Multiscale Simulation of Turbulent Flow interacting with Porous Media based on a massively parallel implementation of the Cumulant Lattice Boltzmann Method

Konstantin Kutscher[a,*], Martin Geier[a], Manfred Krafczyk[a]

[a]*TU Braunschweig, Institute for Computational Modeling in Civil Engineering (iRMB), Braunschweig, Germany*

Abstract

Flow noise during takeoff and landing of commercial aircrafts can be substantially reduced by the use of porous surface layers in suitable sections of wing profiles. On the other hand (passive) porosity and roughness of surfaces tend to have an adverse effect on the boundary layer and thus on the lift of wings. This results in the need to be able to predict the aerodynamic effects of porous segments of the surface by numerical methods for aerodynamic design of wings, taking into account porosity and roughness as a function of Reynolds number. The application of the RANS equations for this task requires additional modeling terms such as the permeability and the Kozeny-Carman parameter as well as the turbulent fluctuations of the velocity field to be determined by Direct Numerical Simulation (DNS) / Large Eddy Simulation (LES) simulations based on the lattice Boltzmann method which is the main focus of this contribution. For the simulation of the flow at the pore scale we use the cumulant lattice Boltzmann method. Due to the inherent requirements of resolving both the turbulence on the scale of an airfoil and the flow inside the pore scale resolved porous medium the resulting simulations require more than a billion grid nodes on a refined three-dimensional mesh leading to massively parallel simulations. We discuss modeling and computational aspects of our approach and present computational results including experimental validations.

Keywords: lattice Boltzmann, cumulants, grid refinement, compact interpolation, porous media, turbulence

1. Introduction

Fluid flow through porous media and over porous surfaces occurs in many natural processes such as ground water flows, atmospheric flows over soil or in chemical engineering processes related to heat exchangers or catalytic surfaces. Despite the importance of porous media flow and the large amount of available literature, knowledge about transitional and turbulent flow in porous media and over porous surfaces on the microscopic level is still relatively limited. Beyond the pore scale Darcy's law is extended to the Ergun, Forchheimer or Brinkman's Equations which are assumed to be appropriate if the Reynolds number Re is larger than $O(10)$ where the actual value depends both on the internal structure of the porous medium and how Re is defined. The first scientific investigations of non-linear relations between driving pressure gradients and fluxes in porous media can be found e.g. in [1, 2, 3, 4, 5, 6, 7] which served as a basis for later investigations [8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18]. An overview of the topic is given by [19]. More recent work is discussed in [20, 21, 22, 23]. A focus on Reynolds Averaged Navier Stokes (RANS) closures for porous media is described in [24, 25, 26, 27]. While the latter work focuses on modeling the Darcy scale, i.e. it relies on homogenization approaches which do not take into account the detailed structure of the pore space, several works addressed the direct solution of the Stokes (for $Re \leq 1$) or Navier-Stokes equation for higher Reynolds number by either considering flow in periodic domains of basic geometric units such as bcc-type sphere packings [28] or reconstructions of the pore space based on tomographic data [29, 30]. Especially for the latter class of problems, the lattice Boltzmann approach has been accepted as a very reliable approach [31, 28, 32, 33, 34, 35, 36]. In terms of computational efficiency the cumulant LBM has been evaluated as one of the most promising schemes for

*Corresponding author
Email address: kutscher@irmb.tu-bs.de (Konstantin Kutscher)

porous media transport simulations in the saturated and laminar regime [37]. In this work we will demonstrate the suitability of this scheme also for the turbulent regime. The motivation for this study is the development of taylored porous media for noise reduction of wing profiles. Herr showed in [38] that the use of a porous trailing edge leads to a significant reduction of noise. The details of this mechanism seem to depend on various parameters such as the porosity and flow resistivity of the porous material and are not yet understood in detail. On the other hand, the application of a porous trailing edge can lead to a decrease of the lift performance. The aerodynamic effects on the scale of an aircraft can be investigated by means of volume and Reynolds-averaged Navier-Stokes (VRANS) modeling [39]. The application of this approach for wings with porous trailing edges requires additional terms that are based on Darcy and Forchheimer parameters [40]. These quantities as well as the turbulence statistics of the resulting boundary layers are being investigated with pore-scale simulations based on the cumulant lattice Boltzmann method which is implemented as part of the massively parallel VirtualFluids-LBM framework. The suitability of the cumulant LBM for the computation of acoustic absorption properties of porous media has already been demonstrated in [41].

The first part of the paper describes the numerical method which is used in the subsequent simulations. In the second part we present validation and simulation results of laminar and turbulent flows in boundary layer flows over porous surfaces.

2. The cumulant LBM kernel

The simulations in this paper are conducted with the cumulant lattice Boltzmann model [42, 43, 44, 45, 46, 47, 48, 49]. The cumulant lattice Boltzmann model belongs to the class of lattice Boltzmann models with multiple independent relaxation times for different observable quantities. In this context the term Multi Relaxation Time model is usually reserved for models based on a raw moment transformation [50, 51, 52, 53]. The raw moment representation in these models caused some violations of Galilean invariance, which was overcome by replacing the raw moments by central moments (centered around the velocity of the fluid at the local grid node) in the so-called cascaded lattice Boltzmann methods [54, 55, 56, 57, 58, 59] which has subsequently been improved to the Factorized Central Moment method to remove part of the hyperviscosity [60, 61, 62, 63] and saw various extensions, for example for the simulation of multi-phase flow [64, 65, 66] and thermal flows [67] and for the efficient simulation of steady state flows [68]. The central moment methods improve Galilean invariance as compared to the raw moment approach. However, a spurious coupling between the equivalent partial differential equations for the moments remain [42]. This implies that the leading error strongly depends on the choice of the relaxation rates. This problem has now been solved by replacing the central moments in the collision operator by cumulants [69, 42]. Cumulants are statistically independent observable quantities. The cumulants of a physical system can be understood as the intensive physical quantities such as velocity and temperature. In comparison, the corresponding moments are understood as momentum and internal energy. It is interesting to note that the cumulants, which are mathematically more complicated than moments, appear to be simpler from an intuitive point of view. Cumulants and moments contain the same physical information, but it is the cumulants that initiate fluxes between bodies, for example momentum flux between two bodies of different velocity and energy flux between bodies of different temperature. Cumulants determine whether or not two bodies are in equilibrium and are therefore a natural basis for describing the equilibration process in a particle collision which is at the heart of the lattice Boltzmann method.

In the past, the cumulant lattice Boltzmann equation has been used mainly in the so-called compressible form with variable density [42]. For the simulation of incompressible flow through a porous material of high hydrodynamical resistance it is advisable to use the fixed density formulation of the lattice Boltzmann equation which is also (inaccurately) called the incompressible lattice Boltzmann equation [70]. To impose a fixed density let us first introduce the density variation $\delta\rho$:

$$\delta\rho = \sum_{i,j,k=-1}^{1} f_{ijk}. \tag{1}$$

Here i, j, k are the quantum numbers for the discrete velocities and f_{ijk} is the corresponding velocity distribution function. The cumulant generating function of a distribution is defined as the logarithm of its Laplace transform. Cumulants are the coefficients of the series expansion of the cumulant generating function. They are obtained by

evaluating successive derivatives at the origin. For a fixed density we define cumulants with a distribution modified to yield unit density [37]:

$$C_{\alpha\beta\gamma} = c^{-\alpha\beta\gamma} \frac{\partial^\alpha \partial^\beta \partial^\gamma}{\partial \Xi^\alpha \partial \Upsilon^\beta \partial Z^\gamma} \ln(\mathcal{L}\{f_{ijk} + w_{ijk}(1-\delta\rho)\})\Big|_{\Xi=\Upsilon=Z=0}. \tag{2}$$

Here $\mathcal{L}$ represents the three-dimensional Laplace transform in velocity space. The variables Ξ, Υ and Z denote the velocity space wave numbers. The distribution function f_{ijk} is already discretized in velocity space. To take the Laplace transform the discretized distribution is written with the Dirac-delta function as [42]:

$$f(\xi, \upsilon, \zeta) = \sum_{i,j,k} f_{ijk} \delta(ic - \xi)\delta(jc - \upsilon)\delta(kc - \zeta), \tag{3}$$

where ξ, υ and ζ are the components of microscopic velocity corresponding to Ξ, Υ and Z respectively. The transformed distribution is a spectral representation of the original distribution. The lattice velocity is $c = \Delta x/\Delta t$, the ratio of grid spacing and time step. The weights w_{ijk} for the D3Q27 model under consideration here are explicitly given as $w_{000} = 8/27$, $w_{100} = 2/27$, $w_{110} = 1/54$, $w_{111} = 1/216$ and so on by permutation of indices. Equation (2) normalizes the distribution function at any node to unit density and the cumulant transform can be applied the usual way. Computing cumulants directly from (2) is not straightforward. Instead, it is instructive to compare eq. (2) to the formal definition of raw moments:

$$M_{\alpha\beta\gamma} = c^{-\alpha\beta\gamma} \frac{\partial^\alpha \partial^\beta \partial^\gamma}{\partial \Xi^\alpha \partial \Upsilon^\beta \partial Z^\gamma} \mathcal{L}\{f_{ijk} + w_{ijk}(1-\delta\rho)\}\Big|_{\Xi=\Upsilon=Z=0} = \sum_{i,j,k} i^\alpha j^\beta k^\gamma (f_{ijk} + w_{ijk}(1-\delta\rho)). \tag{4}$$

Equation (2) and (4) are distinguished only by the presence of the logarithm in (2). By applying the chain rule to (2) it is possible to express all cumulants in terms of raw moments. In this way it is possible to compute cumulants directly from evaluating the simplified expression on the right of equation (4). For obtaining explicit equations expressing cumulants in terms of moments it is recommended to evaluate the following expressions in a computer algebra system:

$$C_{\alpha\beta\gamma} = \frac{\partial^\alpha \partial^\beta \partial^\gamma}{\partial \Xi^\alpha \partial \Upsilon^\beta \partial Z^\gamma} \ln(\Theta(\Xi, \Upsilon, Z)). \tag{5}$$

This will produce an equation in derivatives of an arbitrary function $\Theta(\Xi, \Upsilon, Z)$, each of which can be identified with a raw moment, i.e.:

$$\mathcal{M}_{\alpha\beta\gamma} = \frac{\partial^\alpha \partial^\beta \partial^\gamma}{\partial \Xi^\alpha \partial \Upsilon^\beta \partial Z^\gamma} \Theta(\Xi, \Upsilon, Z). \tag{6}$$

The relationship between $C_{\alpha\beta\gamma}$ and $M_{\alpha\beta\gamma}$ is the same as between $C_{\alpha\beta\gamma}$ and $\mathcal{M}_{\alpha\beta\gamma}$ such that cumulants can be computed from raw moments. For example considering the case C_{200}:

$$\begin{aligned} C_{200} &= \frac{\partial^2}{\partial \Xi^2} \ln(\Theta(\Xi, \Upsilon, Z)) && (7)\\ &= \frac{\frac{\partial^2}{\partial \Xi^2}\Theta(\Xi, \Upsilon, Z)}{\Theta(\Xi, \Upsilon, Z)} - \frac{\left(\frac{\partial}{\partial \Xi}\Theta(\Xi, \Upsilon, Z)\right)^2}{\Theta(\Xi, \Upsilon, Z)} && (8)\\ &= \frac{\mathcal{M}_{200}}{\mathcal{M}_{000}} - \frac{\mathcal{M}_{100}^2}{\mathcal{M}_{000}}. && (9) \end{aligned}$$

Hence by analogy C_{200} is computed as:

$$C_{200} = \frac{M_{200}}{M_{000}} - \frac{M_{100}^2}{M_{000}}. \tag{10}$$

In the following we will make use of the order of cumulants by which we mean the sum of the indices $\alpha + \beta + \gamma$. Asymptotic analysis as shown in Appendix G of [42] and in [48] reveals the relationship between the cumulants and

the Navier-Stokes equation. Only cumulants up to third order are required for consistency with the Navier-Stokes equation. However, cumulants of higher order have a strong influence on the numerical accuracy. This is in particular true for cumulants of order four which appear at the same order as the leading error of the scheme. Through setting appropriate relaxation rates for fourth order cumulants the leading error can be substantially reduced as recently shown in [48]. However, those recent results were not yet considered in the current study.

