# Numerical Modeling of Fluid Flow in Porous Media and in Driven Colloidal Suspensions

Jens Harting, Thomas Zauner, Rudolf Weeber, and Rudolf Hilfer

Institut für Computerphysik, Pfaffenwaldring 27, 70569 Stuttgart, Germany

**Summary.** This article summarizes some of our main efforts performed on the computing facilities provided by the high performance computing centers in Stuttgart and Karlsruhe. At first, large scale lattice Boltzmann simulations are utilized to support resolution dependent analysis of geometrical and transport properties of a porous sandstone model. The second part of this report focuses on Brownian dynamics simulations of optical tweezer experiments where a large colloidal particle is dragged through a polymer solution and a colloidal crystal. The aim of these simulations is to improve our understanding of structuring effects, jamming behavior and defect formation in such colloidal systems.

## 1 Resolution Dependent Analysis of Geometrical and Transport Properties of a Porous Sandstone Model

Geometrical characterization of porous media and the calculation of transport parameters present an ongoing challenge in many scientific areas such as petroleum physics, environmental physics (aquifers), biophysics (membranes) and material science. We perform large scale lattice-Boltzmann simulations to investigate the permeability of computer generated samples of quartzitic sandstone at different resolutions. To obtain these laboratory scale samples a continuum model is discretized at different resolutions. This allows to obtain high precision estimates of the permeability and other quantities by extrapolating the resolution dependent results.

### 1.1 Simulation Method and Implementation

The lattice-Boltzmann (hereafter LB) simulation technique is based on the well-established connection between the dynamics of a dilute gas and the Navier-Stokes equations [3]. We consider the time evolution of the one-particle velocity distribution function $n(\mathbf{r}, \mathbf{v}, t)$, which defines the density of particles

with velocity $\mathbf{v}$ around the space-time point $(\mathbf{r}, t)$. By introducing the assumption of molecular chaos, i.e. that successive binary collisions in a dilute gas are uncorrelated, Boltzmann was able to derive the integro-differential equation for $n$ named after him [3]

$$\partial_t n + \mathbf{v} \cdot \nabla n = \left( \frac{dn}{dt} \right)_{coll} , \tag{1}$$

where the right hand side describes the change in $n$ due to collisions.

The LB technique arose from the realization that only a small set of discrete velocities is necessary to simulate the Navier-Stokes equations [4]. Much of the kinetic theory of dilute gases can be rewritten in a discretized version. The time evolution of the distribution functions $n$ is described by a discrete analogue of the Boltzmann equation [10]:

$$n_i(\mathbf{r} + \mathbf{c}_i \Delta t, t + \Delta t) = n_i(\mathbf{r}, t) + \Delta_i(\mathbf{r}, t) , \tag{2}$$

where $\Delta_i$ is a multi-particle collision term. Here, $n_i(\mathbf{r}, t)$ gives the density of particles with velocity $\mathbf{c}_i$ at $(\mathbf{r}, t)$. In our simulations, we use 19 different discrete velocities $\mathbf{c}_i$. The hydrodynamic fields, mass density $\varrho$ and momentum density $\mathbf{j} = \varrho \mathbf{u}$ are moments of this velocity distribution:

$$\varrho = \sum_i n_i , \qquad \mathbf{j} = \varrho \mathbf{u} = \sum_i n_i \mathbf{c}_i . \tag{3}$$

We use a linear collision operator,

$$\Delta_i = -\frac{1}{\tau}(n_i - n_i^{eq}) , \tag{4}$$

where we assume that the local particle distribution relaxes to an equilibrium state $n_i^{eq}$ at a single rate $\tau$ [1]. By employing the Chapman-Enskog expansion [3, 4] it can be shown that the equilibrium distribution

$$n_i^{eq} = \varrho \omega^{c_i} \left[ 1 + 3\mathbf{c}_i \cdot \mathbf{u} + \frac{9}{2}(\mathbf{c}_i \cdot \mathbf{u})^2 - \frac{3}{2}u^2 \right] , \tag{5}$$

with the coefficients $\omega^{c_i}$ corresponding to the three different absolute values $c_i = |\mathbf{c}_i|$,

$$\omega^0 = \frac{1}{3} , \qquad \omega^1 = \frac{1}{18} , \qquad \omega^{\sqrt{2}} = \frac{1}{36} , \tag{6}$$

and the kinematic viscosity

$$\nu = \frac{\eta}{\varrho_f} = \frac{2\tau - 1}{6} , \tag{7}$$

properly recovers the Navier-Stokes equations

$$\frac{\partial u}{\partial t} + (u\nabla)u = -\frac{1}{\varrho}\nabla p + \frac{\eta}{\varrho}\Delta u \,, \qquad \nabla u = 0 \,. \tag{8}$$

