen und einigen Modellannahmen sind in der Tabelle 7.3 auch die Werte $S_{ex}(a)$

angegeben. Diese wurden mit einem aufwendigen direkten Algorithmus berechnet, siehe [52], der mehr als 1100 CPU-Stunden dauerte. Das Ergebnis $S_{mf}(a)$ hingegen, welches hervorragend mit den 'teuren' Ergebnissen $S_{ex}(a)$ übereinstimmt, kann aus den Grauerthistogrammen quasi instantan berechnet werden.

7.4 Auflösungsabhängige LPT- und Permeabilitätsberechnung

Für diese Analyse wurden die Gesamtprobe von 1.5 cm Seitenlänge sowie diverse Unterproben diskretisiert, siehe Tabelle 7.4. Es wird analysiert, bei welcher Auflösung und bei welcher REV-Größe aussagekräftige Ergebnisse, gegebenenfalls über eine Extrapolation, erhalten werden können. Die Ergebnisse der LPT-Berechnungen sind in Abbildung 7.7 dargestellt. Die lokale Porositätsverteilung erweist sich in diesem Beispiel als sensitive Größe, die beim Übergang auf die Auflösung $a = 5 \mu\text{m}$ eine Verteilung mit signifikant abweichendem kleineren Mittelwert liefert. Es werden ab dieser Auflösung Bereiche als Gesteinsmatrix diskretisiert, die vorher zumeist als Porenraum gewertet wurden. Die Ergebnisse der Permeabilitätsberechnung sind in Abbildung 7.8 dargestellt. Die Gitter-Boltzmann Permeabilitätsberechnungen streuen für die Unterproben stark, selbst bei guten Auflösungen wie $a = 2.5 \mu\text{m}$.

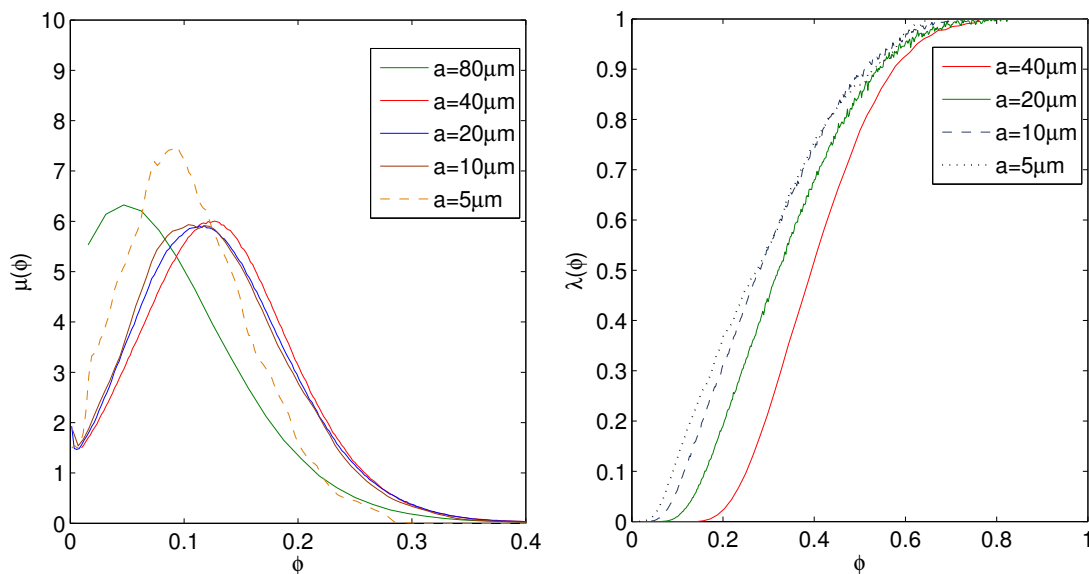


Abbildung 7.7: Auflösungsabhängige LPT-Berechnungen für $L = 5120 \mu\text{m}$. Die lokale Porositätsverteilung (links) zeigt beim Übergang von $a = 10 \mu\text{m}$ auf $a = 5 \mu\text{m}$ eine Verteilung mit einem neuen Erwartungswert. Die lokale Perkolationswahrscheinlichkeit konvergiert mit der Auflösung.

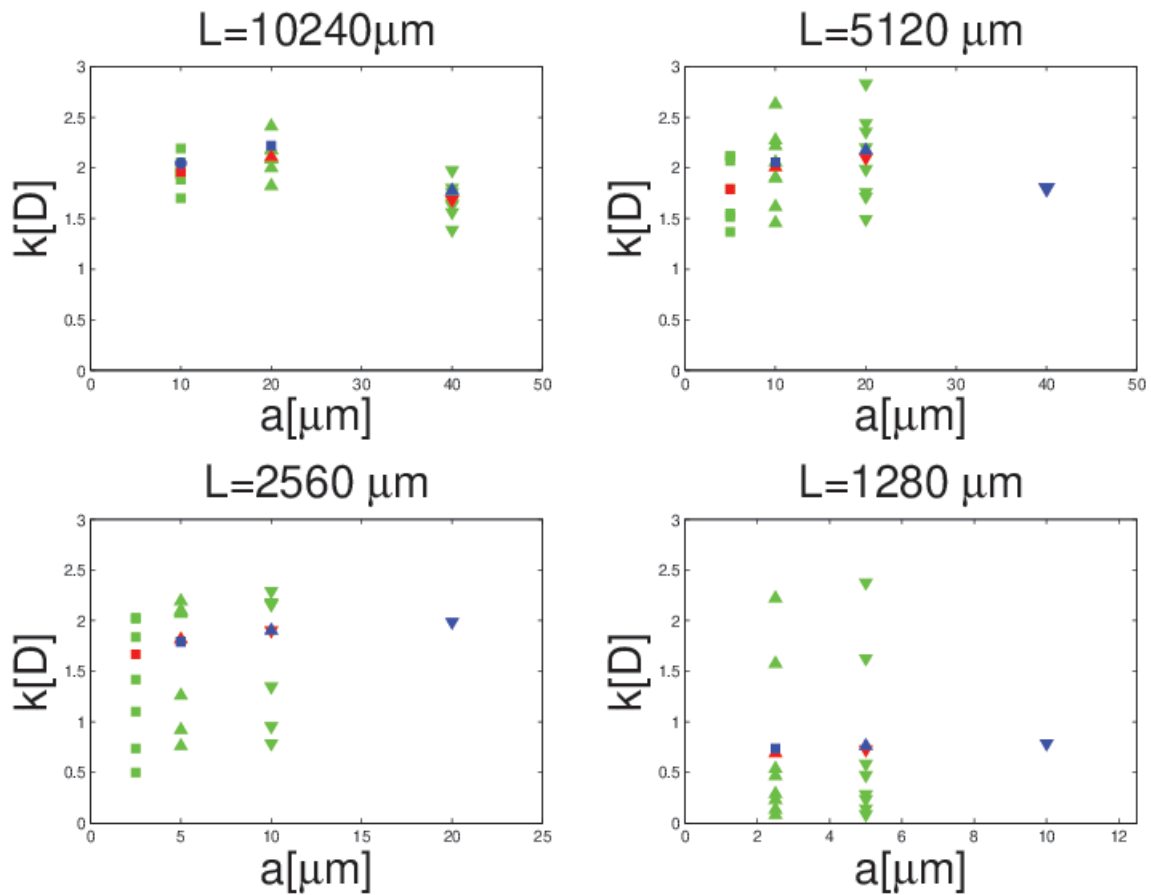


Abbildung 7.8: Auflösungsabhängige Permeabilitätsberechnungen aus Gitter-Boltzmann-Simulationen. Die Auflösung ist a , die Seitenlänge der kubischen Proben ist L und die Permeabilität κ in Darcy. Die verschiedenen Symbole kennzeichnen die Größen, der zur Berechnung verwendeten kubischen Teilgebiete. Blau für die gesamte Probe. Rot für die Hälfte. Das grüne Dreieck für Viertel-Proben, das umgekehrte grüne Dreieck für Achtel-Proben und das grüne Quadrat für Sechzehntel-Proben.

a [μm]	80	40	20	20	10	10	5	2.5
L^* [cm]	1.024	1.024	1.024	0.512	1.024	0.256	0.128	0.064
N	1	1	1	8	1	64		

Tabelle 7.4: L^* ist die Seitenlänge des Kubus bei der Auflösung a . Es wurden N diskretisierte Unterproben erstellt bzw. bei $N = 1$ wurde das Gesamtgebiet diskretisiert.

7.5 Fazit

In diesem Abschnitt wurden Datensätze vorgestellt, die bei der Diskretisierung eines sehr großen realistischen Modells für Fontainebleau-Sandstein bei Auflösungen $a = 117.12 \mu\text{m}, \dots, 0.4576 \mu\text{m}$ entstanden sind. Die Daten wurden der Allgemeinheit zur Verfügung gestellt. Es wurde eine schnelle prädiktive Methode vorgestellt, um aus den Grauwertverteilungen geometrische Größen zur Charakterisierung zu berechnen. Anstatt eine Probe einmalig mit einer sehr guten Auflösung zu diskretisieren und einen aufwendigen Algorithmus zur Bestimmung von geometrischen Größen zu verwenden, ist es effizienter, die Probe bei mehreren Auflösungen zu diskretisieren und eine Berechnungsmethode aus geometrischen und 'mean-field'-Annahmen zu nutzen, die nur die Grauwertstatistiken verwendet, und die Ergebnisse dann zu extrapolieren. Eine weitere sehr sensitive Analyse-Methode, die „differentiale Porosimetrie“, wird in der Arbeit [39] vorgestellt. Darin wird aufgezeigt wie subtile auflösungsabhängige Messgrößen bzw. Inhomogenitäten in der digitalen Probe erkannt werden können.

8 Modellierung und Berechnungen für Bentheimer Sandstein

In diesem Abschnitt wird der oben vorgestellte Algorithmus zur Generierung eines Kontinuumsmodells für Bentheimer Sandstein benutzt, siehe [3]. Als Ausgangsdaten standen Mikrocomputertomographie-Datensätze (2D) und die Verteilungen der Korngrößen zur Verfügung. In diesem Sandstein ist die Bandbreite an unterschiedlichen Korntypen im Vergleich zum oben vorgestellten Fontainebleau-Sandstein groß. Daher wurden hier 512 Korntypen verwendet, anstatt 99 wie beim Fontainebleau-Sandstein. Es kommen viele länglichen Korntypen vor, was die Verwendung eines Algorithmus zur Packung von Kugeln fraglich macht. Die Modellierung mit den zur Verfügung stehenden Algorithmen ist bei dieser Anwendung besonders aufwendig und kritisch. Das entstandene Modell wurde in verschiedenen Auflösungen diskretisiert, es wurden charakteristische Größen wie Porosität und Permeabilität berechnet und Berechnungen im Rahmen der LPT durchgeführt. Die Ergebnisse wurden zur Evaluation mit den experimentellen Datensätzen verglichen.