For implementation purposes it is not optimal to compute the moments by the definitions (4) directly since this would add a constant offset to all distributions which would have an adverse effect on the numerical conditioning of the cumulant transformation. In order to improve the conditioning of the cumulant transform we eliminated the constant offset step by step from all forthcoming operations. Although the resulting expressions are somewhat complicated, we list them below in order to enable the reader to implement our method and reproduce our results.

The first order cumulants are the velocity components, which in the case of the constant density lattice Boltzmann equation are computed without considering the mass:

$$u = F_x/2 + \sum_{i,j,k=-1}^{1} i f_{ijk}, \qquad v = F_y/2 + \sum_{i,j,k=-1}^{1} j f_{ijk}, \qquad w = F_z/2 + \sum_{i,j,k=-1}^{1} k f_{ijk}. \tag{11}$$

Here F_x, F_y, and F_z denote the components of a forcing term if applicable.

2.1. Forward central moment transformation

Cumulants are efficiently computed from central moments. Central moments can be computed directly through:

$$\kappa_{\alpha\beta\gamma} = \sum_{i,j,k=-1}^{1} (i-u)^{\alpha}(j-v)^{\beta}(k-w)^{\gamma} f_{ijk}. \tag{12}$$

Instead of using the direct way we use the fast central moment transform:

$$\begin{aligned}
\kappa_{ij|0} &= (f_{ij1} + f_{ij\bar{1}}) + \mathrm{f}_{ij0}, && (13)\\
\kappa_{ij|1} &= (f_{ij1} - f_{ij\bar{1}}) - w(\kappa_{ij|0} + \mathrm{K}_{ij|0}(1-\delta\rho)), && (14)\\
\kappa_{ij|2} &= (f_{ij1} + f_{ij\bar{1}}) - 2w(f_{ij1} - \mathrm{f}_{ij\bar{1}}) + w^2(\kappa_{ij|0} + \mathrm{K}_{ij|0}(1-\delta\rho)), && (15)\\
\kappa_{i|0\gamma} &= (\kappa_{i1|\gamma} + \kappa_{i\bar{1}|\gamma}) + \kappa_{i0|\gamma}, && (16)\\
\kappa_{i|1\gamma} &= (\kappa_{i1|\gamma} - \kappa_{i\bar{1}|\gamma}) - v(\kappa_{i|0\gamma} + \mathrm{K}_{i|0\gamma}(1-\delta\rho)), && (17)\\
\kappa_{i|2\gamma} &= (\kappa_{i1|\gamma} + \kappa_{i\bar{1}|\gamma}) - 2v(\kappa_{i1|\gamma} - \kappa_{i\bar{1}|\gamma}) + v^2(\kappa_{i|0\gamma} + \mathrm{K}_{i|0\gamma}(1-\delta\rho)), && (18)\\
\kappa_{0\beta\gamma} &= (\kappa_{1|\beta\gamma} + \kappa_{\bar{1}|\beta\gamma}) + \kappa_{0|\beta\gamma}, && (19)\\
\kappa_{1\beta\gamma} &= (\kappa_{1|\beta\gamma} - \kappa_{\bar{1}|\beta\gamma}) - u(\kappa_{0\beta\gamma} + \mathrm{K}_{0\beta\gamma}(1-\delta\rho)), && (20)\\
\kappa_{2\beta\gamma} &= (\kappa_{1|\beta\gamma} + \kappa_{\bar{1}|\beta\gamma}) - 2u(\kappa_{1|\beta\gamma} - \kappa_{\bar{1}|\beta\gamma}) + u^2(\kappa_{0\beta\gamma} + \mathrm{K}_{0\beta\gamma}(1-\delta\rho)). && (21)
\end{aligned}$$

The symbols on the left hand side of equations (13) to (18) denote chimeras that are partly moments and partly distributions. For example the chimera $\kappa_{00|0}$ can be understood as a distribution in its first two indices ($x-$ and $y-$dimensions) while it is a central moment in $z-$dimension. The chimeras are only used inside the transformation for reducing the computational cost. Equations (13) to (21) represent a divide and conquer algorithm for the fast computation of central moments. The fast central moment transform is equivalent to the direct central moment transform but it requires only a third of the floating point operations of the latter.

Since we started from the original distributions f_{ijk} rather than from the modified ones (i.e. $f_{ijk} + w_{ijk}(1-\delta\rho)$) we have introduced the constant parameters K that can be computed from the weights w_{ijk}:

$$\mathrm{K}_{ij|\gamma} = \sum_k k^\gamma w_{ijk}, \tag{22}$$

$$\mathrm{K}_{i|\beta\gamma} = \sum_j j^\beta \mathrm{K}_{ij|\gamma}, \tag{23}$$

$$\mathrm{K}_{\alpha\beta\gamma} = \sum_i i^\alpha \mathrm{K}_{i|\beta\gamma}. \tag{24}$$

2.2. Forward cumulant transformation

The non-conserved cumulants are computed from the central moments. In what follows we omit equations for cumulants which can be obtained by permuting indices in the listed equations. The first few cumulants are seen to be identical to the first few central moments:

$$C_{110} = \kappa_{110}, \tag{25}$$

$$C_{200} = \kappa_{200}, \tag{26}$$

$$C_{120} = \kappa_{120}, \tag{27}$$

$$C_{111} = \kappa_{111}. \tag{28}$$

Equations for C_{101} and C_{011} and so on are obtained by permuting the indices in the above equations. Differences from central moments start at order four:

$$C_{211} = \kappa_{211} - ((\kappa_{200} + 1/3)\kappa_{011} + 2\kappa_{110}\kappa_{101}), \tag{29}$$

$$C_{220} = \kappa_{220} - ((\kappa_{200}\kappa_{020} + 2\kappa_{110}^2) + (\kappa_{200} + \kappa_{020})/3(1-\delta\rho) - \delta\rho(1-\delta\rho)/9), \tag{30}$$

$$C_{122} = \kappa_{122} - ((\kappa_{002}\kappa_{120} + \kappa_{020}\kappa_{102} + 4\kappa_{011}\kappa_{111} + 2(\kappa_{101}\kappa_{021} + \kappa_{110}\kappa_{012})) + (\kappa_{120} + \kappa_{102})(1-\delta\rho)/3), \tag{31}$$

$$\begin{aligned} C_{222} = {} & \kappa_{222} - (4\kappa_{111}^2 - 16\kappa_{011}\kappa_{101}\kappa_{110} - 2\kappa_{220}\kappa_{202}\kappa_{022} + \kappa_{022}\kappa_{200} + \kappa_{202}\kappa_{020} + \kappa_{022}\kappa_{200} \\ & -4(\kappa_{020}\kappa_{101}^2 + \kappa_{200}\kappa_{011}^2 + \kappa_{002}\kappa_{110}^2) + 2(\kappa_{102}\kappa_{120} + \kappa_{012}\kappa_{210} + \kappa_{021}\kappa_{201}) \\ & -\frac{2}{3}(\kappa_{002}\kappa_{020} + \kappa_{002}\kappa_{200} + \kappa_{020}\kappa_{200} + 2(\kappa_{011}^2 + \kappa_{101}^2 + \kappa_{110}^2))(1-\delta\rho) \\ & +\frac{1}{3}(\kappa_{022} + \kappa_{202} + \kappa_{220})(1-\delta\rho) - \frac{1}{9}(\kappa_{002} + \kappa_{020} + \kappa_{200})(1-\delta\rho)(1-2\delta\rho) \\ & +4(\kappa_{112}\kappa_{110} + \kappa_{121}\kappa_{101} + \kappa_{211}\kappa_{011}) - \frac{1}{27}((1-\delta\rho) - 3(1-\delta\rho)^2 + 2(1-\delta\rho)^3)), \end{aligned} \tag{32}$$

and so on by permuting indices.

2.3. Collision

The collision thus explicitly reads:

$$C_{110}^* = (1-\omega_1)C_{110}, \tag{33}$$

$$C_{101}^* = (1-\omega_1)C_{101}, \tag{34}$$

$$C_{011}^* = (1-\omega_1)C_{011}. \tag{35}$$

For correcting some error terms and for enhancing the interpolation order in the grid refinement it is useful to calculate the first derivatives of the velocity:

$$
\begin{aligned}
D_x u &= -\frac{\omega_1}{2}(2C_{200} - C_{020} - C_{002}) - \frac{\omega_2}{2}(C_{200} + C_{020} + C_{002} - \kappa_{000}), && (36)\\
D_y v &= D_x u + \frac{3\omega_1}{2}(C_{200} - C_{020}), && (37)\\
D_z w &= D_x u + \frac{3\omega_1}{2}(C_{200} - C_{002}), && (38)\\
D_x v + D_y u &= -3\omega_1 C_{110}, && (39)\\
D_x w + D_z u &= -3\omega_1 C_{101}, && (40)\\
D_y w + D_z v &= -3\omega_1 C_{011}, && (41)
\end{aligned}
$$

We note that Equations (39) to (41) are required only for the grid refinement and might be skipped in the usual collision step. We also note that only the shear rate can be computed directly from the cumulants which is not possible for the vorticity. Equations (36) to (38) are required to modify the following cumulants in their equilibrium:

$$
\begin{aligned}
C^*_{200} - C^*_{020} &= (1-\omega_1)(C_{200} - C_{020}) - 3\left(1 - \frac{\omega_1}{2}\right)(u^2 D_x u - v^2 D_y v), && (42)\\
C^*_{200} - C^*_{002} &= (1-\omega_1)(C_{200} - C_{002}) - 3\left(1 - \frac{\omega_1}{2}\right)(u^2 D_x u - w^2 D_z w), && (43)\\
C^*_{200} + C^*_{020} + C^*_{002} &= \kappa_{000}\omega_2 + (1-\omega_2)(C_{200} + C_{020} + C_{002}) - 3\left(1 - \frac{\omega_2}{2}\right)(u^2 D_x u + v^2 D_y v + w^2 D_z w). && (44)
\end{aligned}
$$

Computing the gradients of velocity from the second order cumulants and feeding them back into the equilibria of the corresponding cumulants improves the Galilean invariance of the viscosity. Details on this can be found in Appendix H of [42].

The equilibria for all remaining cumulants are zero:

$$
\begin{aligned}
C^*_{120} + C^*_{102} &= (1-\omega_3)(C_{120} + C_{102}), && (45)\\
C^*_{210} + C^*_{012} &= (1-\omega_3)(C_{210} + C_{012}), && (46)\\
C^*_{201} + C^*_{021} &= (1-\omega_3)(C_{201} + C_{021}), && (47)\\
C^*_{120} - C^*_{102} &= (1-\omega_4)(C_{120} - C_{102}), && (48)\\
C^*_{210} - C^*_{012} &= (1-\omega_4)(C_{210} - C_{012}), && (49)\\
C^*_{201} - C^*_{021} &= (1-\omega_4)(C_{201} - C_{021}), && (50)\\
C^*_{111} &= (1-\omega_5)C_{111}, && (51)\\
C^*_{220} - 2C^*_{202} + C^*_{022} &= (1-\omega_6)(C_{220} - 2C_{202} + C_{022}), && (52)\\
C^*_{220} + C^*_{202} - 2C^*_{022} &= (1-\omega_6)(C_{220} + C_{202} - 2C_{022}), && (53)\\
C^*_{220} + C^*_{202} + C^*_{022} &= (1-\omega_7)(C_{220} + C_{202} + C_{022}), && (54)\\
C^*_{211} &= (1-\omega_8)C_{211}, && (55)\\
C^*_{121} &= (1-\omega_8)C_{121}, && (56)\\
C^*_{112} &= (1-\omega_8)C_{112}, && (57)\\
C^*_{221} &= (1-\omega_9)C_{221}, && (58)\\
C^*_{212} &= (1-\omega_9)C_{212}, && (59)\\
C^*_{122} &= (1-\omega_9)C_{122}, && (60)\\
C^*_{222} &= (1-\omega_{10})C_{222}. && (61)
\end{aligned}
$$

Note that the relaxation of the third order and fourth order cumulants are grouped. This grouping is irrelevant as long as $\omega_3 = \omega_4$ and $\omega_6 = \omega_7$. Grouping the cumulants in this way provides us with the largest number of

independently adjustable relaxation rates without compromising the isotropy of the method. The grouping is based on the rotational properties of the respective quantities. For example $C_{120} + C_{102}$ is invariant under rotation about the Ξ axis while $C_{120} - C_{102}$ is sensitive to such a rotation. The combination $C_{220} + C_{202} + C_{022}$ is the isotropic part of the fourth order cumulants and is hence given another relaxation rate than the non-isotropic fourth order cumulants. This grouping becomes essential when the relaxation rates of the higher order cumulants are optimized for higher accuracy [48]. The higher order cumulants or moments are often dismissed as unimportant but they play a big role in superconvergent lattice Boltzmann schemes as recently shown for the diffusion part [48, 49] and the advection part [71] of the LBM. The higher order moments can also be useful for solving for additional equations as has been recently shown for the recovery of the vorticity from the fifth order moments in a MRT setting [72]. The same approach can be directly implemented in the cumulant method too. However, using the fifth order moments/cumulants to compute the vorticity implies the wrong value for the fifth cumulants which inevitably limits the accuracy to second order accuracy while the cumulant method with correct fifth moments can be fourth order accurate when using the proper parametrization [48].