We use LB3D [6], a highly scalable parallel LB code, to implement the model. LB3D is written in Fortran 90 and designed to run on distributed-memory parallel computers, using MPI for communication. It can handle up to three different fluid species and is able to model flow in complex geometries as it occurs for example in porous media. In each simulation, the fluid is discretized onto a cubic lattice, each lattice point containing information about the fluid in the corresponding region of space. Each lattice site requires about a kilobyte of memory per lattice site so that, for example, a simulation on a $128^3$ lattice would require around 2.2GB of memory. The code runs at over $6 \cdot 10^5$ lattice site updates per second per CPU on a recent machine, and has been observed to have roughly linear scaling up to order $3 \cdot 10^3$ compute nodes (see below).

Simulations on larger scales have not been possible so far due to the lack of access to a machine with a higher processor count. The largest simulation we performed used a $1536^3$ lattice and ran on the AMD Opteron based cluster in Karlsruhe. There, it is not possible to use a larger lattice since the amount of memory per CPU is limited to 4GB and only 1024 processes are allowed within a single job. On the NEC SX-8 in Stuttgart, typical system sizes are of the order of $256 \times 256 \times 512$ lattice sites. The output from a simulation usually takes the form of a single floating-point number for each lattice site, representing, for example, the density of a fluid at that site. Therefore, a density field snapshot from a $128^3$ system would produce output files of around 8MB.

Writing data to disk is one of the bottlenecks in large scale simulations. If one simulates a $1024^3$ system, each data file is 4GB in size. The situation gets even more critical when it comes to the files needed to restart a simulation. Then, the state of the full simulation lattice has to be written to disk requiring 0.5TB of disk space. LB3D is able to benefit from the parallel file systems available on many large machines today, by using the MPI-IO based parallel HDF5 data format [7]. Our code is very robust regarding different platforms or cluster interconnects: even with moderate inter-node bandwidths it achieves almost linear scaling for large processor counts with the only limitation being the available memory per node. The platforms on which our code has been successfully used include various supercomputers like the NEC SX-8, IBM pSeries, SGI Altix and Origin, Cray T3E, Compaq Alpha clusters, as well as low cost 32- and 64-bit Linux clusters.

During the last year, a substantial effort has been invested to improve the performance of LB3D and to optimize it for the simulation of flow in porous media. Already during the previous reporting period, we improved the performance on the SX8 in Stuttgart substantially by rearranging parts of the code and by trying to increase the length of the loops. These changes were proposed by the HLRS support staff. However, while the code scales very well with the number of processors used, the single CPU performance is still below

what one could expect from a lattice Boltzmann implementation on a vector machine. The vector operation ratio is about 93%, but due to the inherent structure of our multiphase implementation, the average loop length is only between 20 and 30. Thus, the performance of our code stays below 1GFlop/s. For this reason, we are currently performing most of our simulations on the Opteron cluster *XC2* in Karlsruhe. Our code performs extremely well there and shows almost linear scaling to up to 1024 CPU's. Further, due to extensive improvements of the code during the last year, we were able to increase the per CPU performance by a factor of about two.

The lattice Boltzmann code can read voxel based 3D representations of porous media. Such data can either be obtained from XMT measurements of real samples or from numerical models. In order to compute the permeability of such a sample, the velocity field $v(x)$ and pressure field $p(x)$ as created by the LB simulations are required. For a liquid with dynamic viscosity $\eta$, the permeability $\kappa$ is defined according to Darcy's law

$$\langle v(x)\rangle_{x\in S} = -\frac{\kappa}{\eta}\langle \nabla p\rangle_{x\in S} \tag{9}$$

and thus

$$\kappa = -\eta\frac{\langle v(x)\rangle_{x\in S}}{\langle \nabla p\rangle_{x\in S}}, \tag{10}$$

with $\langle v(x)\rangle_{x\in S}$ being the velocity component in flow direction averaged over the full pore space $S$. We approximate the average pressure gradient $\langle \nabla p\rangle_{x\in S}$ of the full sample like

$$\langle \nabla p\rangle_{x\in S} \approx \frac{\langle p(x)\rangle_{x\in OUT} - \langle p(x)\rangle_{x\in IN}}{aL}, \tag{11}$$

with IN/OUT representing the plane, perpendicular to the flow direction, in front of / behind the porous medium, the total length $L$ (in voxel) and $a$ the resolution. The accelerating force, in positive z-direction, is applied as a body force only in the first quarter of the "IN-flow" buffer before the sample, another buffed "OUT-flow" was added after the sample. Periodic boundary conditions in flow direction are being imposed. Fig. 1 shows the simulation setup for a sample with resolution $a = 10\mu m$ and voxel dimension 136x136x168 and the two "IN/OUT-flow" buffers.