8.1 Analyse und Aufbereitung der experimentellen Daten

Der Originaldatensatz bestand aus 831 2D-Schnitten eines μCT -Datensatzes mit einer Auflösung von $3.7 \mu m$, siehe Abb. 8.1. Jedes Bild besteht aus 2048×2048 Voxel, wobei jedes Voxel einen Grauwert repräsentiert. Ein Voxel wird auf Datenebene durch eine 4 Byte (32 Bit)-Gleitkommazahl kodiert, die Größe eines Bildes ist also 16 MB und die des gesamten Datensatzes, also $2048 \times 2048 \times 831$ Voxel, fast 13 GB groß. Des Weiteren standen elektromikroskopische Aufnahmen (SEM) zur Verfügung in denen die Körnerformen und deren Größenverteilung optisch besonders gut zu erkennen sind, siehe Abb. 8.68.7. Für die Modellierung und Berechnungen wurden die original μCT -Datensätze

zentriert zugeschnitten auf eine Größe von 1024x1024. Es wurden nur 512 Querschnitte verwendet, da auf den restlichen Schnitten Artefakte der Aufnahme zu sehen waren. Nach dem Zuschnitt wurden die Daten segmentiert und dabei die Bereiche, Körner, Füllungen, Zementierungen und Porenraum definiert, siehe Abb. 8.2. In einem komplizierten Verfahren, bei dem verschiedene Filterfunktionen und Segmentierungstechniken, siehe [30], [72] und [62], angewandt wurden, siehe [3] für Details, wurden die Volumenanteile der definierten Bereiche ermittelt. Die Ergebnisse sind in Tabelle 8.1 angegeben. Für die weiteren Berechnungen der Minkowski-Funktionale, Größen aus der LPT und für die Permeabilität werden hier die Ergebnisse für eine Segmentierung nach der Methode von Otsu [62] verwendet.

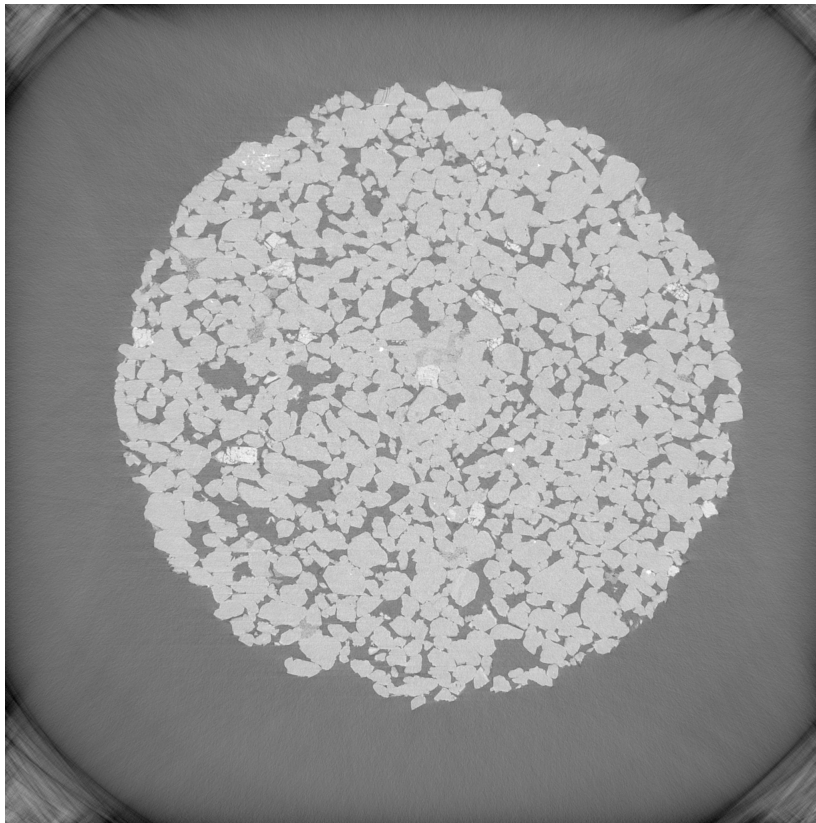


Abbildung 8.1: 2D-Schnitt des μCT -Datensatzes bei einer Auflösung von $3.7 \mu m$. Die Bildgröße ist 2048×2048 . Die verschiedenen Gesteinsregionen sind hellgrau der Porenraum ist dunkelgrau.

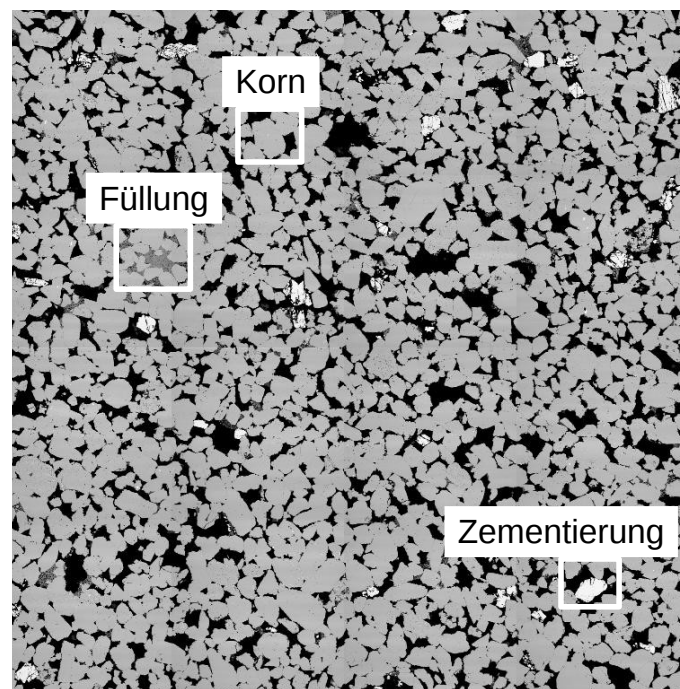


Abbildung 8.2: Segmentierter und auf 1024x1024 zurechtgeschnittener μCT -Datensatz. Der Porenraum ist schwarz und die verschiedenen Gesteinsregionen sind unterschiedlich grau. Es wurden drei Gesteinsregionen definiert. Die Körner, die den Hauptanteil an der Gesteinsmatrix haben, sind hellgrau dargestellt, die Füllungen aus Sedimenten in dunklerem Grau und die Zementierungen des Quarzes in Weiß.

Korn-Anteil	Sedimentierung	Zementierung	Porenraum
59.6 %	0.93 %	0.97 %	21.4 %

Tabelle 8.1: Verteilungen der verschiedenen definierten Volumenteile

Die Ergebnisse der Minkowski-Funktionale für die spezifische Oberfläche S_V , mittlere Krümmung K_V und die totale Krümmung T_V sind in der Tabelle 8.2 angegeben. Die Ergebnisse aus der LPT, bei Verwendung einer kubischen Messzelle der Seitenlänge L , die lokale Porosität $\mu(\phi, L)$, die lokalen Perkolationswahrscheinlichkeiten $\lambda(\phi, L)$ sowie der Gesamtanteil perkolierender Messzellen $p(L)$ sind in den Abbildungen 8.3, 8.4 und 8.5 dargestellt. Als Transportgröße wurde die Permeabilität κ mittels Gitter-Boltzmann-Simulationen bestimmt zu $\kappa = 3090 mD$.

Spezifische Oberfläche S_V [mm^{-1}]	Mittlere Krümmung K_V [mm^{-2}]	Totale Krümmung T_V [mm]
17.3802	298.743	-16154.4

Tabelle 8.2: Die spezifische Oberfläche S_V , das spezifische Integral der mittleren Krümmung K_V und das spezifische Integral der totalen Krümmung T_V der segmentierten experimentellen Daten des Bentheimer Sandsteins.

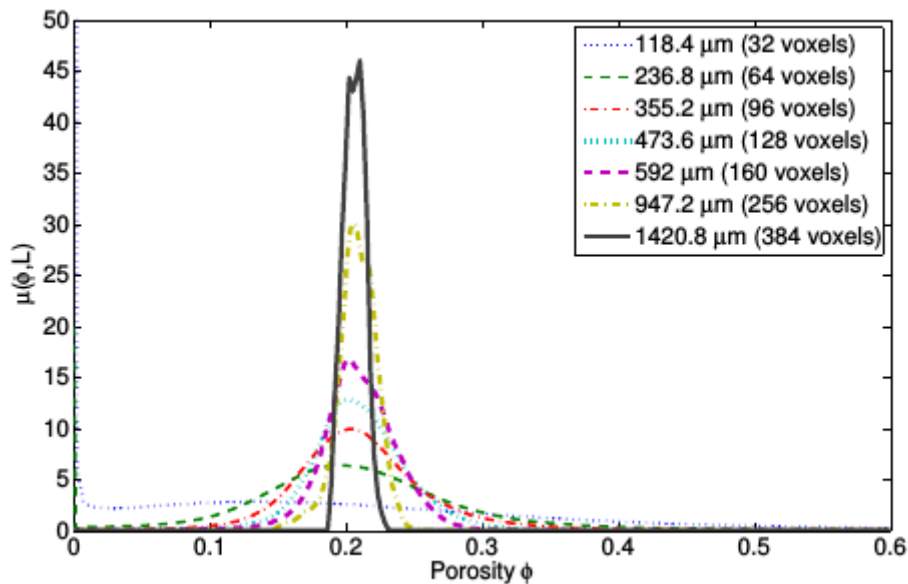


Abbildung 8.3: Lokale Porosität $\mu(\phi, L)$ des experimentellen Bentheimer Datensatzes für kubische Messzellen mit Seitenlängen L von 32-384 Voxel.