2.4. Backward cumulant transformation

After the collision the cumulants have to be transformed back into central moments:

$$\kappa^*_{211} = C^*_{211} + ((\kappa^*_{200} + 1/3)\kappa^*_{011} + 2\kappa^*_{110}\kappa^*_{101}), \tag{62}$$

$$\kappa^*_{220} = C^*_{220} + ((\kappa^*_{200}\kappa^*_{020} + 2\kappa^{*2}_{110}) + (\kappa^*_{200} + \kappa^*_{020})/3(1-\delta\rho) - \delta\rho(1-\delta\rho)/9), \tag{63}$$

$$\kappa^*_{122} = C^*_{122} + ((\kappa^*_{002}\kappa^*_{120} + \kappa^*_{020}\kappa^*_{102} + 4\kappa^*_{011}\kappa^*_{111} + 2(\kappa^*_{101}\kappa^*_{021} + \kappa^*_{110}\kappa^*_{012})) + (\kappa^*_{120} + \kappa^*_{102})(1-\delta\rho)/3), \tag{64}$$

$$\begin{aligned}\kappa^*_{222} = {} & C^*_{222} + (4\kappa^{*2}_{111} - 16\kappa^*_{011}\kappa^*_{101}\kappa^*_{110} - 2\kappa^*_{220}\kappa^*_{202}\kappa^*_{022} + \kappa^*_{022}\kappa^*_{200} + \kappa^*_{202}\kappa^*_{020} + \kappa^*_{022}\kappa^*_{200} \\ & -4(\kappa^*_{020}\kappa^{*2}_{101} + \kappa^*_{200}\kappa^{*2}_{011} + \kappa^*_{002}\kappa^{*2}_{110}) + 2(\kappa^*_{102}\kappa^*_{120} + \kappa^*_{012}\kappa^*_{210} + \kappa^*_{021}\kappa^*_{201}) \\ & -\frac{2}{3}(\kappa^*_{002}\kappa^*_{020} + \kappa^*_{002}\kappa^*_{200} + \kappa^*_{020}\kappa^*_{200} + 2(\kappa^{*2}_{011} + \kappa^{*2}_{101} + \kappa^{*2}_{110}))(1-\delta\rho) \\ & +\frac{1}{3}(\kappa^*_{022} + \kappa^*_{202} + \kappa^*_{220})(1-\delta\rho) - \frac{1}{9}(\kappa^*_{002} + \kappa^*_{020} + \kappa^*_{200})(1-\delta\rho)(1-2\delta\rho) \\ & +4(\kappa^*_{112}\kappa^*_{110} + \kappa^*_{121}\kappa^*_{101} + \kappa^*_{211}\kappa^*_{011}) - \frac{1}{27}((1-\delta\rho) - 3(1-\delta\rho)^2 + 2(1-\delta\rho)^3)).\end{aligned} \tag{65}$$

The remaining central moments are obtained by permuting the corresponding indices. Second and third order central moments are directly obtained from the cumulants according to equations (25)-(28). In order for the forcing to take effect, the first central moments have to change sign:

$$\kappa^*_{100} = -\kappa_{100}, \tag{66}$$

$$\kappa^*_{010} = -\kappa_{010}, \tag{67}$$

$$\kappa^*_{001} = -\kappa_{001}. \tag{68}$$

This is done because half the force was applied in a different frame of reference prior to the central moment transformation.

2.5. Backward central moment transformation

The back transformation to distributions reads:

$$\kappa^*_{0|\beta\gamma} = \kappa^*_{0\beta\gamma}(1-(u/c)^2) - 2(u/c)\kappa^*_{1\beta\gamma} - \kappa^*_{2\beta\gamma} - K_{0\beta\gamma}(1-\delta\rho)(u/c)^2, \quad (69)$$

$$\kappa^*_{\bar{1}|\beta\gamma} = ((\kappa^*_{0\beta\gamma} + \mathrm{K}_{0\beta\gamma}(1-\delta\rho))((u/c)^2 - u/c) + \kappa^*_{1\beta\gamma}(2u/c-1) + \kappa^*_{2\beta\gamma})/2, \quad (70)$$

$$\kappa^*_{1|\beta\gamma} = ((\kappa^*_{0\beta\gamma} + \mathrm{K}_{0\beta\gamma}(1-\delta\rho))((u/c)^2 + u/c) + \kappa^*_{1\beta\gamma}(2u/c+1) + \kappa^*_{2\beta\gamma})/2, \quad (71)$$

$$\kappa^*_{i0|\gamma} = \kappa^*_{i|0\gamma}(1-(v/c)^2) - 2(v/c)\kappa^*_{i|1\gamma} - \kappa^*_{i|2\gamma} - K_{i|0\gamma}(1-\delta\rho)(v/c)^2, \quad (72)$$

$$\kappa^*_{i\bar{1}|\gamma} = ((\kappa^*_{i|0\gamma} + \mathrm{K}_{i|0\gamma}(1-\delta\rho))((v/c)^2 - v/c) + \kappa^*_{i|1\gamma}(2v/c-1) + \kappa^*_{i|2\gamma})/2, \quad (73)$$

$$\kappa^*_{i1|\gamma} = ((\kappa^*_{i|0\gamma} + \mathrm{K}_{i|0\gamma}(1-\delta\rho))((v/c)^2 + v/c) + \kappa^*_{i|1\gamma}(2v/c+1) + \kappa^*_{i|2\gamma})/2, \quad (74)$$

$$f^*_{ij0} = \kappa^*_{ij|0}(1-(w/c)^2) - 2(w/c)\kappa^*_{ij|1} - \kappa^*_{ij|2} - K_{ij|0}(1-\delta\rho)(w/c)^2, \quad (75)$$

$$f^*_{ij\bar{1}} = ((\kappa^*_{ij|0} + \mathrm{K}_{ij|0}(1-\delta\rho))((w/c)^2 - w/c) + \kappa^*_{ij|1}(2w/c-1) + \kappa^*_{ij|2})/2, \quad (76)$$

$$f^*_{ij1} = ((\kappa^*_{ij|0} + \mathrm{K}_{ij|0}(1-\delta\rho))((w/c)^2 + w/c) + \kappa^*_{ij|1}(2w/c+1) + \kappa^*_{ij|2})/2. \quad (77)$$

Note that we write the crystallographic indices of the lattice directions in Miller notation with $\bar{1} = -1$ [73].

2.6. Relaxation rates

From the different relaxation rates only ω_1 has a leading order influence on the results. The kinematic shear viscosity ν of the model is set by choosing ω_1 such that:

$$\nu = \frac{1}{3}\left(\frac{1}{\omega_1} - \frac{1}{2}\right)\frac{\Delta x^2}{\Delta t}. \quad (78)$$

The remaining relaxation rates are set here to unity ($\omega_2 = \cdots = \omega_{10} = 1$). This is a very stable choice but more accurate choices exist [48].

3. Grid refinement

Grid refinement for the lattice Boltzmann method has been pioneered twenty years ago by Filippova and Hänel [74, 75] but remains to be an active subject of research [76, 77, 78, 79]. The idea behind the grid-refinement is to use nested Cartesian grids of different resolution which allow to apply the *unmodified* lattice Boltzmann equation in the bulk of the domain and to provide suitable coupling mechanisms at the interfaces between such domains. We shall note here that not all grid refinement techniques conform to this principle [80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91]. Those models are not discussed here.

It is common to distinguish between interpolation methods [74, 75, 92, 93, 76, 94, 78] and volumetric methods [95, 96, 79]. Interpolation methods couple two grids of different resolution by interpolation of the distributions while volumetric methods split up the Wigner-Seitz cells of the lattice nodes when going from the coarse to the fine grid and average over several Wigner-Seitz cells when going from fine to coarse grids. Despite their popularity, the interpolation methods suffer from certain problems that render them useless for flows of high Reynolds number unless certain correction steps are introduced [76, 78]. The most important problem of these models is that no interpolation is usually performed when distributions are moved from the fine to the coarse grids since all nodes on the coarse grids have matching neighbors on the fine grid side. This of course violates the Nyquist-Shannon sampling theorem [97] according to which the signal being decimated must not contain any frequencies which cannot be represented on the coarse grid side. It is hence obvious that a naive transfer of data from the high resolution grid to the lower resolution grid must fail unless the simulation is well resolved also on the coarse grid side. This is certainly not the case in large eddy simulations and this is the reason why the interpolation methods needs to be supplemented by explicit filtering [76]. Volumetric grid coupling does not suffer from this problem since the grids of different resolution are staggered in space and this alone acts as an implicit low pass filter that conforms with the Nyquist-Shannon theorem. Volumetric grid refinement techniques are therefore more robust than interpolation methods. However, it is important to understand that this is not due to the way data is mapped from one grid to the other (interpolation versus volumetric) but it results from the differences in the alignments of the grids (over-lapping versus staggered).

Recognizing this important difference we proposed [98, 99, 100] to use compact interpolation on staggered girds, and thereby combining the advantages of both approaches with the additional advantage of quadratic interpolation order on a minimal stencil. The method described below has successfully be applied to many applications including, in particular, highly turbulent flows [42, 43, 44, 45, 46, 49]. It is suitable for block structured grids and list based grids [101]. Recently, a similar method has been used by Qi et al. [102].

Our grid refinement technique works by coupling two grids of different resolution through interpolation. At each interface the grid is refined by a factor of two and we use acoustic scaling such that two time steps on the fine grid correspond to one time step on the coarse grid. Interpolation at the interface between two grids serve two purposes: First, the hydrodynamic quantities have to be transferred from one grid to the other at the highest affordable accuracy. Second, the data going from the fine grid to the coarse grid have to be filtered in such a way that they do not contain waves which cannot be represented on the coarse grid. At the same time the interpolation of the velocity field should be quadratic in space such that no numerical viscosity is introduced.