### 1.2 Sample Creation and Simulation Setup

Because appropriately sized digital samples at sufficient resolutions are not available from experimental data, a continuum model of a quartzitic sandstone was discretized at different resolutions and sizes (Table 1) and then thresholded to generate digital voxel data. For details explaining the model see [2]. With increasing resolution $a$ the microstructure becomes more and more resolved at the expense of increasing the amount of data and CPU time

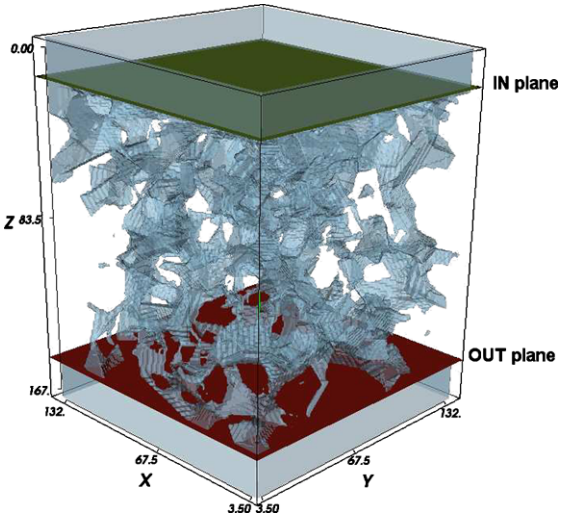


**Fig. 1.** Porous medium with pore space (blue) and the two IN/OUT-flow buffers shown. The flow is in positive $z$-direction, the sample size is 136x136x168 voxel, at a resolution of $a = 10\mu m$

in simulations. To find a good compromise between a high enough resolution and manageable systems for LB-simulations, geometrical characterizations for each of our samples were calculated (See Fig. 3,4).

**Table 1.** List of available digitized samples

| Sidelength [voxel] | Resolution $a[\mu m]$ | Number of samples |
|---|---|---|
| 256 | 80 | 1 |
| 256 | 40 | 1 |
| 512/256 | 20 | 1/8 |
| 512/256 | 10 | 8/16 |
| 512/256 | 5 | 8/16 |
| 512/256 | 2.5 | 8/16 |

Typical lattice-Boltzmann simulation required 50,000–80,000 simulation steps to reach stationary flow within the pore space. Simulations were performed on 64 to 512 CPU's. For the total of 83 samples (including calibration test runs) more than 80,000 CPU hours were required. The LB3D code scales linearly and memory usage was moderate, making investigations of larger samples feasible and tempting. Fig. 2 shows the average time for one simulation step and per voxel in nano seconds (left axis). The average time was calculated from more than 100 runs on different systems with varying parameters and

sizes. The speedup factor is defined as

$$speedup(n) = \frac{(total\ runtime\ on\ 4\ CPU's)}{(total\ runtime\ on\ n\ CPU's)}$$

The total runtime of a simulation is the time the program needs for it's full execution. The speedup factor was defined with reference to a run on 4 CPU's because on 4-core systems one physical node has 4 CPU's. The communication and I/O overhead for a simulation run on one physical node is negligible, compared to network communications.

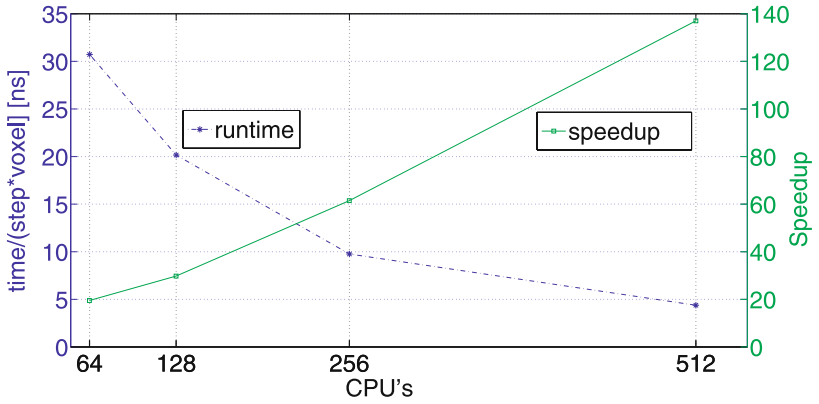


**Fig. 2.** LB3D code scaling behavior. The sequential lattice Boltzmann code is known to scale linearly with the sample size (voxel). The time per simulation step and per voxel is shown on the left axis. The speedup, shown on the right axis