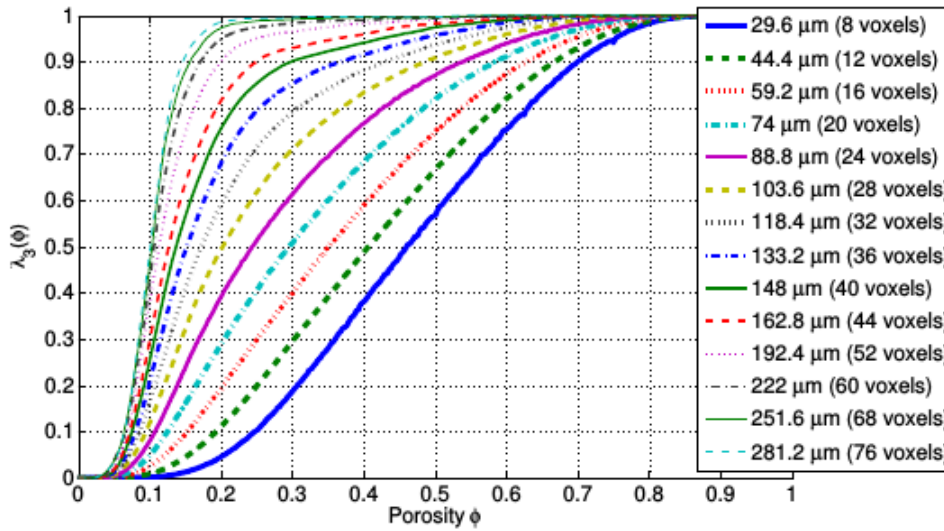


Abbildung 8.4: Lokale Perkolationswahrscheinlichkeiten $\lambda(\phi, L)$ des experimentellen Bentheimer Datensatzes für kubische Messzellen mit Seitenlängen L von 8-76 Voxel.

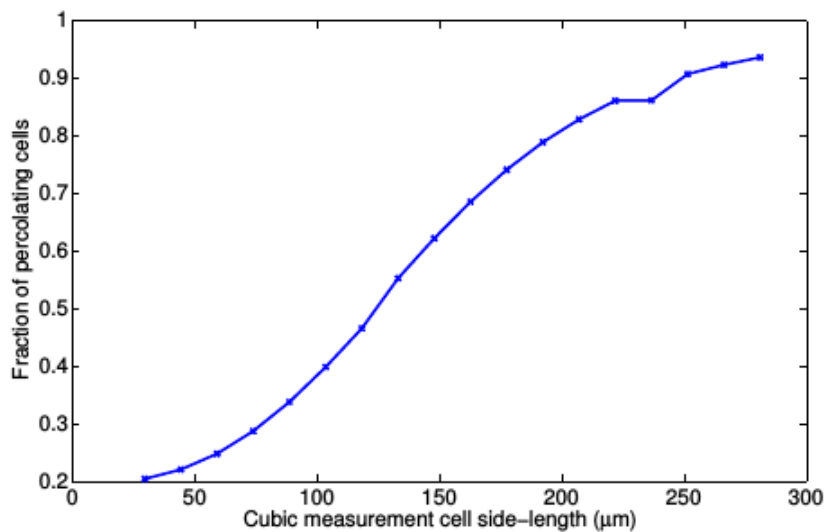


Abbildung 8.5: Der Gesamtanteil perkolierender Messzellen $p(L)$ in Abhängigkeit von der Seitenlänge L [μm] einer kubischen Messzelle.

8.2 Modellierung des Bentheimer Sandsteins

8.2.1 Generierung neuer Quarztypen

Die Quarztypen, welche z. B. bei der Modellierung von Fontainebleau-Sandstein ihre Verwendung finden, eignen sich nicht zur Modellierung von Bentheimer Sandstein. Diese ursprünglichen Quarztypen sind von kugelhähnlicher isotroper Form und haben nur eine geringe Bandbreite an unterschiedlicher Oberflächenstruktur. Im Bentheimer Sandstein existiert jedoch eine große Bandbreite an Körnern unterschiedlicher Elongation und Oberflächenbeschaffenheit, siehe Abb. 8.8. Im Rahmen der Modellierung wurden Korntypenklassen generiert, die aus 36-121 Facetten bestehen. Für jede Korntypenklasse mit einer festen Anzahl an Facetten wurden verschiedene Bandbreiten an Elongationen generiert. Für diese Parametervariationen wurden über 30 kleine Modelle generiert, um einen sinnvollen Parametersatz zu finden.

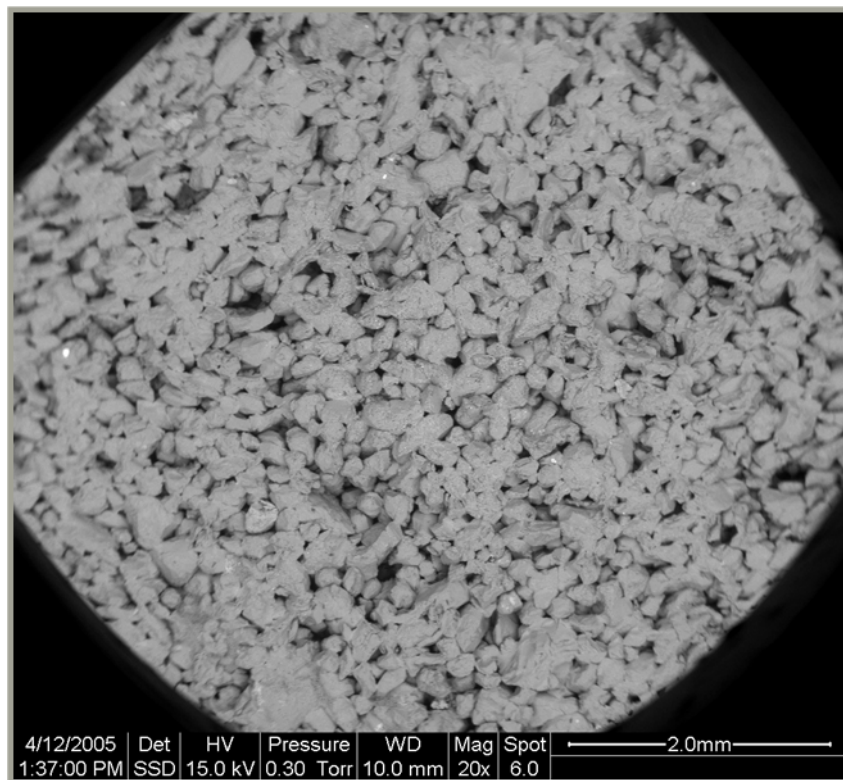


Abbildung 8.6: 2D-SEM-Datensatz bei einer 20-fachen Vergrößerung. Hier sind die Packungsdichte und die Orientierung der Körner optisch gut zu erkennen.

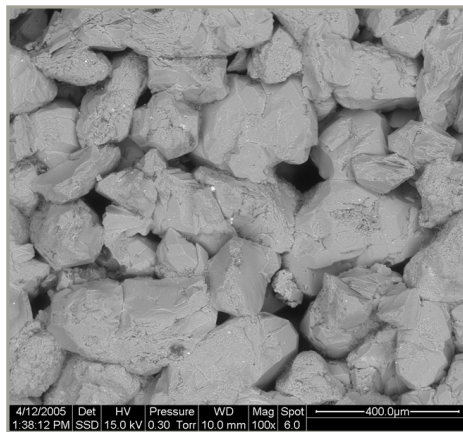


Abbildung 8.7: 2D-SEM-Datensatz bei einer 100-fachen Vergrößerung. Die einzelnen stark unterschiedlichen Korntypen sind klar zu erkennen.

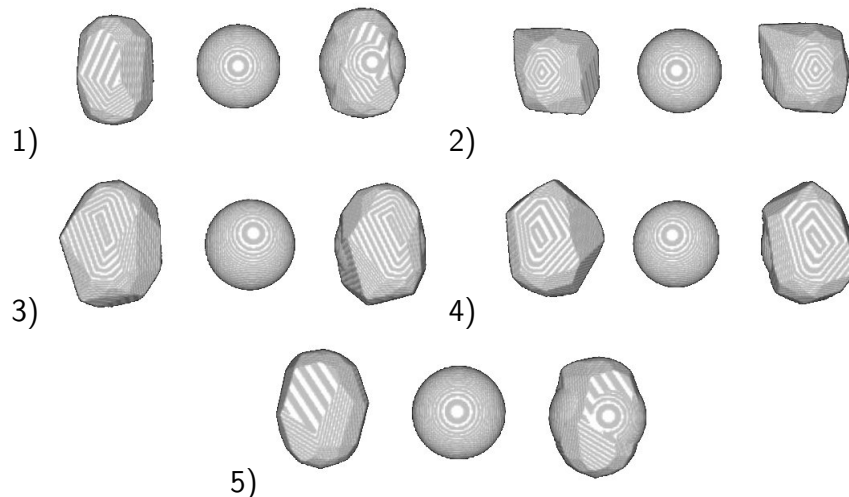


Abbildung 8.8: Fünf exemplarisch ausgewählte Korntypen. Links die Körner, in der Mitte die entsprechenden Kugeln, und Rechts deren Überlagerung. Deutlich zu sehen ist, wie das Konzept der Kugelpackung hier an seine Grenzen gerät.

8.2.2 Anisotrope 'lokale' Dekoration der Kugelpackung

Für diese Anwendung wurde der Algorithmus zur Dekoration der Kugelpackung angepasst. Anders als bisher sind die Orientierungen der einzelnen Körner jetzt abhängig von der Kugelgröße und von

der lokalen Umgebung. Dieser neue iterativ arbeitende Algorithmus trägt den Namen 'Overlapp Minimizer' (OM). Dabei wird das Simulationsgebiet in sich überlappende Zellen mit Volumen V^{OM} unterteilt. Die Größen und Orientierungen der Körner werden in mehreren Iterationen optimiert. Dabei werden nur Körner innerhalb der selben Zelle berücksichtigt. Der Algorithmus skaliert daher global linear mit der Anzahl der benötigten Körner. Die Optimierung verwendet eine fest vorgegebene Anzahl an Iterationen pro Korn N_K^{OM} und pro Zelle N_Z^{OM} . Für Details siehe [3].

8.2.3 Final verwendete Parameter bei der Modellierung

Nachdem in kleineren Modellierungen die neuen Korntypen und Dekorationsstrategien festgelegt wurden, siehe Abschnitte 8.2.1, 8.2.2, wurden weitere 80 Modelle der vollen Größe konstruiert und evaluiert, um die folgende optimale Parameterwahl festzulegen, siehe Tabelle 8.3. Ein qualitativer optischer Vergleich einiger binärer Querschnitte des Modells und des experimentellem Datensatzes ist in Abb. 8.9 zu sehen.