For the compact interpolation of the velocity components we start from quadratic interpolation functions:

$$u(x,y,z) = a_0 + a_x x + a_y y + a_z z + a_{xy}xy + a_{xz}xz + a_{yz}yz + a_{xx}x^2 + a_{yy}y^2 + a_{zz}z^2 + a_{xyz}xyz, \quad (79)$$

$$v(x,y,z) = b_0 + b_x x + b_y y + b_z z + b_{xy}xy + b_{xz}xz + b_{yz}yz + b_{xx}x^2 + b_{yy}y^2 + b_{zz}z^2 + b_{xyz}xyz, \quad (80)$$

$$w(x,y,z) = c_0 + d_x x + c_y y + c_z z + c_{xy}xy + c_{xz}xz + c_{yz}yz + c_{xx}x^2 + c_{yy}y^2 + c_{zz}z^2 + c_{xyz}xyz. \quad (81)$$

To determine the 33 coefficients in the equations we use the known velocities u, v, w at the nodes assumed to be at positions $\{\pm 1/2, \pm 1/2, \pm 1/2\}$ in the local coordinate system centered in the middle of the cell. However, these are only $8 \times 3 = 24$ independent quantities. The missing nine values are obtained from the gradients of velocity as computed from the pre-collision cumulants by equations (36) to (41). However, in order to to do this we have to reduce the $8 \times 5 = 40$ derivatives at the eight nodes to nine values. This is done by averaging each of the five derivatives in two spatial directions and taking the central difference in the third dimension. For example, we obtain the second derivative $D_x(D_x v + D_y u)$ by:

$$D_x(D_x v + D_y u) = \frac{1}{4}\left(\sum_{j,k=0}^{1} (D_x v + D_y u)_{(1/2)(j-1/2)(k-1/2)} - (D_x v + D_y u)_{(-1/2)(j-1/2)(k-1/2)} \right). \quad (82)$$

where the indices of the derivatives refer to the coordinates in x, y and z direction in the local coordinate system of the source cell. By taking the corresponding derivatives of the (79) - (81) we find the following relationships:

$$D_x(D_x v + D_y u) = 2b_{xx} + a_{xy}, \quad (83)$$

$$D_y(D_x v + D_y u) = 2a_{yy} + b_{xy}, \quad (84)$$

$$D_x(D_x w + D_z u) = 2c_{xx} + a_{xz}, \quad (85)$$

$$D_z(D_x v + D_z w) = 2a_{zz} + c_{xz}, \quad (86)$$

$$D_y(D_y w + D_z u) = 2c_{yy} + b_{yz}, \quad (87)$$

$$D_z(D_y w + D_z u) = 2b_{zz} + c_{yz}, \quad (88)$$

$$D_x(D_x u) = 2a_{xx}, \quad (89)$$

$$D_y(D_y v) = 2b_{yy}, \quad (90)$$

$$D_z(D_z w) = 2c_{zz}. \quad (91)$$

The remaining information is taken from the actual velocities at the nodes. The resulting set of equations is solved for the parameters $a_\star$, $b_\star$ and $c_\star$. In addition to the velocity the density must be interpolated which is done by tri-linear interpolation. The interpolation function reads:

$$\delta\rho(x,y,z) = d_0 + d_x + d_y y + d_z z + d_{xy}xy + d_{xz}xz + d_{yz}yz + d_{xyz}xyz. \quad (92)$$

The parameters are obtained by equating $\delta\rho(x,y,z)$ with the densities at the eight vertexes of the source cell. Depending on whether the grid refinement is used for the lattice Boltzmann equation with varying density [42] or with constant density we define for the case with varying density: $\rho = 1 + \delta\rho$ and for the constant density case as used in the current paper $\rho = 1$.

By evaluating the interpolation functions $u(x,y,z)$, $v(x,y,z)$, $w(x,y,z)$ and $\delta\rho(x,y,z)$ at the coordinates of the destination nodes, we can compute the macroscopic quantities at these nodes. In addition, we can easily compute the non-equilibrium part of the second order cumulants. We know from asymptotic analysis that they are related to the derivatives of velocity via equations (36) to (41). The derivatives of the velocities are obtained by taking the analytical derivative of the polynomials (79) to (81). Thus, the pre-collision second order cumulants at the destination nodes are computed as:

$$C_{011} = -\frac{\sigma\rho}{3\omega_{1d}}(b_z + c_y + \mathcal{A}_{011}(x,y,z)), \tag{93}$$

$$C_{101} = -\frac{\sigma\rho}{3\omega_{1d}}(a_z + c_x + \mathcal{A}_{101}(x,y,z)), \tag{94}$$

$$C_{110} = -\frac{\sigma\rho}{3\omega_{1d}}(a_y + b_x + \mathcal{A}_{110}(x,y,z)), \tag{95}$$

$$C_{200} = \frac{\delta\rho}{3} - \frac{2\sigma\rho}{9\omega_1 d}(2a_x - b_y - c_z + \mathcal{B}(x,y,z) + C(x,y,z)), \tag{96}$$

$$C_{020} = \frac{\delta\rho}{3} - \frac{2\sigma\rho}{9\omega_1 d}(2(b_y - \mathcal{B}(x,y,z)) - a_x - c_z + C(x,y,z)), \tag{97}$$

$$C_{002} = \frac{\delta\rho}{3} - \frac{2\sigma\rho}{9\omega_1 d}(-a_x - b_y + \mathcal{B}(x,y,z) + 2(c_z - C(x,y,z))), \tag{98}$$

where ω_{1d} is the value of ω_1 on the destination grid. The parameter σ is introduced to scale the derivatives to the coordinate system of the destination grid. For coarse to fine interpolation it takes the value $\sigma_{c\to f} = 1/2$ and for fine to coarse interpolation it takes the value $\sigma_{f\to c} = 2$.

The space dependent functions are:

$$\mathcal{A}_{011}(x,y,z) = b_{xz}x + c_{xy}x + b_{zy}y + 2c_{yy}y + b_{xyz}xy + 2b_{zz}z + c_{yz}z + c_{xyz}xz, \tag{99}$$

$$\mathcal{A}_{111}(x,y,z) = a_{xz}x + 2c_{xx}x + a_{yz}y + 2c_{xy}y + a_{xyz}xy + 2a_{zz}z + c_{xz}z + c_{xyz}yz, \tag{100}$$

$$\mathcal{A}_{110}(x,y,z) = a_{xy}x + 2b_{xx}x + 2a_{yy}y + b_{xy}y + a_{yz}z + b_{xz}z + a_{xyz}xz + b_{xyz}yz, \tag{101}$$

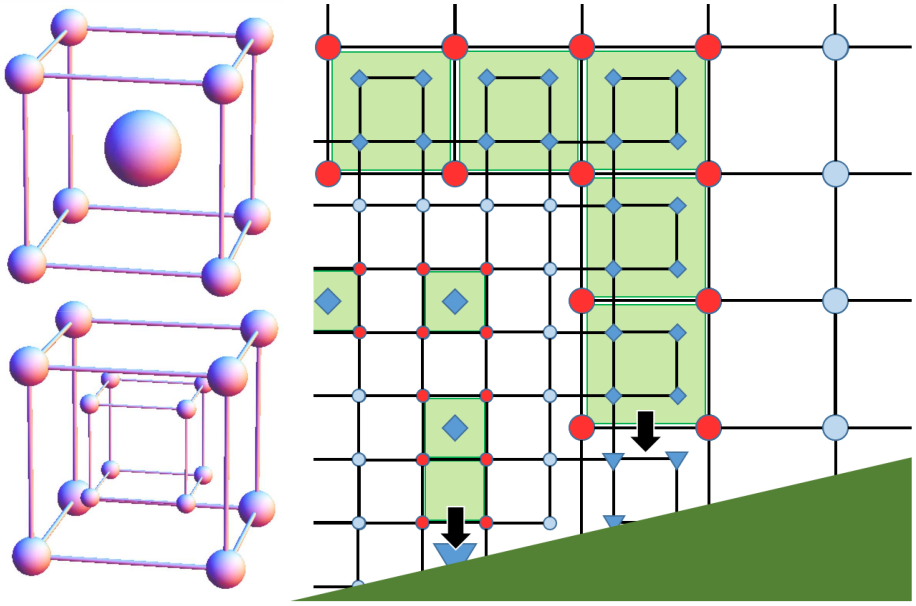
$$\mathcal{B}(x,y,z) = 2a_{xx}x - b_{xy}x + a_{xy}y - 2b_{yy}y + a_{xz}z - b_{yz}z - b_{xyz}xz + a_{xyz}yz, \tag{102}$$

$$C(x,y,z) = 2a_{xx}x - c_{xz}x + a_{xy}y - c_{yz}y - c_{xyz}xy + a_{xz}z - 2c_{zz}z + a_{xyz}yz. \tag{103}$$

Beside the conserved cumulants and the cumulants of second order all remaining cumulants can be set to zero.

The coupling algorithm between the grids is analogous to the two-dimensional case described in [103, 100]. For each coarse time step there are two fine time steps, one that is synchronous with the coarse time step and one that is asynchronous. Instead of interpolating in time we use a small overlap between the grids. Before the synchronous time step, the pre-collision distribution on either side of the interface are reconstructed from the interpolation in the overlap region. On the fine side two rows of nodes are interpolated together. If the source cell is incomplete because it cuts into a boundary condition and some of the vertices are not inside the fluid domain, the method uses a nearest complete source cell and evaluates the interpolation function outside of that cell such that extrapolation towards the boundary is used (see Figure 1). In the asynchronous fine time step, no grid coupling is performed and the first line of fine nodes erodes away. Before the next coupling the first line of nodes on the coarse grid and the second line on the fine grid become invalid due to lack of nearest neighbors. In the following synchronization all invalid nodes are refilled by interpolation and the cycle begins again.

In order to have the same speed of sound on both sides of the interface, which is a physical condition for zero reflection at the interface [104, 105], we use acoustic scaling. This means the velocity is not scaled but the diffusion



(a) Interpolation cells (b) Grid interface

Figure 1: Interpolation cells and grid interface. The upper cell on the left side is the fine to coarse interpolation cell. The data from the nodes at the vertices are used to compute the data at the coarse node in the center. The lower cell is the coarse to fine interpolation cell. The data at the vertices of the outer cell is used to interpolate to the eight fine nodes inside the cell. The picture at the right shows the arrangement of the grid with the interpolation cells being highlighted. The grid allows for a small overlap between the fine and the coarse grids in order to avoid interpolation in time. Circles represent regular nodes while diamonds denote the nodes that are obtained from interpolation. If the interface cuts into a boundary condition not all vertices associated with the interpolation cell are available and the respective node (marked by triangles) have to be extrapolated from the nearest complete source cell as indicated by the arrows. This grid coupling method does not need to distinguish between different orientations of the wall and between planes, corners and edges. It is therefore well suited for arbitrary shape grid refinement which is obviously mandatory for porous media.

coefficients and hence the relaxation rates have to be scaled such that:

$$\tau_{1f} = 2\tau_{1c} \Leftrightarrow \left(\frac{1}{\omega_{1f}} - \frac{1}{2}\right) = 2\left(\frac{1}{\omega_{1c}} - \frac{1}{2}\right), \tag{104}$$

where indices f and c indicate the values for the fine and the coarse grid, respectively.

Applying the diffusive scaling with $\tau_{1f} = \tau_{1c}$ and $\Delta t_f = 4\Delta t_c$ is avoided in grid refinement as this would lead to a jump in the speed of sound such that the interface would acoustically behave like a solid boundary. Even if sound waves are not our primary interest here we note that they are always present. A reflective grid interface would act as a parasitic resonator. This is completely avoided by using acoustic scaling.

3.1. Evaluation of numerical viscosity at grid interfaces

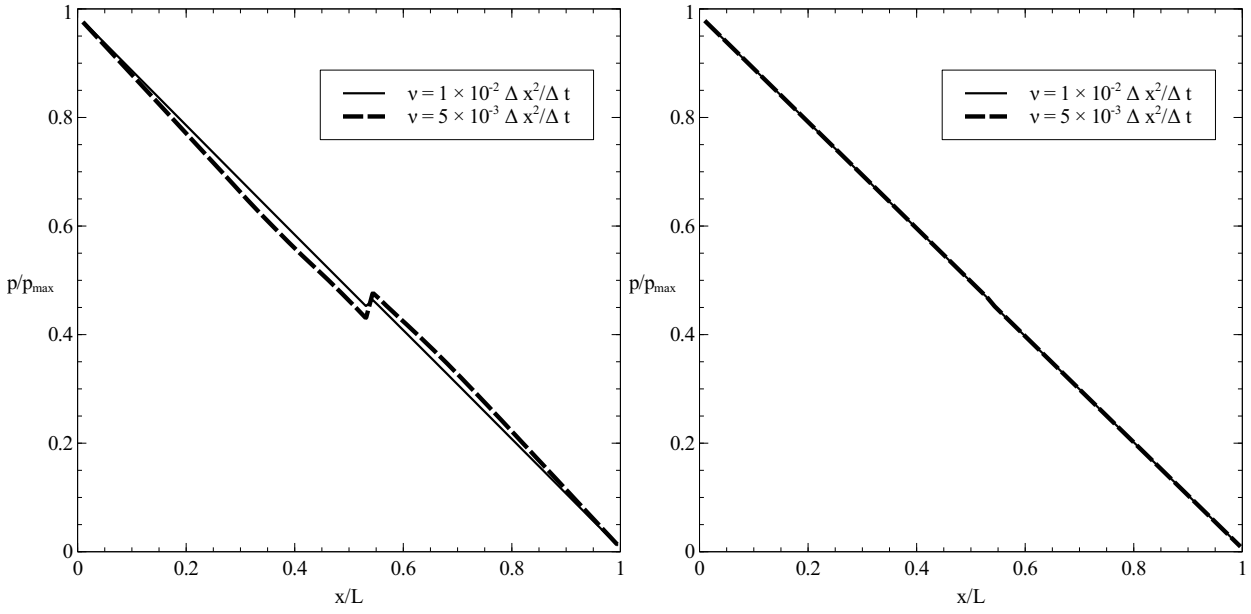
We emphasized that quadratic interpolation is necessary to avoid the introduction of numerical viscosity at the grid interface. To demonstrate this point we conduct a simple test with pressure driven flow through a channel as described in [76]. Dissipation P in a steady flow can be computed from the volume flux Φ_V and the pressure drop Δp over the volume:

$$P = \Phi_V \Delta p. \tag{105}$$

The dissipation in a fluid is a function of its viscosity and is constant along the length of a channel. If the effective viscosity suddenly changes we expect a discontinuity in dissipation, which in a channel flow with constant flow rate

implies a jump in the pressure gradient which is easily detected. Thus, a simple laminar channel flow is a minimalistic and suitable example to investigate the presence or absence of numerical viscosity at the grid interface.