### 1.3 Geometric Properties at Different Resolutions

For each sample geometrical properties such as the porosity, specific surface and mean/total curvature have been calculated to investigate their behavior with changing resolution and subsampling. From the resolution dependent porosity $\phi(a)$ (Fig. 3) conclusions can be drawn which resolution ranges approximate the true physical porosity well enough and thus, are suitable for simulations. Increasing the resolution (lower $a$) beyond a certain point (here approx. $a = 5 - 10\mu m$) will not justify the increase in data and CPU time but on the other hand low resolutions ($a = 50 - 80\mu m$) will most likely not yield relevant simulation data, because not even the sample porosity is close to the true porosity of the physical sandstone. As can be seen in (Fig. 3), subsamples much smaller than $256\mu m$ will not represent the full sample well. The data

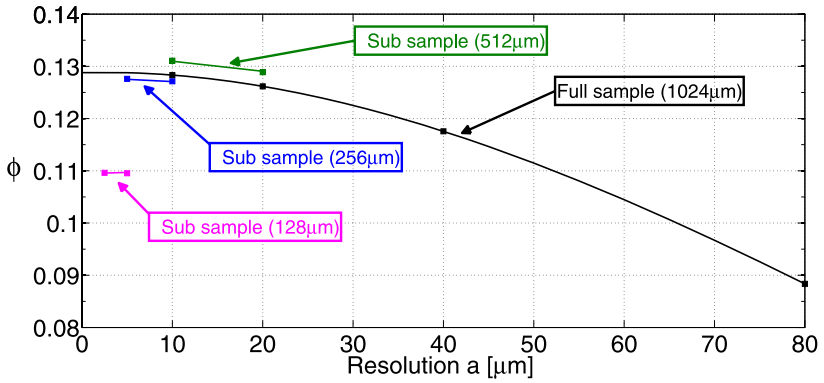


**Fig. 3.** Porosity $\phi$ for the full sample and subsamples at different resolutions a. The data for the full sample is extrapolated to the true physical porosity at a=0. With increasing resolution the porosity approaches the true physical porosity. Large enough subsamples, $512\mu m$ and $256\mu m$, approximate the full sample (size $1024\mu m$) porosity well. Subsamples at $128\mu m$ are not representative for the full sample

for the full sample (black line) can be extrapolated to approximate the true physical porosity as $a \to 0$.

Local porosity distributions [8, 9] are defined as

$$\mu(\phi, L, a) = \frac{1}{m} \sum_{x \in S} \delta(\phi - \phi(x, L, a))$$

where $\phi(x, L, a)$ is the local porosity of a measurement cell with sidelength $L$ at the position $x$ and a sample resolutions $a$. $S$ is the full sample, $m$ is the number of evaluated measurement cells and $\delta$ is the Dirac delta function.

To calculate this local porosity distribution a small cubic measurement cell, with sidelength $L$, is being moved through the whole sample and at all positions $x$, the local porosity $\phi(x, L, a)$ within the cell is calculated.

The local porosity distributions $\mu(\phi, L, a)$, as shown in Fig. 4, give the probability density that a randomly placed measurement cell with sidelength $L$ has a porosity $\phi$ when the sample resolution is $a$. In our case they have a maximum close to the porosity of the full sample at that resolution. For resolutions $a = 40, 20, 10\mu m$ the distributions are well converged. Together with other geometric properties defined within Local porosity theory $\mu(\phi, L, a)$ can be used to estimate the permeability [8].

The lattice Boltzmann simulations calculated the velocity field and pressure field for all available samples with different resolutions. In Fig. 5 the z-Component $v_z$, component in flow direction, of the velocity field for a sample with resolution $a = 10\mu m$ and voxel size 136x136x168 is shown. The pore structure of the same sample is shown in Fig. 1. The "IN-flow" buffer, where the liquid is accelerated to a speed of approx 0.002 lattice units, is shown in