Parameter	Wert
Dichte der Kugelpackung	556 Kugeln/mm ³
Kugelgrößen	45-280 μm
maximale Überlappung σ	0.1
OM Zellvolumen V^{OM}	500 μm^3
OM Iterationen pro Zelle N_Z^{OM}	3
OM Iterationen pro Korn N_K^{OM}	5
Facetten pro Korn	36
Maximale Elongation der Körner	0.4
Anzahl an Korntypen	512

Tabelle 8.3: Verwendete Parameter für das finale Bentheimer Modell

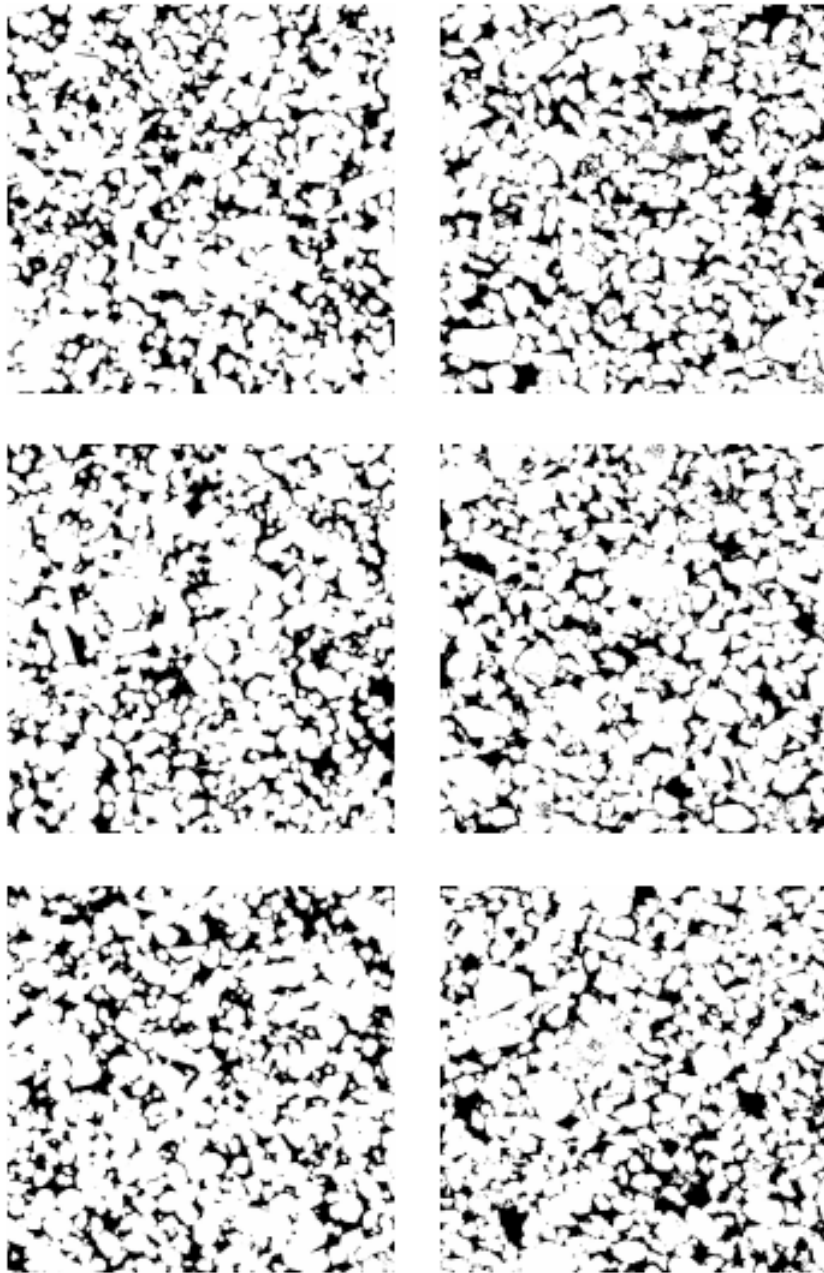


Abbildung 8.9: Optischer Vergleich zwischen binären Querschnitten des Bentheimer Modells (links) und den experimentellen Daten (rechts). Beide Datensätze wurden mit der Otsu-Methode segmentiert. Der Porenraum ist schwarz und die Gesteinsmatrix weiß.

8.3 Analyse und Vergleich mit den experimentellen Daten

Daten

Zum Vergleich wurde das Modell bei einer Auflösung von $3.7 \mu\text{m}$ diskretisiert, genau wie die experimentellen Daten, und mit dem Otsu-Verfahren segmentiert. Als Bewertung des Modells wurden die spezifische Oberfläche S_V , die mittlere Krümmung K_V und die totale Krümmung T_V berechnet, siehe Tabelle 8.4.

Daten	ϕ	$S_V [\text{mm}^{-1}]$	$K_V [\text{mm}^{-2}]$	$T_V [\text{mm}]$
Modell	0.21643	15.6476	138.101	-4194.72
Experiment	0.21468	17.3802	298.743	-16154.4

Tabelle 8.4: Vergleich der Porosität ϕ , spezifische Oberfläche S_V , mittlere Krümmung K_V und totalen Krümmung T_V zwischen Modell und Experiment.

Aus der LPT wurden die lokale Porosität $\mu(\phi, L)$, die lokalen Perkolationswahrscheinlichkeiten $\lambda(\phi, L)$ sowie der Gesamtanteil perkolierender Messzellen $p(L)$ berechnet und mit den experimentellen Daten verglichen, siehe folgende Abbildungen.

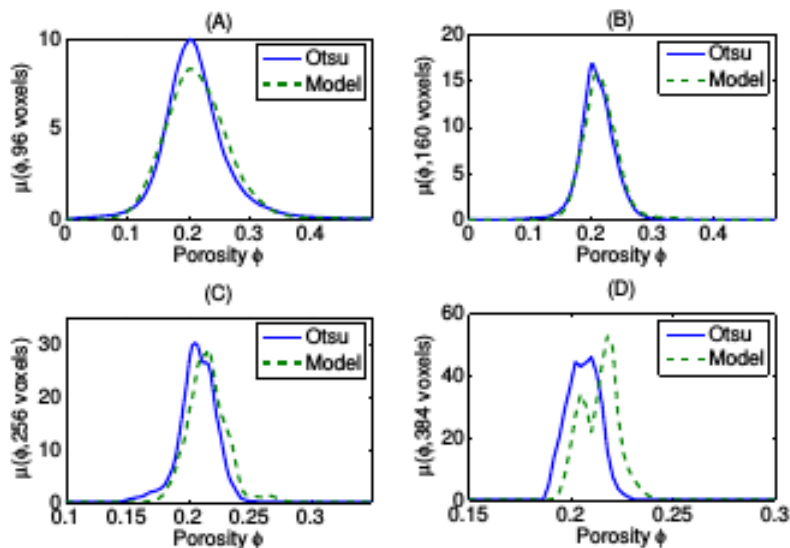


Abbildung 8.10: Der Vergleich lokaler Porositäten $\mu(\phi, L)$ für verschiedene Messzellengrößen $L = 96(A), 160(B), 256(C), 384(D)$ bei einer Auflösung von $3.7 \mu\text{m}$ zeigt eine gute Übereinstimmung. Bei großen Messzellen (Bild D) gibt es eine starke Abweichung.

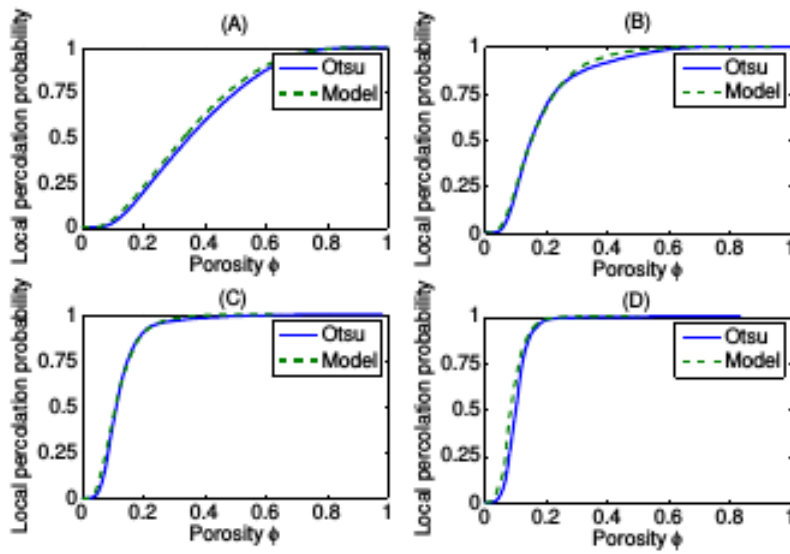


Abbildung 8.11: Der Vergleich der lokalen Perkolationswahrscheinlichkeiten $\lambda(\phi, L)$ für verschiedene Messzellengrößen $L = 16(A), 36(B), 52(C), 72(D)$ bei einer Auflösung von $3.7 \mu\text{m}$ zeigt eine sehr gute Übereinstimmung.

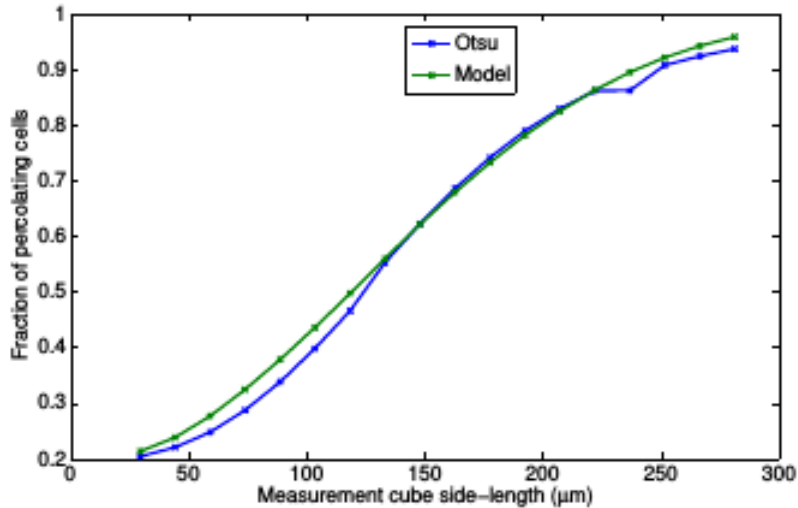


Abbildung 8.12: Der Vergleich des Gesamtanteils der perkolierenden Messzellen $p(L)$ bei einer Auflösung von $3.7 \mu\text{m}$ zeigt eine gute Übereinstimmung.

Eine Gitter-Boltzmann-Vergleichssimulation zwischen den experimentellen Daten und dem diskretisierten Modell ergab gute Übereinstimmung, siehe Tabelle 8.5.

Ergebnis	Experiment	Simulation
Mittlere Flussgeschwindigkeit [$\mu\text{m/s}$]	1.34×10^{-3}	1.26×10^{-3}
Varianz der Flussgeschwindigkeit [$\mu\text{m/s}$]	7.25×10^{-6}	6.17×10^{-6}
Permeabilität [mD]	3074	2872

Tabelle 8.5: Vergleich der Gitter-Boltzmann-Simulation zwischen Modell und experimentellen Daten. Die Werte für die Geschwindigkeiten und die Permeabilität stimmten gut überein.