Here we conduct two simulations, one using compact interpolation and one using linear interpolation for velocity. The setup consists of a channel of length $L = 40\Delta x_c$, width $L/2$ and thickness $2\Delta x$ with periodic boundary conditions in the third direction. The flow is driven by a pressure $p_{max} = 10^{-7}$ at the inlet and zero pressure at the outlet. Two different viscosities were used to demonstrate their influence on the accuracy of the interpolation. The grid is refined upstream from $x = L/2$. The pressure along the channel is depicted in Figure 2. It is seen that the linear interpolation results in a kink in pressure at the grid interface whereas the compact interpolation results in a smooth slope. We also note that the kink in Figure 2(a) increases in magnitude for smaller viscosities. This is because the ratio between numerical and physical viscosity increases with decreasing physical viscosity. The error introduced through linear interpolation is hence more significant for high Reynolds number flow. We thus demonstrated that our method of grid refinement based on compact quadratic interpolation is free of numerical viscosity as seen from the fact that no dissipation occurs at the interface and the slope is constant for both viscosities.



(a) linear interpolation

(b) compact (quadratic) interpolation

Figure 2: Comparison of the pressure drop along a channel for a grid that is refined at $L/2$. Plot (a) shows the case for linear interpolation of velocity. The slop of the pressure has a clear kink which is due to the spurious dissipation originating from the numerical viscosity at the grid interface. Plot (b) shows the result for compact quadratic interpolation where no numerical viscosity is observed.

4. Numerical investigations of permeability and Forchheimer coefficient of porous materials

The Darcy law [106] describes the relation between the pressure gradient $\frac{dp}{dx}$ and the velocity of an incompressible flow through a porous medium as a function of the dynamic viscosity (property of the fluid) μ and permeability (property of the pore space geometry) κ. For isotropic porous materials the model equation is given as:

$$-\frac{dp}{dx} = \frac{\mu}{\kappa} u_D. \tag{106}$$

The Darcy velocity u_D is computed as the product of the averaged pore velocity $\langle \bar{u} \rangle$ and the porosity $\phi = \frac{V_V}{V_T}$ or as the ratio of the flow rate Q and the cross section area A of the porous medium:

$$u_D = \langle \bar{u}_p \rangle \phi = \frac{Q}{A}. \tag{107}$$

The dimensionless characteristic number of flow through porous materials is the pore scale Reynolds number which is based on the pore size L_p:

$$Re = \frac{u_D L_p}{\nu}. \tag{108}$$

The Darcy law is valid only for $Re \ll 1$. Forchheimer [1, 107] extended the Darcy equation for higher pore scale Reynolds numbers by a non-linear additional term:

$$-\frac{dp}{dx} = \frac{\mu u_D}{\kappa} + \beta \rho u_D^2, \tag{109}$$

where ρ is the density of the fluid. If we use the dimensionless Ergun coefficient [106] $c_F = \beta \sqrt{\kappa}$ we find:

$$-\frac{dp}{dx} = \frac{\mu u_D}{\kappa} + \frac{c_F}{\sqrt{\kappa}} \rho u_D^2. \tag{110}$$

Below, the permeability and the Forchheimer coefficient of two materials are obtained by simulation: porous aluminum PA80-110[1] and sintered bronze SBP120[2]. The geometry is obtained by computer tomography (CT) (see Figure 3). The pore size of the porous materials is between 80 and 110 µm for porous aluminum and app. 120 µm for sinter bronze. The porosity is 46% and 36%, respectively. Both scans are discretized with the native resolution of the scan Δx =3.9 µm. Uniform Cartesian grids with about 2.5 billion grid points are used for the simulation. This translates to about 68 billion degrees of freedom. The flow is driven by a pressure difference between two opposing sides of the domain. All other sides are closed with no-slip boundary conditions. Simulations are conducted for different Reynolds numbers. Figure 4 shows the computed effective permeability and the fitted Forchheimer equation as a function of the pore scale Reynolds number. For sufficiently low Reynolds numbers the permeability is independent of the Reynolds number. Above $Re \sim 0.1$ the permeability starts to decrease with growing Reynolds number. The data is fitted to the Forchheimer equation in order to determine the Forchheimer coefficient β. Table 1 shows that the static air flow resistivity (r_{sim}) of porous aluminum and sintered bronze computed with the cumulant LBM is in essentially quantitative agreement with experimental data (r_{exp}). The resistivity has been determined experimentally at the Physikalisch-Technische Bundesanstalt Braunschweig using the alternating airflow method (ISO9053/DIN EN 29053) [38]. The conversion of the permeability κ to the flow resistivity r is accomplished by the following transformation

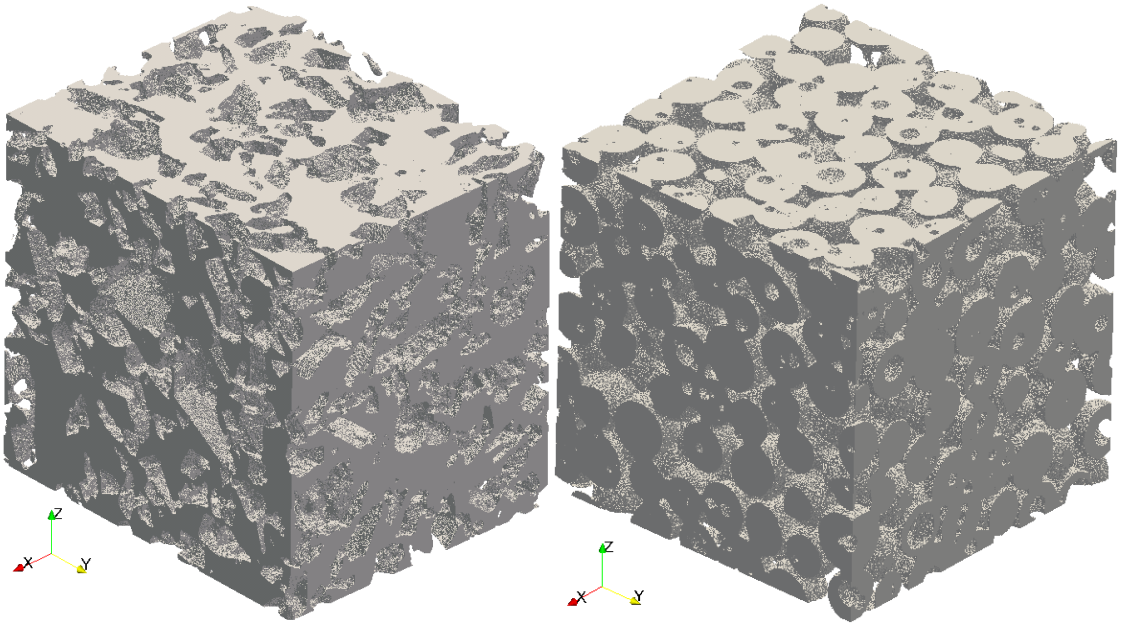
$$r = \frac{\mu_0}{\kappa} \quad \text{with} \quad \mu_0 = 18.24 \times 10^{-6} \frac{kg}{ms}, \tag{111}$$

where the value of the dynamic viscosity μ_0 corresponds to the viscosity of air at a temperature of 20 °C.

Simulation	Material	$\kappa \times 10^{-10} [m^2]$	$C_F[-]$	$r_{sim}[Ns/m^4]$	$r_{exp}[Ns/m^4]$	Deviation [%]
Case A	PA80-110	1.28675	5.0583	142,279	145,490	2.2
Case B	SBP120	2.81863	0.69117	64,712	64,686	-0.04

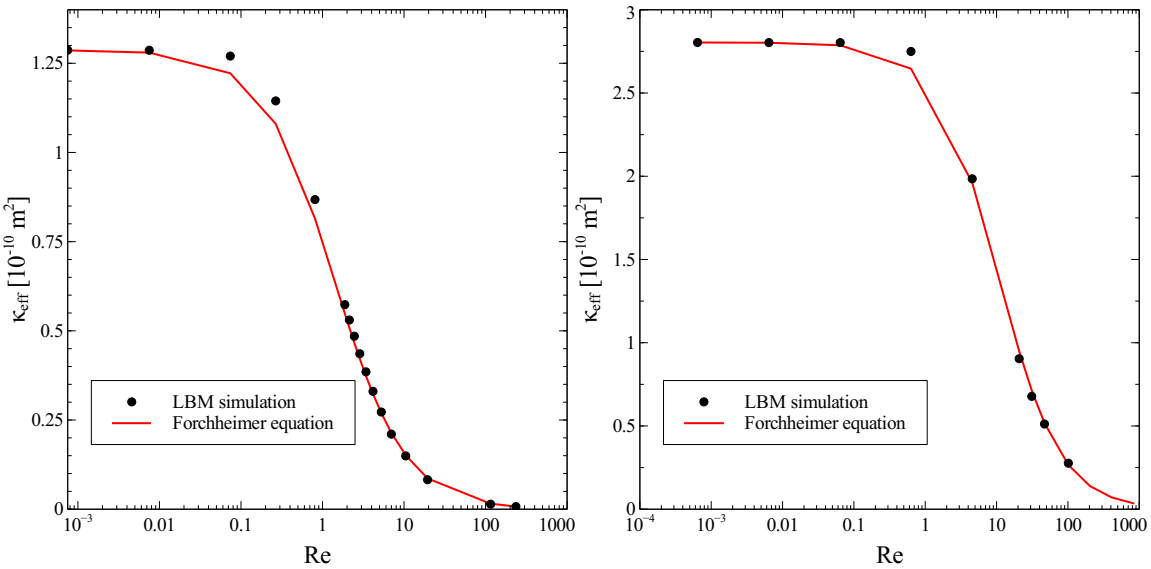
Table 1: Permeability, Forcheimer-coefficient and flow resistance of PA80-110 and SBP120 samples. Experiments are described in [38].

[1] produced by Exxentis AG, Schartenfelsstrasse 6, 5430 Wettingen, Switzerland
[2] produced by GKN Sinter Metals Filters GmbH, Dahlienstrasse 43, 42477 Radevormwald, Germany



(a) PA80-110 (b) SBP120

Figure 3: CT scans of porous aluminum PA80-110 (a) and sinter bronze SBP120 (b).



(a) PA80-110 (b) SBP120

Figure 4: Effective permeability of porous aluminum PA80-110 (a) and sintered bronze SBP120 (b) and fitted Darcy-Forchheimer equation as a function of Reynolds number.

	$\|\|u_x - u_x^{DNS}\|\|_{l2}$	$\|\|\sqrt{R_{11}^+} - \sqrt{R_{11}^{+DNS}}\|\|_{l2}$	$\|\|\sqrt{R_{33}^+} - \sqrt{R_{33}^{+DNS}}\|\|_{l2}$
LBM	0.0349	0.1479	0.1276
VRANS	0.0249	0.0678	0.164

Table 2: Deviation of the LBM and the VRANS results from [111] from the DNS results from [108] as depicted in Figure 7 computed by Equation (112). The VRANS model was designed to capture the Reynolds stresses of the considered kind of porous media accurately. It is observed that our method shows comparable deviations from both the DNS data and the VRANS model. We note that Reynolds stresses are statistical values that are susceptible to turbulent intermittency. Thus the accuracy Reynolds stresses obtained from DNS data is strongly depending on the details of the statistical procedure used which results in a finite and usually unknown uncertainty.