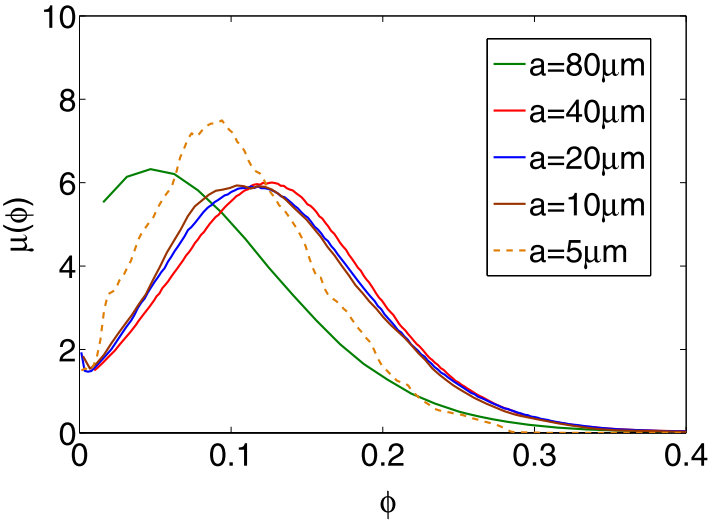


**Fig. 4.** Local porosity distributions for $a$ measurement cell size $L = 320\mu m$ at different resolutions a. Very high resolutions ($a = 5\mu m$) yield bad statistics, at the size of the measurement cell used here. The distributions for resolutions $a = 40, 20, 10\mu m$ are very similar in shape, the mean porosity changes according to Fig. 3

green at the top. All voxel with $v_z > 0$ are shown in translucent blue. The brown isosurfaces depict areas where $v_z > 0.001$ in lattice units, being half of the maximum speed in the "IN-flow" buffer. They represent channels where the liquid flows fast. The flow through the porous medium is not homogeneous, even if the porous medium is quite homogeneous at this resolution.

In addition to the geometric characterizations and the velocity fields high precision permeability calculations (Tbl. 2) for all 83 samples have been performed and are now been critically analyzed for accuracy, to gain further insight into their resolution dependence and thereby into the nature of fluid transport within highly complex geometries.

**Table 2.** Selected permeability results of the full sample ($1024\mu m$) for different resolutions. An relative error of 0.05 has been estimated, resulting from the inaccuracy of the velocity field, calculated by the LB-simulation

| Resolution [$\mu m$] | Permeability [$\mu m^2$] |
|---|---|
| 40 | 1.7±0.043 |
| 20 | 2.2±0.055 |
| 10 | 1.9±0.047 |

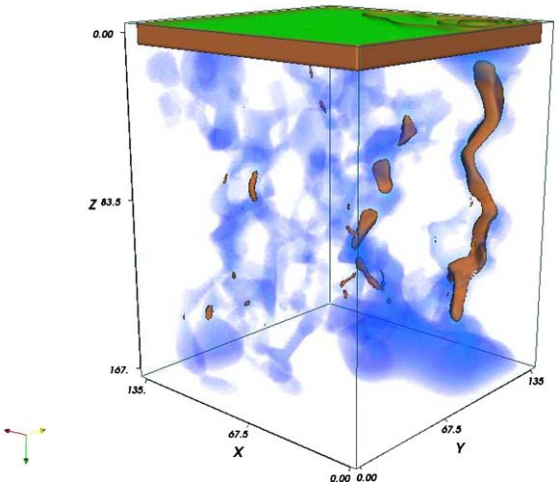


**Fig. 5.** Z-Component of the velocity field. Fluid is accelerated in the IN-flow buffer on top(green). Shown in translucent blue is the complete volume fraction with $v_z > 0$. Darker regions correspond to smaller velocities. All areas with a velocity with $v_z > 0.001$ are shown as isosurfaces. Channels with large connected isosurfaces thus carry the main flow through the porous medium

## 2 Simulation of Optical Tweezer Experiments

A colloidal suspension is a mixture of a fluid and particles or droplets with a length scale of some nanometers to micrometers suspended in it. Colloids are a common part of everyday life. Substances like paint, glue, milk, blood and fog are just some examples. Due to their technical applications, colloids are studied in several disciplines, among them physics, chemistry, and engineering.

Colloidal particles are too large to be affected directly by quantum mechanical effects. On the other hand, they are still small enough to be affected by thermal fluctuations. Therefore, colloidal suspensions are an interesting system to study thermodynamic phenomena like diffusion, phase transitions and more rare phenomena like stochastic resonance and critical Casimir forces. In solid state physics, colloidal crystals are used as model systems to study defect formation, crystal structures and melting. In contrast to many systems in these fields, which are on the nanometer scale, colloidal suspensions can be observed and manipulated directly using techniques like video microscopy, confocal microscopy, total internal reflection microscopy (TIRM) and optical tweezers. This offers numerous possibilities to control these systems on a per particle basis.