8.4 Fazit

Durch die aufwendige Generierung neuer Quarztypen und die Erweiterung der Dekorationsmethode hin zu einem anisotropen, 'lokalen' und iterativen Algorithmus konnte mit dem DEM-Algorithmus ein gutes Modell für Bentheimer Sandstein erstellt werden. Der Aufwand für die Erstellung eines solchen kalibrierten Modells ist hoch. Dennoch können solche kalibrierten Modelle dann für eine Vielzahl von Simulationen und Berechnungen verwendet werden. Es wurde im Rahmen der Arbeit aber auch deutlich, dass die Restriktion des DEM-Packungsalgorithmus auf Kugeln letztendlich seine Grenzen hat und eine Erweiterung auf anisotrope unterschiedliche Objekte sehr vielversprechend scheint.

9 Anhang

9.1 Veranschaulichung eines Simulationsschrittes im DEM-Algorithmus

Detaillierter Ablauf eines DEM-Simulationsschrittes am Beispiel einer einzelnen zweidimensionalen Domäne (rot berandet) mit inneren Zellen (blau), Randzellen (grün) und Replikazellen (grau). Graue Replikazellen entsprechen grünen Randzellen benachbarter Domänen.

1. 'Rand-Replika-Schritt': Jede Domäne sendet die Daten aller Kugeln, die sich in ihren Randzellen befinden, an benachbarte Domänen. Dort werden die Kugeln in die entsprechenden Replikazellen sortiert. Wie oben beschrieben, reichen dazu 6 Punkt-zu-Punkt-Kommunikationen zwischen den Domänen aus. Bild 9.1.
2. 'Kraft-Schritt': In jeder Domäne wird in allen inneren Zellen die Kraftberechnung wie oben beschrieben durchgeführt. Dazu werden Techniken mit Abschneideradien, Nachbarschaftslisten und weitere Steuerungsmechanismen verwendet. Für diesen Schritt müssen lokal auch die Kugeldaten aus den Replikazellen vorliegen, die im vorhergegangenen Schritt aktualisiert wurden. Bild 9.2.
3. 'Geschwindigkeits-Schritt': Allen Kugeln wird eine Geschwindigkeit zugeordnet. Auch hier können Regelungsmechanismen korrigierend eingreifen. Die Replikazellen werden 'geleert'. Bild 9.3.
4. 'Verschiebe-Schritt': Neue Kugelpositionen werden durch den Strömer-Verlet-Algorithmus berechnet. Dadurch werden im Allgemeinen auch die Replikazellen neu belegt durch Kugeln, die aus den Randzellen einwandern. Bild 9.4.
5. 'Replika-Rand-Schritt': Die Daten der Kugeln aus den Replikazellen werden an die zuständigen Domänen gesendet. Auch hier sind 6 Punkt-Zu-Punkt-Kommunikationen ausreichend. In den

Domänen werden die ankommenden Kugeln in die entsprechenden Randzellen einsortiert und die Wechselwirkungslisten werden eventuell aktualisiert. Bild 9.5.

6. 'Replika-Leeren': Die Replikazellen jeder Domäne werden geleert. Bild 9.6.

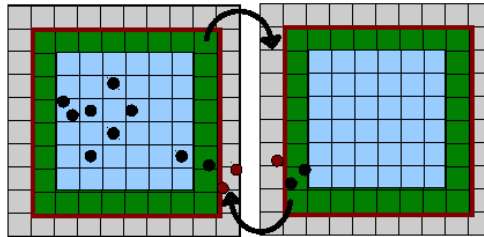


Abbildung 9.1: Rand-Replika-Schritt: Rand-Kugeln (schwarz) werden zu Replika-Kugeln (braun).

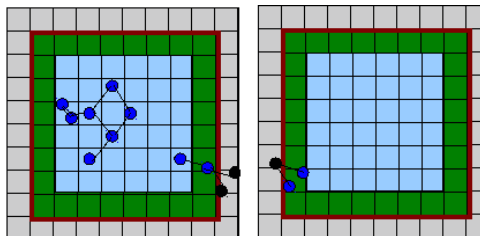


Abbildung 9.2: Kraft-Schritt: Für alle internen Kugeln (blau) wird unter Berücksichtigung der Wechselwirkungslisten, hier symbolisiert durch Linien zwischen Kugeln, eine wirkende Gesamtkraft ermittelt.

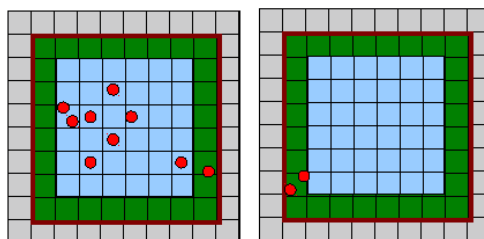


Abbildung 9.3: Geschwindigkeits-Schritt: Alle internen Kugeln erhalten eine neue Geschwindigkeit, symbolisiert durch rote Einfärbung. Die Replikazellen werden geleert.

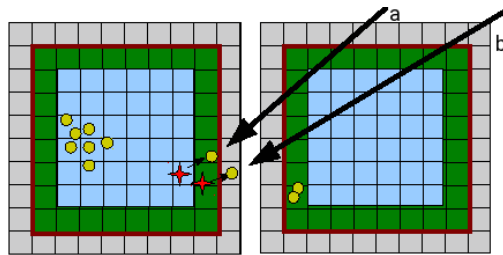


Abbildung 9.4: Verschiebe-Schritt: Die neuen Kugelpositionen werden berechnet. Dabei können Kugeln in Randzellen, am Beispiel (a), und in Replikazellen, Beispiel (b), einwandern.

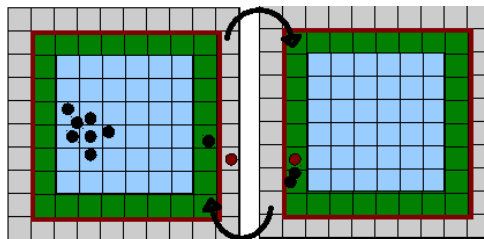


Abbildung 9.5: Replika-Rand-Schritt. Kugeln aus Replikazellen werden zu Kugeln in Randzellen in der benachbarten Domäne, hier am Beispiel einer braunen Kugel, die von der linken in die rechte Domäne wandert.

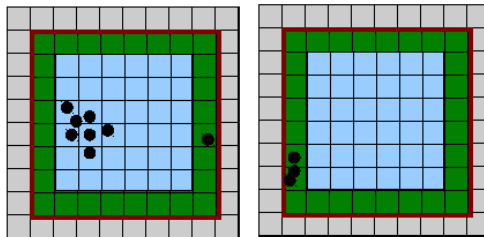


Abbildung 9.6: Replika-Leeren: Die Replikazellen müssen am Ende des Zyklus geleert werden.

9.2 Technische Dokumentation des DEM-Programms 'PackDEM'

Das Programm 'PackDEM' ist die parallele Implementierung des in Abschnitt 5 vorgestellten Algorithmus.

9.2.1 Einführung in das Message-Parsing-Interface und dessen Verwendung in 'PackDEM'

Für die parallele Implementierung wurden eine Gebietszerlegung mit Inter-Prozess-Kommunikationen ausgewählt. Als technologische Plattform für diese Konzepte wurde der MPI-Standard 2.1 gewählt [24]. Dieser Standard definiert mögliche Datentypen, Kommunikationsformen, Gebietszerlegungen und die dabei angenommene Topologie sowie die mögliche dynamische Erschaffung oder Abschaltung von MPI-Prozessen. Bei den Kommunikationsformen wird prinzipiell unterschieden zwischen 'Point-To-Point' Kommunikationen bei denen immer nur ein Sender und ein Empfänger beteiligt sind, den 'Collective-Communications', bei denen als Sender und Empfänger mehrere MPI-Prozesse beteiligt sind und sogenannte 'One-Sided-Communications', bei denen MPI-Prozesse Datenstrukturen anbieten und diese von anderen MPI-Prozessen ohne vorherige Absprache beschrieben oder gelesen werden dürfen. Auch ist es möglich, MPI-Prozesse in sogenannte Gruppen und Kommunikatoren einzuteilen und dadurch Zusammengehörigkeiten und Rollenkonzepte zu definieren. Weiterhin werden bei vielen Kommunikationsformen eine sogenannte blockierende Version ('blocking') und eine nicht blockierende ('non-blocking') Variante angeboten. Bei den blockierenden Varianten wartet der sendende MPI-Prozess innerhalb des MPI-Funktionsaufrufs auf eine Empfangsbestätigung, während er bei den nicht-blockierenden direkt nach dem Absenden in den aufrufenden Programmcode zurückkehrt. Bei nicht blockierenden Kommunikationen liegt es an den MPI-Prozessen selber, gegebenenfalls in einem zweiten Kommunikationsschritt sicherzustellen, dass alle Daten übertragen wurden.

Das Programm 'PackDEM' nutzt zur Gebietszerlegung und Aufteilung der Gebiete auf die vorhandenen Recheneinheiten die innerhalb von MPI genannten 'Cartesian Constructors' und die dazu gehörenden Hilfsfunktionen [24], siehe Seiten 244-254. Als Kommunikationsmechanismen werden sowohl blockierende als auch nicht blockierende Varianten genutzt. Innerhalb der zeitkritischen Schleifen verwendet das Programm nicht blockierende Kommunikationen und für die Synchronisation am Anfang

bzw. am Ende der Simulation sowie zu Synchronisationsabfragen wird auf blockierende Varianten zurückgegriffen.

Der MPI-Standard in seinen verschiedenen Versionen wird von einem Komitee verabschiedet, siehe 'www.mpi-forum.org'. Es gibt zahlreiche Interessengruppen, Universitäten und industrielle Firmen, die dessen Implementierung auf den verschiedenen Hardware-Plattformen und in unterschiedlichen Programmiersprachen anbieten. Unter den frei verfügbaren Quellcode offenen Implementierungen sind am weitestverbreiteten die 'MPICH' bzw. 'MPICH2' -Implementierung, zu finden auf 'www.mpich.org' und die 'OpenMPI' Implementierung, die unter 'www.open-mpi.org' zu finden ist. Das Programm PackDEM wurde mit 'MPICH2' und mit 'OpenMPI' getestet. Technisch gesehen bringen beide Implementierungen eine eigene Version eines C++ Compiler mit, der die nötigen Bibliotheken einbindet und der den Code für das Starten eines MPI-Programms aufbereitet.

9.2.2 Voraussetzungen und Kompilierung des ausführbaren Programms 'PackDEM'

Damit das Programm 'PackDEM' kompiliert und ausgeführt werden kann, müssen folgende Voraussetzungen gegeben sein.