5. Channel flow above porous bed

In this section the turbulent boundary layer above a porous material is investigated. The first example considers an artificial porous medium and it is shown here since DNS reference data are available for comparison. The second example considers a realistic porous material obtained from a CT scan.

5.1. Turbulent channel with cube array

As a validation study for a turbulent boundary layer over a porous medium we computed a channel flow over an artificial porous material. Figure 5 shows the geometric setup of the simulation. The channel height H is used for the definition of the computational domain, which measures $3H \times 2H \times 2H$. The lower half of the channel is filled with an array of cubes representing a generic porous medium. The cube size and the distance between the cubes is $H/20$. The boundary conditions are periodic in flow direction. A forcing is used to drive the flow with the target Reynolds number $Re = 5500 = u_{mean} * H/\nu$, starting from an initial plug flow with constant velocity in the domain. No-slip boundary conditions are used at the wall. Three refinement levels are used. The resolution of the upper boundary layer is chosen to obtain a dimensionless wall distance of the first grid node of $z^+ \simeq 2$. The case is not physical as the cubes float while being stationary. The case is considered here for comparison to the DNS data from Breugem and Boersma [108] where this setup was chosen as a generic benchmark. Figure 6 shows the instantaneous flow using the Q-criterion [109, 110] colored by the normalized velocity magnitude together with a zoom area. It gives an idea of the complexity of the flow field at a resolution of about 90 million grid points and about 2.3 billion degrees of freedom. Figure 7 shows time averaged velocity profile and RMS values of the fluctuations in comparison with DNS data from Breugem and Boersma [108] and with result of the VRANS simulation [111, 112] (see also [40]). The streamwise mean velocity profile of the LBM simulation is in good agreement with DNS and VRANS. After reaching a dynamic equilibrium for the bulk flow the macroscopic quantities are volume averaged with the cellular filter from [113]. The volume averaged data is obtained every 2250 time steps (on the coarsest grid), which correspond to half an eddy turnover time (ETOT). The data is averaged in time over 50 of such intervals. We further compare the result of our simulation and those from the VRANS simulation to the DNS by calculating the $l2-$norm of quantity Φ:

$$\|\Phi - \Phi^{DNS}\|_{l2} = \sqrt{\frac{\int(\Phi - \Phi^{DNS})^2 dV}{\int(\Phi^{DNS})^2 dV}}. \tag{112}$$

The results for the velocity and two components of the Reynolds stress are shown in Table 2. The results of the LBM simulation show a small but notable deviation from the DNS as well as the VRANS simulations. We note that the VRANS model was designed to capture accurate Reynolds stresses for homogeneous porous media and was calibrated to exactly the DNS data from Breugem. The VRANS result can therefore be regarded as a gold standard in this context. We also note that the DNS data has an unknown uncertainty especially for the Reynolds stresses which are statistical quantities. Finally we note that flow over a porous medium might dependent strongly on the anisotropy of the pores [114].

5.2. Turbulent channel with a porous aluminum bed

After the successful validation with an artificial porous medium, a channel partially filled with porous aluminum PA80-110 is simulated (see Fig. 8). The length of the computational domain is 64 mm, the height is 17 mm and

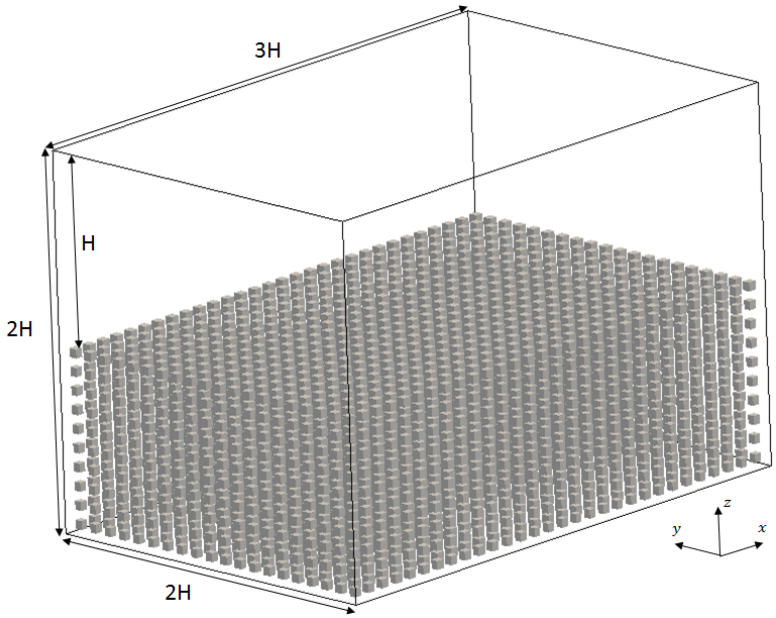


Figure 5: Geometric setup of channel with an array of cubes.

the depth is 8 mm. Periodic boundary conditions are used in streamwise and spanwise directions. No-slip boundary conditions are used in wall-normal directions. Adaptive forcing based on a proportional-integral-derivative (PID) controller is applied in order to achieve the bulk Reynolds number $Re = 51000$. The porous bed is constructed with 32 periodic copies of the porous aluminum PA80-110 scan, which has a dimension of 4 mm × 4 mm × 1 mm. The computational grid consist of approx. 1.2×10^9 grid nodes. The resolution of the grid close to the top and bottom walls and inside the porous aluminum is 10 µm corresponding to the normalized wall distance on the top of $z^+ \simeq 1$. The simulation reaches dynamic equilibrium after 50 ETOT. Additional 10 ETOT are used for collecting the turbulence statistics.

The mean velocity profile of a turbulent flow satisfies the law of the wall [115].

$$U^+ = \frac{1}{K} \ln z^+ + C^+. \tag{113}$$

In (113) U^+ is the streamwise mean velocity and z^+ is the wall distance normalized by the friction velocity u_τ. K is the von Kármán constant and C^+ is a constant. For smooth walls $K \approx 0.41$ and $C^+ \approx 5$ [106]. The friction velocity is defined as

$$u_\tau = \sqrt{\frac{\tau}{\rho}}, \tag{114}$$

where τ is the shear stress and ρ is the density of the fluid.

Figure 9 shows the streamwise mean velocity of the channel flow with smooth walls and with porous aluminum in comparison with DNS data for the smooth wall [116]. The friction Reynolds number of the DNS is $Re_\tau \approx 2000$ [116, 117] and is close to $Re_\tau \approx 2474$ on the smooth wall of the channel. The logarithmic law for flow over a completely rough wall is defined as [118]:

$$U^+ = \frac{1}{K} \ln\left(\frac{z^+}{z_0^+}\right), \tag{115}$$

where z_0^+ is the characteristic roughness length. Nikuradse's equivalent sand roughness k_s [118] describes the velocity

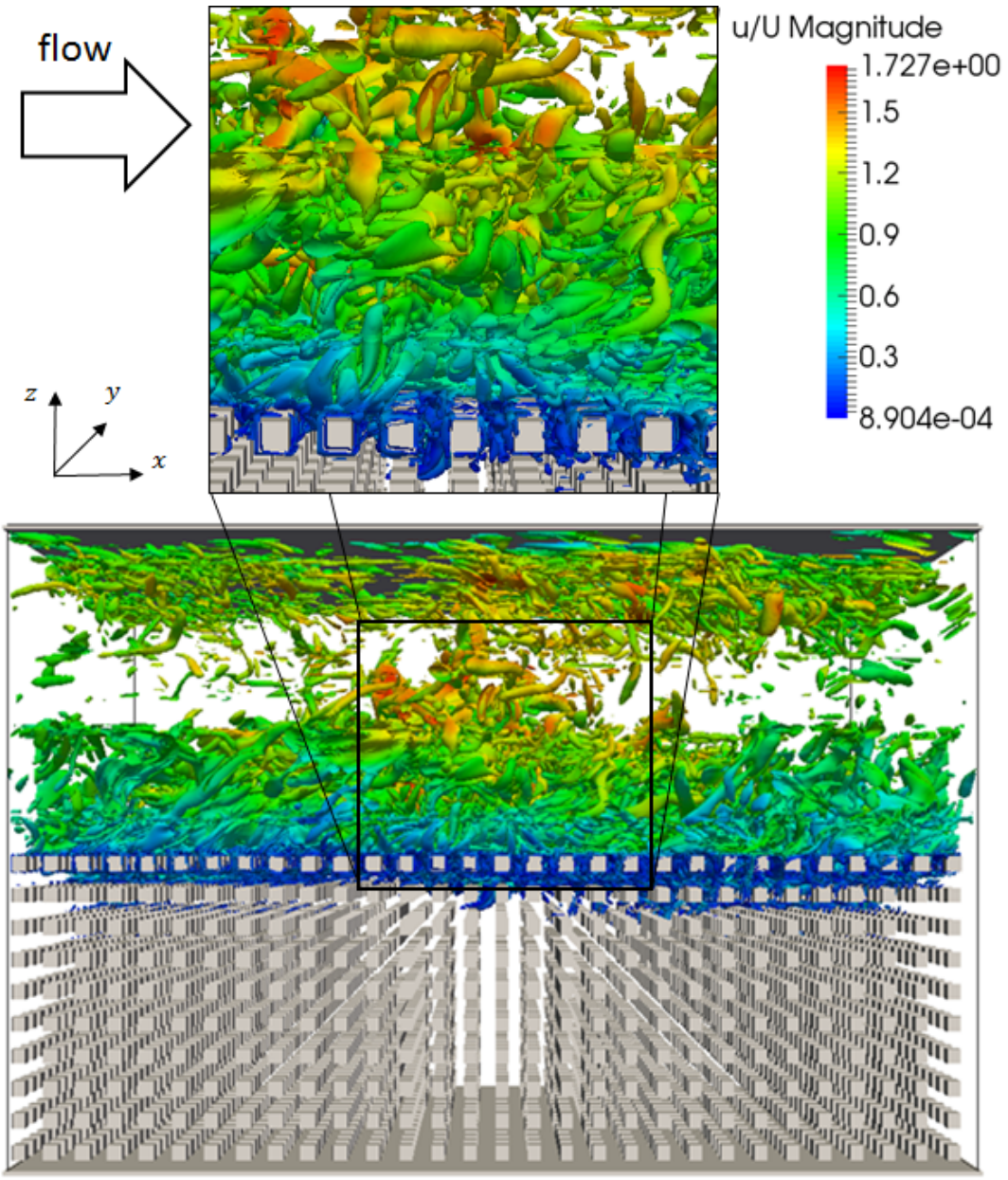


Figure 6: Iso-surfaces of Q-criterion [110] colored by the normalized velocity magnitude.

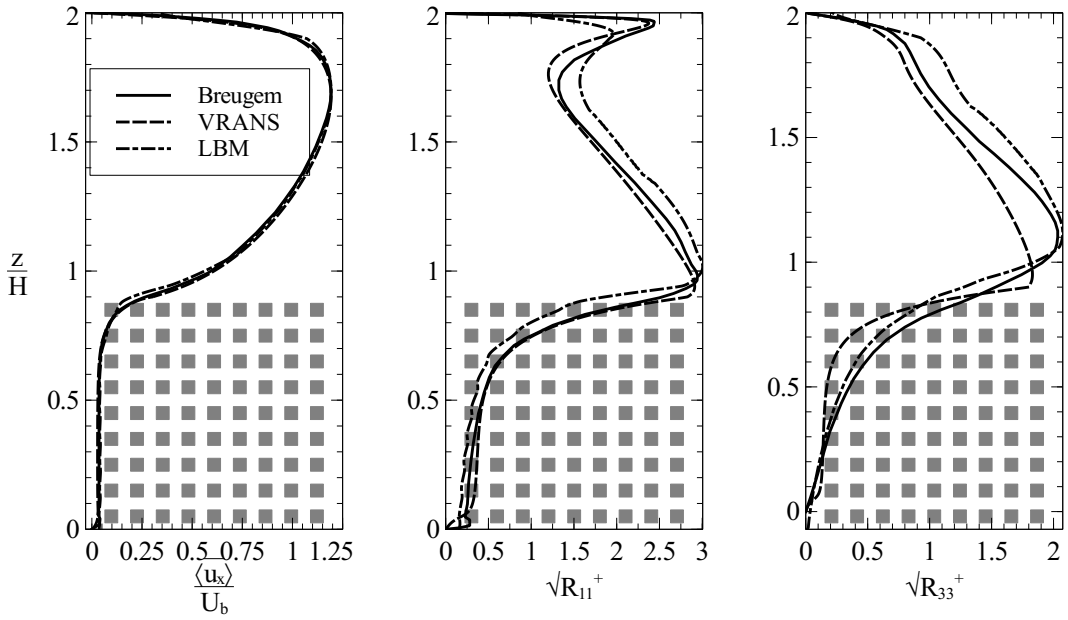


Figure 7: The normalized streamwise mean velocity and Reynolds stress in streamwise and wall-normal direction. We compare our results to the DNS results from Breugem [108] and the VRANS results from [111]. A quantitative comparison of the deviations of the LBM and VRANS results from the DNS data is presented in Table 2.