We study dynamical properties of colloidal suspensions using computer simulations. The advantage of simulations is that parameters can be controlled in ways that are not accessible in experiments. Also, in many cases, information is available that cannot be measured directly in a real system. Over the past decades, several simulation techniques have become available, which model different aspects of suspensions.

Methods like molecular dynamics and Brownian dynamics only model the dynamics of the suspended colloidal particles and handle the solvent implicitly by adding simplified forces to mimic the solvents behavior. Other approaches like lattice-Boltzmann models, dissipative particle dynamics and stochastic rotation dynamics simulate the complete fluid and couple it to suspended particles. While the first set of methods are computationally very efficient, more complicated hydrodynamic effects are usually not taken into account (polarization of the solvent). The methods that do simulate the full fluid field can reproduce hydrodynamic effects, but achieving quantitative accuracy far from equilibrium is still a challenge. Here, we focus on the Brownian dynamics technique in order to be able to simulate very large systems with an affordable amount of CPU time.

The dynamics of driven suspensions can be examined by dragging a colloidal particle through it. In this article we present our simulations of a particle dragged through a colloidal crystal and a suspension of coiled polymers using an optical tweezer. The focus of the optical tweezer is moved with time, thereby pulling the impurity along.

Optical tweezers trap a colloid (or even an atom) in the focus of a laser beam: this is because a dielectric is always driven along the field gradient. They are a very important tool in soft condensed matter physics: colloids can not only be trapped, they can also be moved around individually by moving the focus of the laser beam. Thereby the colloidal system can be controlled with an accuracy that would be impossible for an atomic system. The optical tweezer is modeled with a harmonic potential, i.e. as if the impurity were connected with the trap center by an ideal spring.

The simulations are motivated by experiments performed by R. Dullens in the group of C. Bechinger in Stuttgart and C. Gutsche in the group of F. Kremer in Leipzig, respectively. In both cases, the simulation parameters are chosen to reproduce the experimental conditions as closely as possible.

### 2.1 Simulation Setup

The experiments are simulated using a modified Brownian dynamics (BD) method which includes some of the hydrodynamics caused by the dragged colloid, as explained in Ref. [11]. The colloids/polymers and the probe particle are modeled as hard spheres with their respective radii.

We use a rectangular simulation volume with periodic boundary conditions in all three directions. Due to long range hydrodynamic interactions, large

systems are required in order to reduce finite size effects. Thus, we typically handle several hundred thousand particles in a single simulation.

The probe particle is trapped in a moving parabolic potential $V(r) = \frac{1}{2}\, a\, r^2$, mimicking the optical tweezer. In the case of the polymer suspension, the potential has a spring constant of $a = 7.5 \times 10^{-5}$ pN/nm, which gives a better signal to noise ratio than the experimental value of $8.5 \times 10^{-2}$ pN/nm. Figure 6 shows a cut out of a snapshot of our simulation setup used to describe the experiments performed by the group in Leipzig.

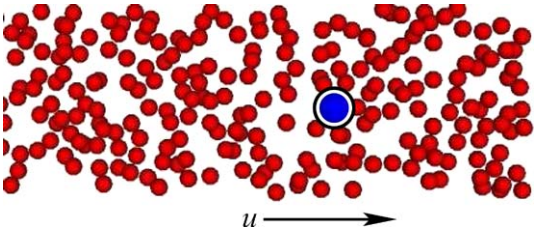


**Fig. 6.** A cut through the simulated system, where a probe particle is dragged through a suspension (from [5]). The arrow indicates the direction of motion of the probe particle

In conventional BD, the two most important aspects of hydrodynamics felt by the suspended particles are taken into account, namely the Stokes friction and the Brownian motion. Correspondingly, this is done by adding to a molecular dynamics simulation two additional forces. The Langevin equation describes the motion a Brownian particle with radius $R$ at position $\mathbf{r}(t)$ as

$$m\,\ddot{\mathbf{r}}(t) = 6\,\pi\,\eta\,R\,\dot{\mathbf{r}}(t) + \mathbf{F}_{\mathrm{rand}}(t) + \mathbf{F}_{\mathrm{ext}}(\mathbf{r}, t), \tag{12}$$

where the first term models the Stokes friction in a solvent of viscosity $\eta$, $\mathbf{F}_{\mathrm{ext}}(\mathbf{r}, t)$ is the sum of all external forces like gravity, forces exerted by other suspended particles, and, for the colloid, the optical trap. $\mathbf{F}_{\mathrm{rand}}(t)$ describes the thermal noise which gives rise to the Brownian motion. The random force on different particles is assumed to be uncorrelated, as well as the force on the same particle at different times. It is further assumed to be Gaussian with zero mean. The mean square deviation of the Gaussian (i.e., the amplitude of the correlator) is given by the fluctuation-dissipation theorem as

$$\langle |\mathbf{F}_{\mathrm{rand}}|^2 \rangle = 12\,\pi\,\eta\,R\,k_B T. \tag{13}$$