1. Die Bibliothek GNU Scientific Library, kurz GSL, muss installiert sein. Diese wird für einige Statistiken und für Zufallszahlen genutzt.
2. Die Bibliothek LibConfig muss installiert sein. Diese Bibliothek liest die Konfigurationsdatei ein und macht den Inhalt als Datenstruktur verfügbar.
3. Es muss eine MPI-Implementation für den Versionsstandard 2.1 verfügbar sein.
4. Es muss ein C++11 Compiler (C++ Code in der Version 11) verfügbar sein. Getestet wurde das Programm mit dem Intel C++ und dem GNU C++ Compiler.

9.2.3 Programmaufruf und Beschreibung der Konfigurationsdatei

Das Programm wird auf einer MPI-fähigen Plattform durch folgenden Aufruf gestartet:

```
> mpiexec -np NMPI ./packdem
```

worin NMPI die Anzahl der zu startenden MPI-Prozesse angibt. Beim Starten lesen die einzelnen MPI-Prozesse die Datei „pack.cfg“, in der die Einstellungen für die Simulation stehen. Die MPI-Umgebung muss dafür sorgen, dass diese Datei von allen Prozessen gefunden wird. Im Allgemeinen wird dazu der lokale Pfad als Netzlaufwerk zur Verfügung gestellt oder die entsprechenden Dateien werden als Paket auf die entsprechende Hardware kopiert.

Als Format für die Konfigurationsdatei wurde der in dem Projekt „LibConfig“ [1] definierte Standard verwendet. In diesem Projekt sind die Konfigurationseinstellungen typischer, strukturiert und in einer gut lesbaren Syntax definiert. Für das Lesen und Schreiben dieser Konfigurationsdateien existieren Bibliotheken für die Sprachen C und C++ und für alle POSIX- und Windows-Betriebssysteme. Die allgemeine Zuweisung erfolgt mit einzelnen

Feldname = Wert;

Statements. In folgender Aufzählung sind die möglichen Feldnamen und Werte angegeben. Der Wert 'String' steht für alle gültigen Zeichenfolgen innerhalb von zwei Anführungszeichen, also z. B. „String01“. Gleitkommazahlen, folgend 'Float' genannt, können allgemein oder normiert in exponentialer Schreibweise angegeben werden also z. B. 123.456 oder 1.23456e2. Ganzzahlige Werte werden folgend 'Int' genannt. Boolesche Werte, folgend 'bool' genannt, können als 'true' oder 'false' angegeben werden. Gruppen, mit Syntax '{Name1=Wert1; Name2=Wert2;}', sind assoziative Container, die Objekte zusammenfassen die innerhalb der Gruppe einen eindeutigen Namen haben müssen. Ihre Werte können wiederum Gruppen, Listen oder skalare Werte sein. Listen, mit Syntax '(Objekt1, Objekt2,...)', fassen Objekte zusammen, die wiederum Listen, Gruppen oder Werte sein können. Sie werden über ihren Index in der Liste referenziert. Ein Vektor mit Syntax '[Wert1,Wert2,...]' fasst skalare Werte von identischem Typ zusammen.

title = String

Name des Simulationslaufes als String. Beispiel: "Packung01".

init = String

Dieses Feld bestimmt, wie das Simulationsgebiet initial mit Kugeln gefüllt wird. Die möglichen Werte sind

"file": Die Kugeltypen, Mittelpunkte und Radien werden aus einer Datei eingelesen. Format und Dateiname sind im Feld 'input_file' angegeben.

"random": Die Kugeln werden unter Berücksichtigung der Felder 'alpha' und 'r' initialisiert.

"homogen": Die Kugeln werden gleichverteilt unter Berücksichtigung des Feldes 'R' initialisiert.

"gfile": Die Kugeln werden nach den Angaben des Feldes 'gfile' und 'gres' initialisiert. Siehe weiter unten.

L = Gruppe

Gruppe mit Namen 'L1', 'L2', 'L3' und dazugehörigen numerischen Werten für die Seitenlängen des Simulationsgebietes.

lc = Gruppe

Gruppe mit Namen 'lc1', 'lc2', 'lc3' und dazugehörigen numerischen Werten für die Seitenlängen der Zellen.

input_file = Gruppe

Gruppe mit den Feldern 'name' und 'type'. Der Name ist der Dateiname, aus dem die Daten für die initiale Konfiguration der Kugeltypen, Kugelmittelpunkte und Kugelradien gelesen werden. Für das Feld 'type' gibt es folgende Werte:

"xyzr": ASCII-Format mit jeweils einer Zeile für eine Kugel mit den Koordinaten in der Reihenfolge x, y, z und dem Radius.

"bin": Binäres Datenformat entsprechend dem Ausgabeformat mit der Option 'bin'.

"dat": ASCII-Datei mit jeweils einer Zeile pro Kugel mit den Daten:

'Kugelnummer Kugeltyp X Y Z R'.

output_dir = Gruppe

Gruppe mit den Feldern 'name' und 'type'. Der 'name' gibt das Verzeichnis an, in dem die Ausgabedateien hinterlegt werden. Das Feld 'type' bestimmt das Format der Ausgabedateien. Wird es auf 'bin' gesetzt, werden die Kugeldaten in einem Format ausgegeben, das direkt als Eingabeformat für den Dekorationsprozess verwendet werden kann. Wird das Feld auf 'dat' gesetzt, erfolgt eine ASCII-Ausgabe mit jeweils einer Zeile pro Kugel mit den numerischen Daten

'Kugeltyp X Y Z R'.

radii_prob = String

Falls in diesem Feld ein gültiger Dateiname angegeben ist und das Feld 'init' auf 'random' gestellt ist, wird die Verteilungsfunktion wie in 5.1 aus der Datei ausgelesen. In jeder Zeile sind ein gültiger Radius und seine zugehörige Wahrscheinlichkeit angegeben.

N = Integer

Anzahl der initialen Kugeln.

Steps = Integer

Maximale Anzahl der Simulationsschritte.

output_step = Integer

Intervall für die Ausgabe aller Kugelpositionen, Radien und Typen.

dt = Float

Größe der Zeitdiskretisierung.

alpha = Float

Parameter in der Radienverteilungsfunktion ' $\exp(-\alpha \cdot r)$ '. Wird dieser Parameter auf 0 gesetzt, werden die initialen Radien gleichverteilt.

mass_type = String

Wenn dieser String auf "mass" gesetzt ist, gilt für alle Kugeln $m_i = 1$. Ist er nicht auf "mass" gesetzt, gilt initial $m_i = 0$ was den Algorithmus zum Anhalten bringen würde. Daher müssen dann die Massen im Quellcode manuell gesetzt werden. Dies ist eine experimentelle Einstellung.

k = Liste

Liste aus Gruppen. Für jeden Kugeltyp muss es eine Gruppe geben. Jede Gruppe enthält einen Wert für die Felder 'ks' und 'ke', Startwert und Endwert der Wechselwirkungskonstante.

tk = Liste

Liste aus Gruppen. Für jeden Kugeltyp muss es eine Gruppe geben. Jede Gruppe enthält einen Wert für die Felder 'tk's' und 'tke', Startwert und Endwert des Zeitschrittintervalls, innerhalb dessen der entsprechende Wert der Wechselwirkungskonstante sich linear ändert.

nu = Liste

Liste aus Gruppen. Für jeden Kugeltyp muss es eine Gruppe geben. Jede Gruppe enthält einen Wert für die Felder 'nus' und 'nue', Startwert und Endwert der Dissipationskonstanten.

tnu = Gruppe

Liste aus Gruppen. Für jeden Kugeltyp muss es eine Gruppe geben. Jede Gruppe enthält einen Wert für die Felder 'tnus' und 'tnue', Startwert und Endwert des Zeitschrittintervalls, innerhalb dessen der entsprechende Wert der Dissipation sich linear ändert.

cutoff = Float

Abschneideradius für die Wechselwirkungsberechnung.

v = Gruppe

Gruppe mit Namen 'v1', 'v2', 'v3' und dazugehörigen numerischen Werten für die initialen Geschwindigkeiten für alle Kugeln.

phases = Integer

Anzahl der vorkommenden Kugeltypen.

overlap = Vektor

Vektor aus Gleitkommazahlen mit den maximal zulässigen Überlappungen für jeden Kugeltyp.

R = Liste

Liste aus Gruppen. Für jeden Kugeltyp muss es eine Gruppe geben. Jede Gruppe enthält einen Wert für die Felder 'Rmin' und 'Rmax', den minimalen und maximalen Radius. Diese Option ist nur aktiv, wenn die Kugeln initialisiert werden aus einer Radienverteilungsfunktion mit einem 'Rmin' und 'Rmax'.

Beispiel bei zwei Kugeltypen: ({Rmin=0.1; Rmax=1;}, {Rmin=10; Rmax=100;});

rho = [Wert; Wert; ...]

Liste aus Gleitkommazahlen mit den maximal zulässigen Punktdichten für jeden Kugeltyp.

gfile = String

gres = Float

„gfile“ enthält den Dateinamen mit den Typnummern für die Kugeln, die in dieser Phase vorkommen sollen.

„gres“ ist die kubische Seitenlänge eines Phasenbereichs.

Beispiel: Es sei $L_1 = L_2 = L_3 = 10$ und $gres = 5$, dann ist das Gesamtgebiet in acht Phasengebiete geteilt. Sollen zum Beispiel drei Kugeltypen vorkommen, könnte eine „gfile“-Datei wie folgend aussehen:

1 1 2 2 3 3 3 3

delete = Boolean

Wenn 'true', dann werden nicht konvergierte Kugeln am Ende der Simulation entfernt.

deletetype = Vektor

Vektor mit booleschen Werten. Für jeden vorkommenden Kugeltyp ein Wert. Nur relevant, wenn die Option 'delete' angeschaltet ist. Nicht konvergierte Kugeln werden entfernt, wenn der Wert für ihren Typen auf 'true' gesetzt ist.

freeze = Boolean

Allgemeine Option zum Einfrieren bestimmter Kugeltypen.

freezetype = Vektor

Vektor mit booleschen Werten. Für jeden vorkommenden Kugeltyp ein Wert. Eingefroren werden nur Kugeln mit entsprechenden Kugeltypen und wenn die allgemeine Option angeschaltet ist.

radii_chg = Boolean

Allgemeine Option zum Anschalten des Radius-Protokolls.

radii_chg_trigger = Float

Wert für minimale Konvergenzrate. Konvergieren weniger Kugeln innerhalb eines Zeitschrittes, springt der Mechanismus an.

radii_chg_rate = Float

Wert für die Änderungen der Kugelradien.

radiitype = Vektor

Für jeden Kugeltyp, der vom Radiusprotokoll betroffen sein soll, muss dieser Vektoreintrag auf 'true' gesetzt sein.

rnd_walk = Boolean

Schalter für das Verwenden von stochastischen Kräften, falls keine anderen Kräfte auf eine Kugel wirken.

step_size = Float

Maximale Distanz, um die eine Kugel durch die stochastischen Kräfte verschoben werden kann. Der Wert wird in Einheiten des eigenen Radius angegeben.

gww = Boolean

Schalter, um festzulegen, ob die Wechselwirkungen zwischen verschiedenen Kugeltypen generell angeschaltet sind.

locking = bool

Ist dieser Wert auf 'true' gesetzt, versucht der Algorithmus ab dem Zeitschritt, der in „lock_step“ gegeben ist, bereits konvergierte Kugeln nicht mehr zu bewegen, um die Konvergenz zu beschleunigen.

lock_step = Integer

Zeitschritt, ab dem der 'locking'-Mechanismus angeschaltet ist.