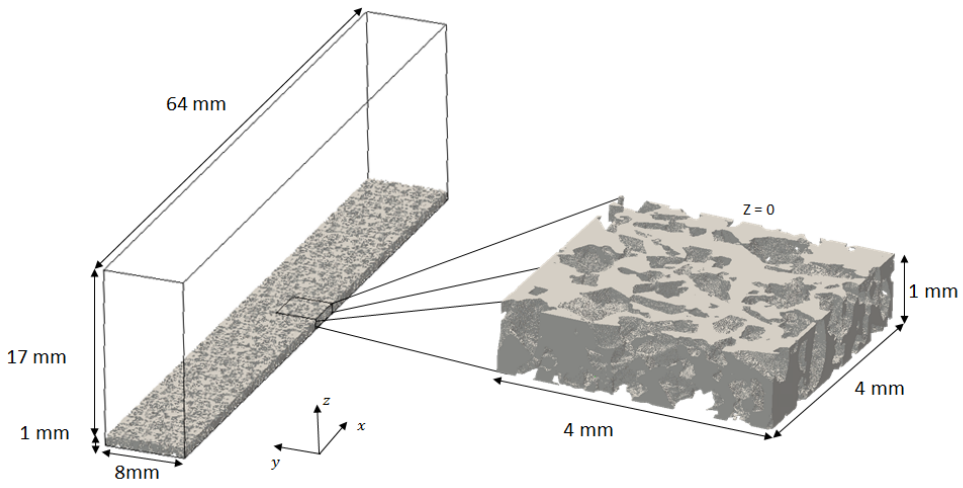


Figure 8: Geometric setup of the turbulent channel with porous aluminum.

distribution across the depth and is defined as a function of roughness length as

$$z_0 = \frac{30}{k_s}. \tag{116}$$

We obtain $z_0^+ = 26.9$ as a z-intercept of the linear fit of the logarithmic part of the mean velocity profile on a semi-logarithmic diagram. The roughness Reynolds number is $\frac{k_s u_\tau}{\nu} = 808$. Flows with a roughness Reynolds number larger than 70 are considered completely rough [106], i.e. viscous effects are assumed to be negligible on the scale of the roughness elements. The streamwise mean velocity profile close to the porous aluminum is shown in Figure 9.

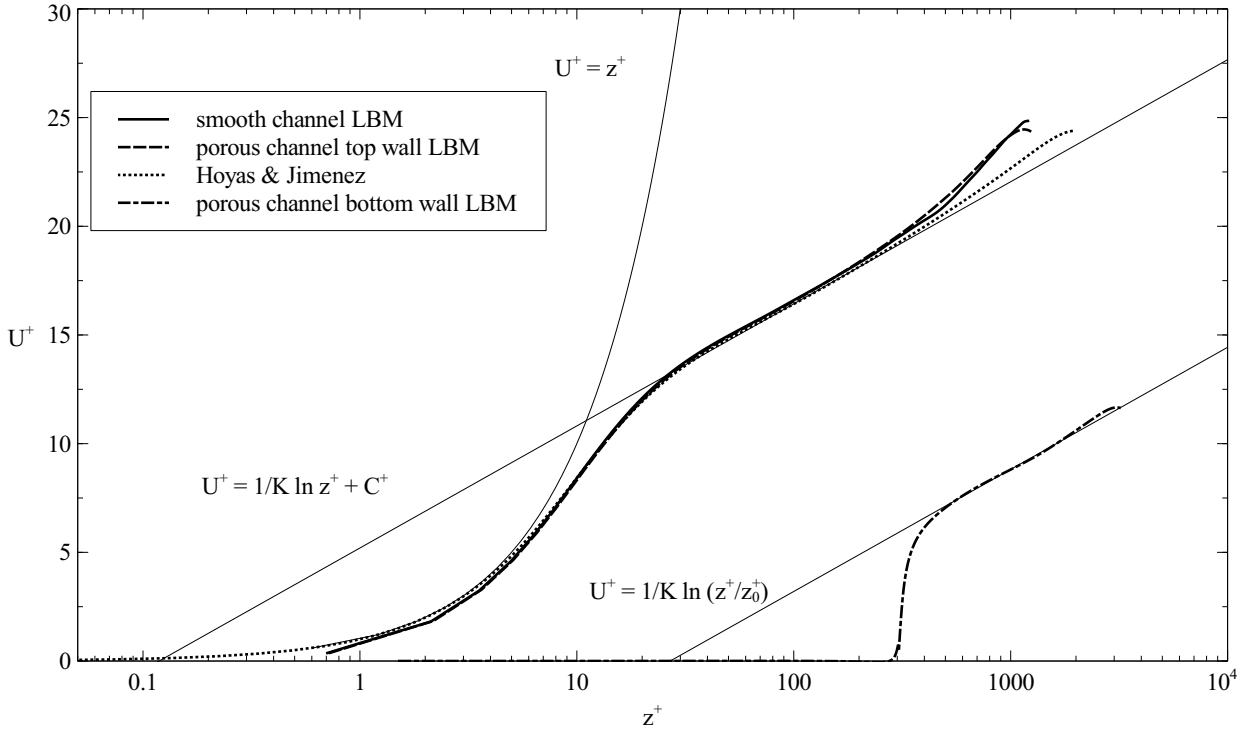


Figure 9: Streamwise mean velocity profile normalized by friction velocity.

Figure 10 shows the streamwise velocity profile normalized by the bulk velocity. In the zoom area we observe that the velocity in the upper part of the porous aluminum is negative, i.e. the flow turns counter-streamwise. Figure 11 shows the instantaneous velocity in the channel and inside the upper most pore of the porous aluminum. It is clearly seen that the flow shows the typical lid-driven cavity behavior. The back-flow represented by the lower part of the vortices in the upper most pores leads to this counter-streamiwse velocity. To the best of our knowledge, this statistically relevant property of the boundary layer inside the porous medium has not been reported before and may play an important role in the design of consistent homogenized boundary conditions at the interface between the porous medium and the airspace for RANS type flow models of flow in porous media.

The averaged Reynolds stress

$$R_{ij} = \frac{1}{A_{S_f}} \int_S \left(\frac{1}{T} \int_T (u_i' u_j') dt \right) dS, \tag{117}$$

where A_{S_f} is the area of the surface S occupied by fluid of the $x - y$ plane and u' is a turbulent fluctuation, is shown in Figure 12. The left picture shows that the magnitude of the Reynolds shear stress close to the rough wall is larger than close to the smooth wall. The increase of the stress corresponds to a higher friction factor $C_f = 2(u_\tau / U_b^2)$ [119] on the rough wall as compared to the smooth wall. The friction factor of the rough wall is found to be 2.088 times the friction factor of the smooth wall in this case. The right picture shows the Reynolds wall-normal stress of the flow close to the top wall in comparison to DNS data.

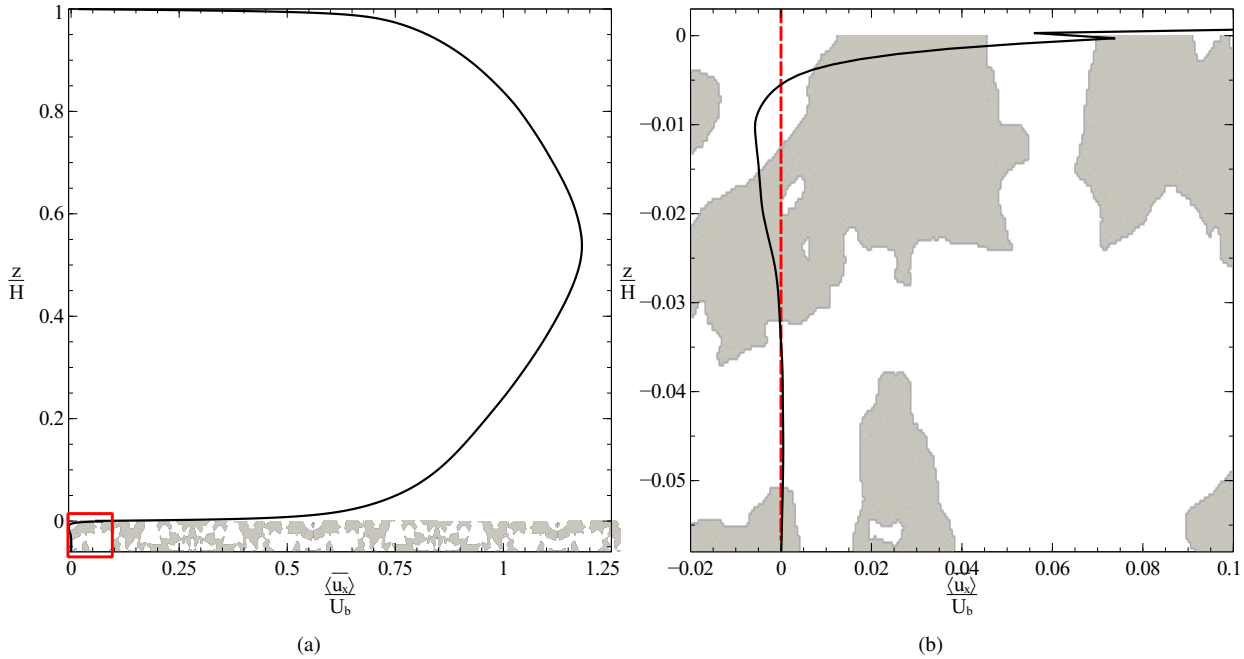


Figure 10: Streamwise mean velocity profile (a), the small rectangle on the left shows zoom area (b). The horizontal velocity component inside the pore close to the surface is observed to be smaller than zero, indicating that the flow moves in the opposite direction compared to the flow above the porous medium. The kink in the velocity profile at the surface of the porous medium (plot b) is due to the sudden increase of volume occupied by fluid. For averaging, the sum of the velocities of all nodes in each $x - y$ plane was divided by the number of nodes in the plane that were occupied by fluid.

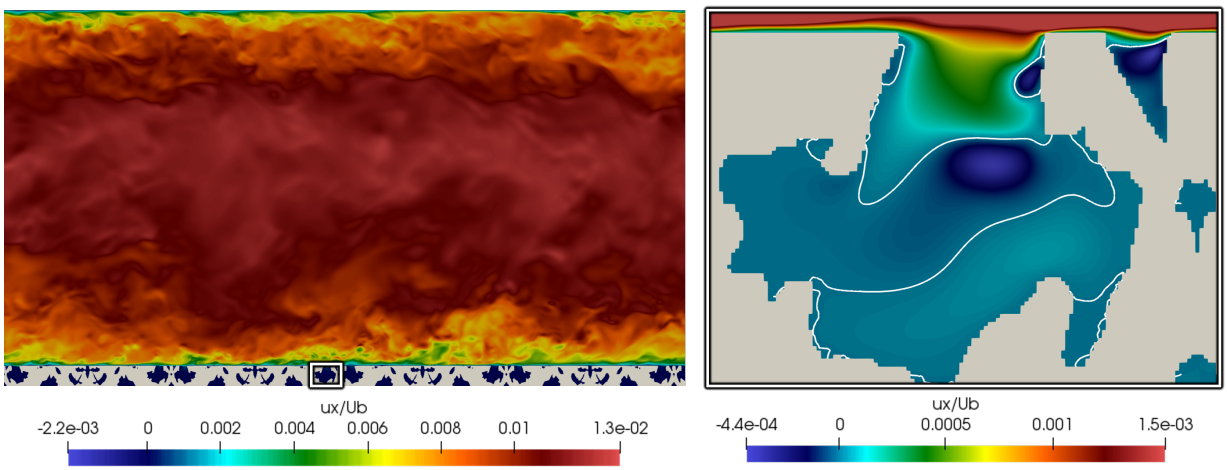


Figure 11: Lid-driven cavity effect in a pore close to the surface of the porous aluminum. The picture on the right shows a zoom into the pore indicated by the small box in the bottom of the picture on the left. The reason why the flow moves counter stream-wise in a certain depth is explained by the vortices formed in the upper most pores. In a certain depth, the vortex at the entrance of the pore recirculates causing the fluid to move in opposite direction to the main flow which is well known from driven cavity flows. The zero-velocity contour for u_x is shown inside the pore. Below the uppermost line the flow moves in opposite direction to the main flow. This effect is not captured by volume averaged RANS models which predict a monotonous decrease of velocity with increasing depth.