This conventional BD scheme is widely used to simulate suspensions because it is well understood, not difficult to implement, and needs much less computational resources than a full simulation of the fluid. However, this simulation method does not resolve hydrodynamic interactions between particles. In particular, the long-ranged hydrodynamic interactions between the dragged

colloid and the surrounding particles are not modeled. However, in the system we consider, these interactions are important, as the dragged colloid moves quickly and has a strong influence on the flow field around it. Therefore, the BD scheme is modified such that the effect caused by the flow field around the dragged colloid is included. This is achieved by calculating the friction force on the small particles with radius $R_c$ not with respect to a resting fluid ($\mathbf{F} = 6\pi\eta R_c \mathbf{u}$), but with respect to the flow field caused by the moving colloid. The friction force then is

$$\mathbf{F} = 6\pi\eta R_c(\mathbf{u} - \mathbf{v}(\mathbf{r})), \tag{14}$$

where $\mathbf{v}(\mathbf{r})$ is the flow field around the moving colloid at a position $\mathbf{r}$ with respect to the colloid's center. This correction leads to the inclusion of two hydrodynamics-mediated effects. Due to the large component of the flow field along the direction of motion, both, in front and behind the probe, small colloids are dragged along. Also the flow of particles is advected around the moving probe particle, i.e., obstacles are moved out of the way to its sides. Both these effects lead to a reduction of drag force on the driven colloid.

### 2.2 Investigating the Stiffness and Occurrence of Defects in Colloidal Crystals

To analyze the effects of the disturbance due to the dragged probe particle, we can either measure the distance that the probe stays behind the focus of the optical tweezer - and thereby the force required to drag the impurity through the crystal - or examine the reaction of the crystal itself. With our simulations we show that in a colloidal crystal, the velocity-force relation for the dragged colloidal particle is close to linear despite the complicated surrounding. It is also shown that the inter-colloid potential does have an influence on the drag force, though not as strong as velocity. This is one example of a result that could not be obtained easily in experiments. Using maps of average particle density and defect distributions, we illustrate the effect of the dragged colloidal particle on the crystal structure.

An example of a rather small system is given in Fig. 7. For large tweezer velocities, much larger crystals are needed and for studying the relaxation of the crystal, we also have to simulate for at least a 100 to 300 seconds of real time causing these simulations to cost up to a few thousand CPU hours each.

### 2.3 Dragging a Colloidal Probe Through a Polymer Suspension

The second system we consider is a suspension of coiled polymers which are modeled as hard spheres. A colloidal particle is dragged through the suspension at high velocities. In the experiment and in the simulation, a drag force is measured that is higher than that calculated from the suspension's viscosity as obtained from a shear rheometer. This increase in drag force can be explained

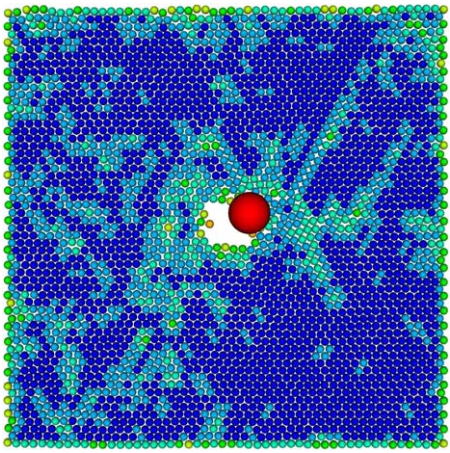

**Fig. 7.** Simulation of a large colloidal particle dragged through a crystal consisting of smaller colloids by means of an optical tweezer. The coloring denotes defects occurring due to the distortion of the crystal

by a jamming of polymers in front of the moving colloidal particle. In contrast to experiments, this jamming can be observed directly in computer simulations as the positions of the polymers are available. The simulation results are compared to dynamic density functional theory calculations by Rauscher et. al and experimental results by Gutsche et al.. A very good quantitative agreement between experiment, theory and simulation is observed [5].