Literaturverzeichnis

- [1] Libconfig. URL <http://www.hyperrealm.github.io/libconfig>. online visited: 2019-5-4.
- [2] T. M. Addiscott and A. P. Whitmore. Simulation of solute leaching in soils of differing permeabilities. *Soil Use and Mngt.*, pages 7(2):94–102, 2007.
- [3] S. Afach. Physical and geometric characterization of stochastic interfaces for the case of bentheimer sandstone. *Master Thesis, Institute for Computational Physics, University of Stuttgart*, 2010.
- [4] M.P. Allen and D. J. Tildesley. *Computer Simulations of Liquids*. Oxford University Press, 1987.
- [5] R.M. Baram and P.G. Lind. Deposition of general ellipsoidal particles. *Phys. Rev. E*, 85, 2012.
- [6] A. Barbetta, C. Cametti, G. Rizzitelli, and M. Dentini. Geometrical characterization of polymeric matrices by means of dielectric spectroscopy measurements. *Soft Matter*, 8, pages 1120–1129, 2012.
- [7] P. L. Bhatnagar, E. P. Gross, and M. Krook. Model for collision processes in gases. I. small amplitude processes in charged and neutral one-component systems. *Phys. Rev.*, 94(3):511, 1954.
- [8] H. Biermann and L. Krueger. *Moderne Methoden der Werkstoffprüfung*. Wiley-Vch, 2015.
- [9] B. Biswal, P. E. Øren, R.J. Held, S. Bakke, and R. Hilfer. Stochastic multiscale model for carbonate rocks. *Phys.Rev. E*, 75:061303, 2007.
- [10] B. Biswal, P.-E. Øren, R. Held, S. Bakke, and R. Hilfer. Stochastic multiscale model for carbonate rocks. *Phys. Rev. E*, page 75(6):061303, 2007.
- [11] M. Bouzidi, M. Firdaouss, and P. Lallemand. Momentum transfer of a Boltzmann-lattice fluid with boundaries. *Phys. Fluids*, pages 13(11):3452–3459, 2001.

-
- [12] S. Chapman and T. G. Cowling. *The mathematical theory of non-uniform gases*. Cambridge University Press, second edition, 1952.
- [13] S. Chen, H. Chen, D. Martínez, and W. H. Matthaeus. Lattice Boltzmann model for simulation of magnetohydrodynamics. *Phys. Rev. Lett.*, 67(27):3776, 1991.
- [14] R. E. Collins. *Flow of fluids through porous materials*. Reinhold chemical engineering series. Reinhold Publishing Company, 1961.
- [15] P. Cosenza, D. Pret, A. Giraud, and S. Hedan. Effect of the clay distribution on the effective elastic properties of shales. *Mechanics of Materials*, 84, pages 55–74, 2015.
- [16] G. W. Delaney, T. Di Matteo, and T. Aste. Combining tomographic imaging and dem simulation to investigate the structure of experimental sphere packings. *J. Soft Matter*, 6: 2992–3006, 2010.
- [17] D. d’Humières and P. Lallemand. Numerical simulations of hydrodynamics with lattice gas automata in two dimensions. *Complex Systems*, 1:599, 1987.
- [18] D. d’Humières, I. Ginzburg, M. Krafczyk, P. Lallemand, and L.-S. Luo. Multiple-relaxation-time lattice Boltzmann models in three dimensions. *Phil. Trans. R. Soc. Lond. A*, pages 360(1792):437–451, 2002.
- [19] K. L. Dixon and R. L. Nichol. Permeability estimation from transient vadose zone pumping tests in shallow coastal-plain sediments. *Env. Geosci.*, pages 12(4):279–289, 2005.
- [20] E. Donaldson, R. Thomas, and P. Lorenz. Wettability determination and its effect on recovery efficiency. *Soc. Pet. Eng., Vol. 9, p. 13*, 1969.
- [21] W. England, A. Mackenzie, D. Mann, and T. Quigley. The movement and entrapment of petroleum fluids in the subsurface. *J. Geol. Soc., Vol. 144, p. 327*, page p. 327, 1987.
- [22] E. F. Evati. *High performance simulation of fluid flow in porous media using lattice Boltzmann method*. Dissertation, Technische Universität München, 2017.
- [23] Z. Fishman and A. Bazlak. Heterogeneous through-plane porosity distributions for treated pemfc gds. *Journal of The Electrochemical Society*, 158 (8), pages 841–845, 2011.
- [24] Message Passing Interface Forum. *MPI: A Message-Passing Interface Standard, Version 2.1*. 2008.

- [25] P. Garrido, F. Concha, and R. Bürger. Settling velocities of particulate systems: 14. unified model of sedimentation, centrifugation and filtration of flocculated suspensions. *Int. J. Mineral Process*, pages 72:57–74, 2003.
- [26] I. Ginzbourg and D. d'Humières. Local second-order boundary methods for lattice Boltzmann models. *J. Stat. Phys.*, 84(5/6):927, 1996.
- [27] I. Ginzburg and D. d'Humières. Multireflection boundary conditions for lattice Boltzmann models. *Phys. Rev. E*, page 68(6):066614, 2003.
- [28] I. Ginzburg, F. Verhaeghe, and D. d'Humières. Study of simple hydrodynamic solutions with the two-relaxation-times lattice Boltzmann scheme. *Comm. Comp. Phys.*, pages 3(3):519–581, 2008..
- [29] I. Ginzburg, F. Verhaeghe, and D. d'Humières. Two-relaxation-time lattice Boltzmann scheme: about parametrization, velocity, pressure and mixed boundary conditions. *Comm. Comp. Phys.*, pages 3(2):427–478, 2008..
- [30] R.C. Gonzalez and R.E. Woods. *Digital Image Processing (3rd Edition)*. 2008.
- [31] S. Grottel, G. Reina, T. Zauner, R. Hilfer, and T. Ertl. Particle-based rendering for porous media. *Proceedings of SIGRAD 2010: Content aggregation and visualization*. Linköping University Electronic Press, 2010.
- [32] Z. Guo, C. Zheng, and B. Shi. Discrete lattice effects on the forcing term in the lattice Boltzmann method. *Phys. Rev. E*, page 65(4):046308, 2002.
- [33] D. Hänel. *Molekulare Gasdynamik: Einführung in die kinetische Theorie der Gase und Lattice-Boltzmann-methoden*. Springer, page 2004.
- [34] M. Hasert and S. Roller. Lattice boltzmann simulation of non-darcy flow in porous media. *Procedia Computer Science 00*, 1-10, 2015.
- [35] X. He, Q. Zou, L. S. Luo, and M. Dembo. Analytic solutions of simple flows and analysis of non-slip boundary conditions for the lattice Boltzmann BGK model. *J. Stat. Phys.*, 87:115, 1997.
- [36] R. Hilfer. Transport and relaxation phenomena in porous media. *Adv. Chem. Phys.*, XCII:299, 1996.

-
- [37] R. Hilfer. Local porosity theory and stochastic reconstruction for porous media. In K. Mecke and D. Stoyan, editors, *Statistical Physics and Spatial Statistics*, volume 554 of *Lecture Notes in Physics*, page 203, Berlin, 2000. Springer.
- [38] R. Hilfer. Multiscale local porosity theory, weak limits, and dielectric response in composite and porous media. *Journal of Mathematical Physics*, 59, page 103511, 2018.
- [39] R. Hilfer and A. Lemmer. Differential porosimetry and permeability for random porous media. *Physical Review E*, 92, page 013305, 2015.
- [40] R. Hilfer and T. Zauner. High-precision synthetic computed tomography of reconstructed porous media. *Phys. Rev. E*, 84, pages 062301–1:5, 2011.
- [41] J. Hyväluoma and Jens Harting. Slip flow over structured surfaces with entrapped microbubbles. *Physical Review Letters*, 100:246001, 2008. arXiv:0801.1448.
- [42] A. Jacob, F. Enzmann, C. Hinz, and M. Kersten. Analysis of variance of porosity and heterogeneity of permeability at the pore scale. *Transport in Porous Media* DOI:10.1007/s11242-019-01342-7, 2019.
- [43] Y. Jobic, P. Kumar, F. Topin, and R. Occelli. Determining permeability tensors of porous media A novel vector kinetic numerical approach. *International Journal of Multiphase Flow*, 110, 2019.
- [44] M. Kataja, K. Hiltunen, and J. Timonen. Flow of water and air in a compressible porous medium. a model of wet pressing of paper. *Journal of Physics D: Applied Physics*, Vol. 25, No. 7, page p. 1053, 1992.
- [45] L.M. Keller, L. Holzer, P. Schuetz, and P. Gasser. Pore space relevant for gas permeability in opalinus clay: Statistical analysis of homogeneity, percolation, and representative volume element. *Journal of geophysical research: Solid Earth*, Vol. 118, pages 2799–2812, 2013.
- [46] Wan KeShu and Xu Qiong. Local porosity distribution of cement paste characterized by x-ray micro-tomography. *Science China*, Vol. 57, No. 5, pages p953–p961, 2014.
- [47] S. Knapek, G. Zumbusch, and A. Caglar. Numerische Simulation in der Molekulardynamik. Springer, 2004.
- [48] A. J. C. Ladd and R. Verberg. Lattice-boltzmann simulations of particle-fluid suspensions. *J. Stat. Phys.*, 104(5):1191, 2001.
- [49] P. Lallemand and L.-S. Luo. Theory of the lattice Boltzmann method: dispersion, dissipation, isotropy, Galilean invariance, and stability. *Phys. Rev. E*, pages 61(6):6546–6562, 2000.