From the simulation data, it is possible to measure the effective polymer concentration around the dragged colloid. To accomplish this, we take about 2000 two dimensional slices of the simulation and move each snapshot such that the position of the colloid coincides in each snapshot. We calculate the probability for each of the $200 \times 200$ bins to be occupied by a polymer by averaging over all snapshots. Polymers accumulate in front of the colloidal particle and the concentration in the back is reduced due to the finite Peclet number of the polymers. For high polymer concentrations, the probability to find a polymer in front of the colloid is close to one. The region right behind the colloid is almost clear of polymers, because the polymers get advected away from the colloid before they can diffuse into this region.

Our model reproduces very well the experimentally found linear relation between drag force and the drag velocity for different polymer concentrations. As higher drag velocities require a larger system (even with periodic boundary conditions) and short numerical time steps, we are limited to about $80\,\mu$m/s by the available computational resources and time. However, the linearity of the drag force with respect to the velocity at high enough velocities allows to extrapolate to the higher velocities used in the experiments.

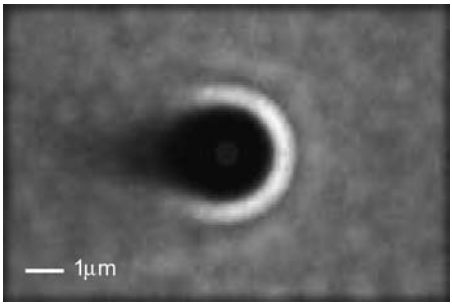


**Fig. 8.** Polymer density around the colloidal particle averaged over 2000 snapshots of the system. Lighter colors correspond to higher polymer densities. Also visible are density oscillations in front of the colloid, which are characteristic for hard sphere systems

## 3 Conclusion

In this report we have presented results from lattice Boltzmann simulations of fluid flow in porous media and the simulation of optical tweezer experiments. In particular the AMD Opteron cluster in Karlsruhe has been found to perform particularly well with our simulation codes. In the case of porous media simulations we have demonstrated that we are able to systematically determine the permeability of digitized quartzitic sandstone samples – even if the resolution of the samples is very high resulting in the need of substantial computational resources.

In the second part of this article we reported on our Brownian dynamics simulations of optical tweezer experiments, where a large probe particle is trapped by a laser beam and dragged either through a colloidal crystal or through a polymer suspension. In both cases, quantitative agreement with experimental data was observed.

### Acknowledgments

We are grateful to the High Performance Computing Center in Stuttgart and the Scientific Supercomputing Center in Karlsruhe for providing access to their NEC SX-8 and HP 4000 machines. We would like to thank Bibhu Biswal, Peter Diez, and Frank Raischel for fruitful discussions. This work was supported by the collaborative research center 716 and the DFG program "nano- and microfluidics".

## References

1. 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.

2. B. Biswal, P.E. Oren, R. Held, S. Bakke, and R. Hilfer. Stochastic multiscale model for carbonate rocks. *Phys.Rev. E*, 75:061303, 2007.
3. S. Chapman and T.G. Cowling. *The mathematical theory of non-uniform gases.* Cambridge University Press, second edition, 1952.
4. U. Frisch, D. d'Humiéres, B. Hasslacher, P. Lallemand, Y. Pomeau, and J.P. Rivet. Lattice gas hydrodynamics in two and three dimensions. *Complex Systems*, 1(4):649, 1987.
5. C. Gutsche, F. Kremer, M. Krüger, M. Rauscher, J. Harting, and R. Weeber. Colloids dragged through a polymer solution: experiment, theory and simulation. *Submitted for publication*, arXiv:0709.4142, 2007.
6. J. Harting, M. Harvey, J. Chin, M. Venturoli, and P.V. Coveney. Large-scale lattice Boltzmann simulations of complex fluids: advances through the advent of computational grids. *Phil. Trans. R. Soc. Lond. A*, 363:1895–1915, 2005.
7. 2003. HDF5 – a general purpose library and file format for storing scientific data, http://hdf.ncsa.uiuc.edu/HDF5.
8. R. Hilfer. Transport and relaxation phenomena in porous media. *Adv. Chem. Phys.*, XCII:299, 1996.
9. 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.
10. A.J.C. Ladd and R. Verberg. Lattice-boltzmann simulations of particle-fluid suspensions. *J. Stat. Phys.*, 104(5):1191, 2001.
11. M. Rauscher, M. Krüger, A. Dominguez, and F. Penna. A dynamic density functional theory for particles in a flowing solvent. *J. Chem. Phys.*, 127:244906, 2007.