- [50] P. Lallemand and L.-S. Luo. Theory of the lattice Boltzmann method: acoustic and thermal properties in two and three dimensions. *Phys. Rev. E*, page 68(3):036706, 2003..
- [51] P. Lallemand and L.-S. Luo. Lattice Boltzmann method for moving boundaries. *J. Comp. Phys.*, pages 184(2):406–421, 2003..
- [52] C. Lang, J. Ohser, and R. Hilfer. On the analysis of spatial binary images. *Journal of Microscopy*, 203:303, 2001.
- [53] L. Latief, B. Biswal, U. Fauzi, and R. Hilfer. Continuum reconstruction of the pore scale microstructure for fontainbleau sandstone. *Physica A*, page 389 1607, 2010.
- [54] A. Lemmer and R. Hilfer. Parallel domain decomposition method with non-blocking communication for flow through porous media. *Journal of Computational Physics*, 281, pages 970–981, 2015.
- [55] J. Li, D. Chen, H. Luan, and Y. Zhang and Y. Fan. Numerical evaluation and prediction of porous implant design and flow performance. *Biomed Res Int.* 1215021., 2018.
- [56] J. Liu, G.C. Pereira, and K. Regenauer-Leib. From characterisation of pore-structures to simulations of pore-scale fluid flow and the upscaling of permeability using microtomography: A case study of heterogeneous carbonates. *International Journal of Rock Mechanics Mining Sciences*, 70, pages 82–89, 2014.
- [57] C. Manwart, U. Aaltosalmi, A. Koponen, R. Hilfer, and J. Timonen. Lattice-boltzmann and finite-difference simulations for the permeability of three-dimensional porous media. *Phys. Rev. E*, 66:016702, 2002.
- [58] Christian Manwart. *Geometrische Modellierung und Transporteigenschaften poröser Medien*. PhD thesis, Fakultät Physik der Universität Stuttgart, Logos Verlag, Berlin, 2001.
- [59] Christian Manwart, U. Aaltosalmi, A. Koponen, R. Hilfer, and J. Timonen. Lattice-boltzmann and finite-difference simulations for the permeability of three-dimensional porous media. *Physical Review E*, 66:016702, 2002.
- [60] A. Narváez, Th. Zauner, F. Raischel, R. Hilfer, and J. Harting. Quantitative analysis of numerical estimates for the permeability of porous media from lattice-boltzmann simulations. *Journal of Statistical Mechanics*, pages 11026–11040, 2010.
- [61] J. Ohser and F. Mücklich. Statistical analysis of microstructures and materials science. *John Wiley & Sons*, pages 127–138, 2000.

-
- [62] N. Otsu. A threshold selection method from gray-level histograms. *IEEE Trans. Syst. Man. Cybernet*, pages 9,62, 1997.
- [63] C. Pan, L.-S. Luo, and C. T. Miller. An evaluation of lattice Boltzmann schemes for porous medium flow simulation. *Computers & Fluids*, pages 35:898–909, 2006.
- [64] J. Petrasch, F. Meier, H. Friess, and A. Steinfeld. Tomography based determination of permeability, dupuit-forchheimer coefficient, and interfacial heat transfer coefficient in reticulate porous ceramics. *International Journal of Heat and Fluid Flow*, Vol. 29, pages 315–326, 2008.
- [65] Y. Qian. Fractional propagation and the elimination of staggered invariants in lattice-BGK models. 8(4):753, 1997.
- [66] Y. H. Qian, D. d’Humières, and P. Lallemand. Lattice BGK models for Navier-Stokes equation. 17(6):479, 1992.
- [67] A. Rabbani, S. Ayatollahi, R. Kharrat, and N. Dashti. Estimation of 3-d pore network coordination number of rocks from watershed segmentation of a single 2-d image. *Advances in Water Resources*, 94, 2016.
- [68] A. Rabbani, P. Mostaghimi, and R. T. Armstrong. Pore network extraction using geometrical domain decomposition. *Advances in Water Resources*, Volume 123, Pages 70-83, 2019.
- [69] C. Recarey, I. Perez, and R. Rosello et al. Advances in particle packing algorithms for generating the medium in the discrete element method. *Computer Methods in Applied Mechanics and Engineering*, <https://doi.org/10.1016/j.cma.2018.11.011>, 2018.
- [70] M. N. Roozbahani, R. Borela, and J. D. Frost. Pore size distribution in granular material microstructure. *Materials*, 10, 1237, 2017.
- [71] S.M. Rubio-Largo, P.G. Lind, D. Maza, and R.C. Hidalgo. Granular gas of ellipsoids: analytical collision detection implemented on gpus. *Comp. Part. Mech*, 6:127–138, 2015.
- [72] P.K. Sahoo and S.S. Wong. A survey of thresholding techniques. *Computer Vision, Graphics and Image Processing* 41, pages 233–260, 1988.
- [73] S. Succi. *The lattice Boltzmann equation for fluid dynamics and beyond*. 2001.
- [74] S. Succi, M. Vergassola, and R. Benzi. Lattice Boltzmann scheme for two-dimensional magnetohydrodynamics. *Phys. Rev. A*, 43(8):4521, 1991.
- [75] L. Verlet. Computer ‘experiments’ on classical fluids. i. thermodynamical properties of lennard jones molecules. *Phys. Rev.* 159, pages 98–103, 1967.

- [76] K. Wieghardt. *Theoretische Strömungslehre. Göttinger Klassik*, page 1957.
- [77] D. A. Wolf-Gladrow. *Lattice-Gas Cellular Automata and lattice Boltzmann models. Springer*, page 2005.
- [78] Peijie Yin and Gao-Feng Zhao. Stochastic reconstruction of gosford sandstone from surface image. *International Journal of Rock Mechanics Mining Sciences*, 70, pages 82–89, 2014.
- [79] M. Zambrano, E. Tondi, L. Mancini, G. Lanzafame and F.X. Trias, F. Arzilli, M. Materazzi, and S. Torrieri. Fluid flow simulation and permeability computation in deformed porous carbonate grainstones. *Advances in Water Resources*, 115, 2018. doi: 10.1016/j.advwatres.2018.02.016.
- [80] T. Zauner. Application of a force field algorithm for creating strongly correlated multiscale sphere packings. *Journal of Computational Physics* 313, pages 662–673, 2016.
- [81] Q. Zou and X. He. On pressure and velocity boundary conditions for the lattice Boltzmann BGK model. *Phys. Fluids*, pages 9(6):1591–1598, 1997.

Danksagung

An erster Stelle danke ich Herrn Professor Dr. Dr. Rudolf Hilfer für seine langjährige Unterstützung und seine Anleitung zum exakten wissenschaftlichen Arbeiten. Für konstruktive Anregungen danke ich ebenso Prof. Dr. Hans-Rainer Trebin, der meine Doktorarbeit als zweiter Gutachter betreut hat.

Meinen ehemaligen Kollegen an der Universität Stuttgart gilt mein Dank für die gute Zusammenarbeit und die vielen anregenden Diskussionen, dabei möchte ich insbesondere Florian Doster, Oliver Hönig, Ariel Narváez und Jens Harting erwähnen.

Für das große Vertrauen und die stete Ermutigung, dass ich meine Ziele erreichen kann, möchte ich meinem Vater besonderen Dank aussprechen.

Der größte Dank gebührt meiner Frau Yasemin. Es gab viele Momente des Zweifels bei denen Sie die entscheidende Stütze war und es gab viele Aufgaben, die Sie übernommen hat, damit ich diese Arbeit fertigstellen konnte.

Eidesstattliche Versicherung über die eigenständig erbrachte Leistung gemäß §6 Absatz 2 der Promotionsordnung der Universität Stuttgart

Die eingereichte Dissertation zum Thema

Hocheffiziente Präzisionsalgorithmen zur Modellierung und Analyse granularer poröser Medien

stellt meine eigenständig erbrachte Leistung dar.

Ich habe ausschließlich die angegebenen Quellen und Hilfsmittel benutzt. Wörtlich oder inhaltlich aus anderen Werken übernommene Angaben habe ich als solche kenntlich gemacht.

Die Bedeutung der Eidesstattlichen Versicherung ist mir bekannt; ebenso die eventuellen strafrechtlichen Folgen, welche aus einer unrichtigen oder unvollständigen eidesstattlichen Versicherung resultieren können (s. Anlage: „Belehrung über die Bedeutung der Eidesstattlichen Versicherung“).

Die Richtigkeit der hier getätigten Angaben bestätige ich und versichere an Eides Statt, nach bestem Wissen die Wahrheit erklärt zu haben.

Großhesselohe, den 12.10.2020

A handwritten signature in blue ink, appearing to read 'T. Zimmer'.

Belehrung über die Bedeutung und die strafrechtlichen Folgen der Eidesstattlichen Versicherung gemäß §6 Absatz 2 der Promotionsordnung der Universität Stuttgart

Der Fachbereich Physik der Universität Stuttgart empfiehlt die Einbindung einer Eidesstattlichen Versicherung in die Dissertation zur Dokumentation der Eigenständigkeit der erbrachten wissenschaftlichen Leistung sowie zum schriftlichen Nachweis, dass die Promovierende/der Promovierende die vorliegende wissenschaftliche Leistung ohne unzulässige Hilfsmittel erbracht hat.

Der Gesetzgeber misst der Eidesstattlichen Versicherung eine hohe Bedeutung bei und hat dementsprechend die vorsätzlich falsche oder fahrlässig falsche Abgabe einer eidesstattlichen Versicherung unter Strafe gestellt. Näheres hierzu regeln die §§156 und 161 des Strafgesetzbuches.

§156 StGB: Falsche Versicherung an Eides Statt

Wer vor einer zur Abnahme einer Versicherung an Eides Statt zuständigen Behörde eine solche Versicherung falsch abgibt oder unter Berufung auf eine solche Versicherung falsch aussagt, wird mit Freiheitsstrafe bis zu drei Jahren oder mit Geldstrafe bestraft.

§161 StGB: Fahrlässiger Falscheid, fahrlässige falsche Versicherung an Eides Statt

(1) Wenn eine der in den §§154 bis 156 bezeichneten Handlungen aus Fahrlässigkeit begangen worden ist, so tritt Freiheitsstrafe bis zu einem Jahr oder Geldstrafe ein.

(2) Straflosigkeit tritt ein, wenn der Täter die falsche Angabe rechtzeitig berichtigt.

Die Vorschriften des §158 Abs. 2 und 3 gelten entsprechend.

Ich habe die Belehrung zur Eidesstattlichen Versicherung zur Kenntnis genommen.

Großhesselohe, den 12.10.2020

A handwritten signature in blue ink, appearing to read 'T. Zimmer'.

Erklärung zur Übereinstimmung der digitalen Version mit der vorgelegten Printversion der Dissertation

Hiermit erkläre ich,

Zauner Thomas, geboren am 08.11.1977 in München,

dass das von mir eingereichte pdf-Dokument zur Dissertation mit dem Thema

**„Hocheffiziente Präzisionsalgorithmen zur Modellierung und Analyse
granularer poröser Medien“**

in Inhalt und Wortlaut der ebenfalls eingereichten Printversion meiner Dissertationsschrift entspricht.

Großhesselohe, den 12.10.2020