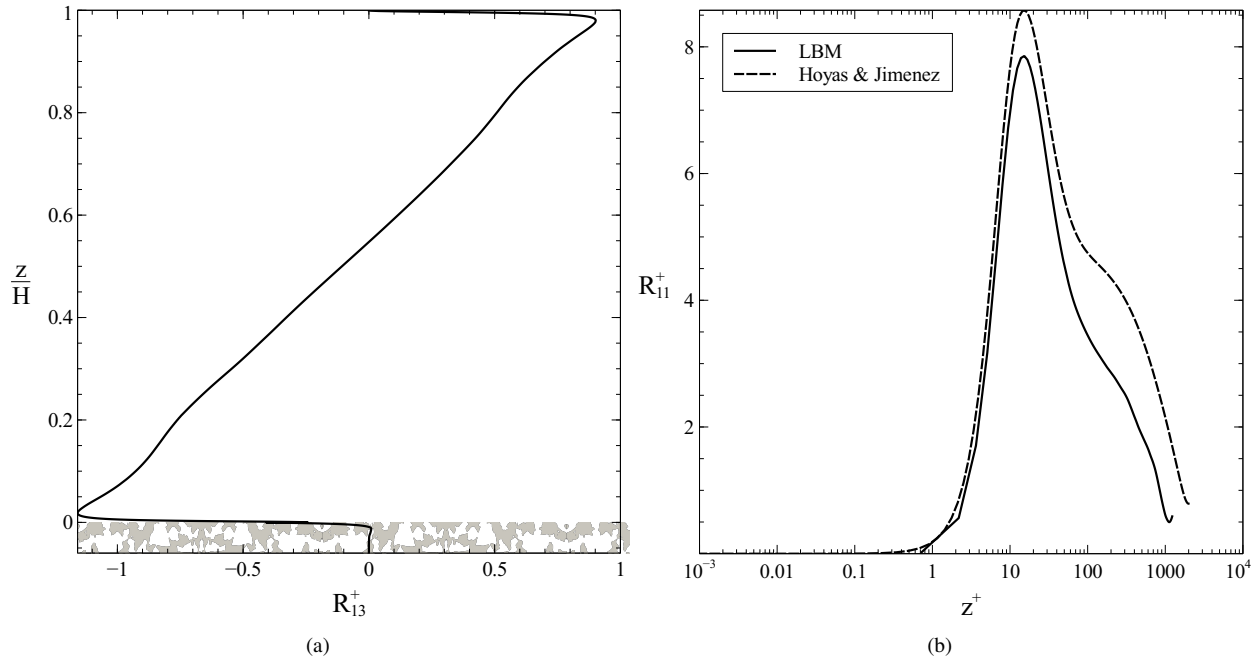


Figure 12: Normalized Reynolds stress profile with respect to channel high (a) and on the top wall (b).

6. Flow around a flat plate with porous inlay

The last example shows the interaction of the porous medium with an external flow in developed turbulence. For this flow around a flat plate with an additional porous inlay at the end is simulated. The simulation setup is close to the experimental one [39]. The experiment was carried out at the low noise wind tunnel of the Institute for Fluid Mechanics at TU Braunschweig (ISM). It is a small low speed wind tunnel with a cross section of 600 mm × 400 mm and an open recirculation region (Eiffel type) with a closed test section. Velocities are measured using Particle Image Velocimetry (PIV). For this case a flat plate of size (1000 mm × 40 mm × 10.5 mm) was positioned in the wind tunnel at an angle of attack of 0°. The Mach number of the flow was about $Ma = 0.05$, which leads to a Reynolds number of about $Re = 10^6$ based on the plates total length. A zigzag-tape was attached close to the leading edge of the plate to trigger turbulence. The porous inlay is positioned at the trailing edge of the plate and adjusted so that the upper surface of the inlay and plate are in the same plane. The metal foam MF450[3] is used as a porous material for the inlay. Some physical properties of the MF450 are presented in Table 3. Both the zigzag tape and the porous inlay are included in our simulation setup. From the PIV measurements mean velocity profiles as well as turbulent fluctuations of the streamwise and vertical component of the velocity are available at a distance of 935 mm from the front edge along a line normal to the surface of the plane. The turbulence intensity of the incoming velocity is very low ($Ti < 0.1\%$). In the numerical setup it was assumed to be zero.

Porosity	ϕ	0.91
Permeability	κ	2.0×10^9 m^2
Typical pore size	L_p	450 µm

Table 3: Physical properties of metalfoam MF450.

[3] produced by Alantum Europe GmbH, Raiffeisenallee 6, 82041 Oberhaching, Germany

For the simulation, a 3D non-uniform Cartesian grid with 6 levels of refinement and about 700 million grid points is used, which translates to about 19 billion degrees of freedom. The full length of the upper part of the plate is resolved with a grid spacing equal to 78 µm corresponding to $z^+ \simeq 2$. This results in $276{\times}10^6$ points in the finest grid layer used to resolve the boundary layer. A no-slip boundary condition is used for the plate and a velocity boundary condition is set at the inflow plane. A zero pressure boundary condition is used for the outflow and a slip boundary condition is used for the upper and lower domain boundaries. The accuracy of all boundary conditions is of second order. The simulation domain is 4 cm wide. At the sides where the domain is cut, periodic boundary conditions are used. For the digital reconstruction of the inlay geometry a CT scan of the metal foam MF450 is used. The sample has dimensions of 5 mm × 5 mm × 1.8 mm and consists of about 822 million voxels. In order to model the inlay the scan was reproduced 128 times and mirrored in order to avoid artifacts at the interfaces. The computational grid is refined at the porous inlay with grid spacing 39 µm corresponding to $z^+ \simeq 1$ and 178×10^6 points. The simulation domain with the porous inlay is sketched in Figure 13.

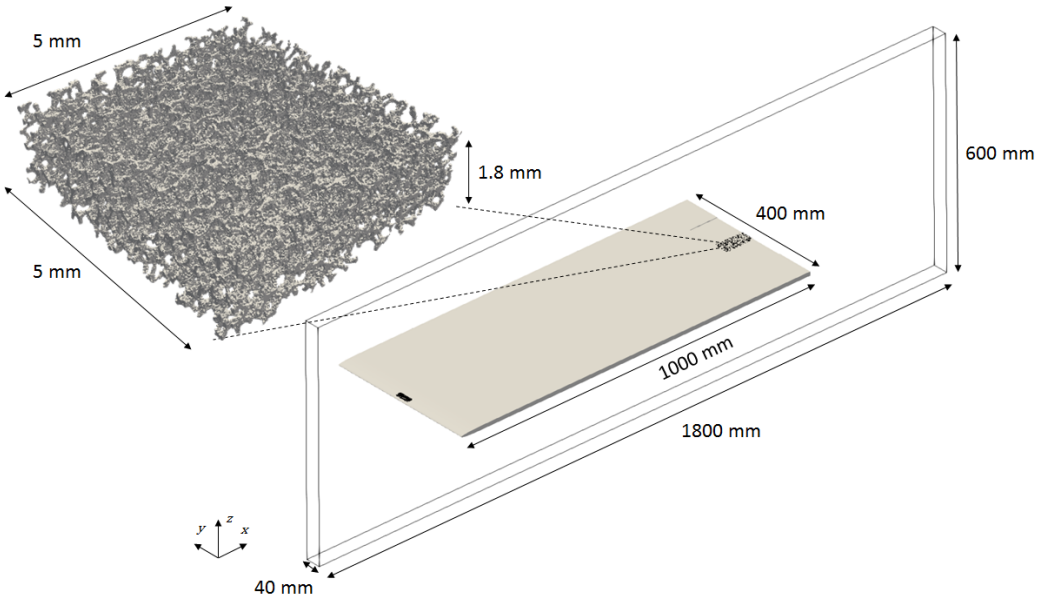


Figure 13: The geometric setup of the simulation of the flat plate with porous inlay.

The simulation uses $\sim 780,000$ core hours. For the six different levels of resolution the time step is refined proportional to the grid spacing. The simulation runs for 94,000 time steps on the coarsest level corresponding to 0.80 s real time. The solver records macroscopic quantities, time averaged velocities and fluctuations every 1000 coarse time steps. The averaged velocities and the Reynolds stresses above the porous medium are monitored. A dynamic steady state is reached after 61,000 time steps on the coarsest level. Figure 14 shows an isosurface of the instantaneous pressure field.

Figure 15 shows the normalized mean velocity profile and the Reynolds shear stress of the simulation of the plate with porous inlay in comparison with experimental data. The coordinate in wall-normal direction is normalized with the boundary layer thickness δ_{99} that is about 17 mm. It is observed that the Reynolds shear stress in our simulation is lower than the one measured by particle image velocimetry. This is to be expected since our method does not resolve all scales down to the Kolomgorov scale. It is known the implicit large eddy models underestimate the magnitude of total Reynolds shear stress [120] as the unresolved scales are essentially ignored.

7. Conclusion

In this work we investigated turbulent flows in porous media and over porous surfaces. The simulations based on the cumulant lattice Boltzmann model show good to excellent agreement with experimental and DNS data. The

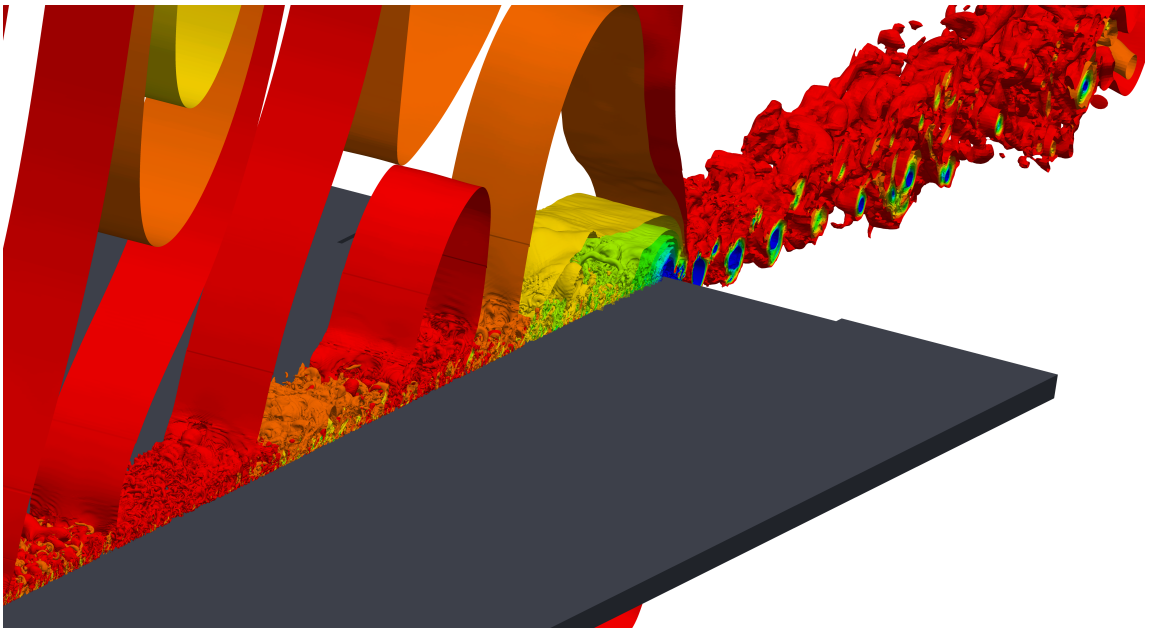


Figure 14: Iso-surfaces of the instantaneous pressure field.

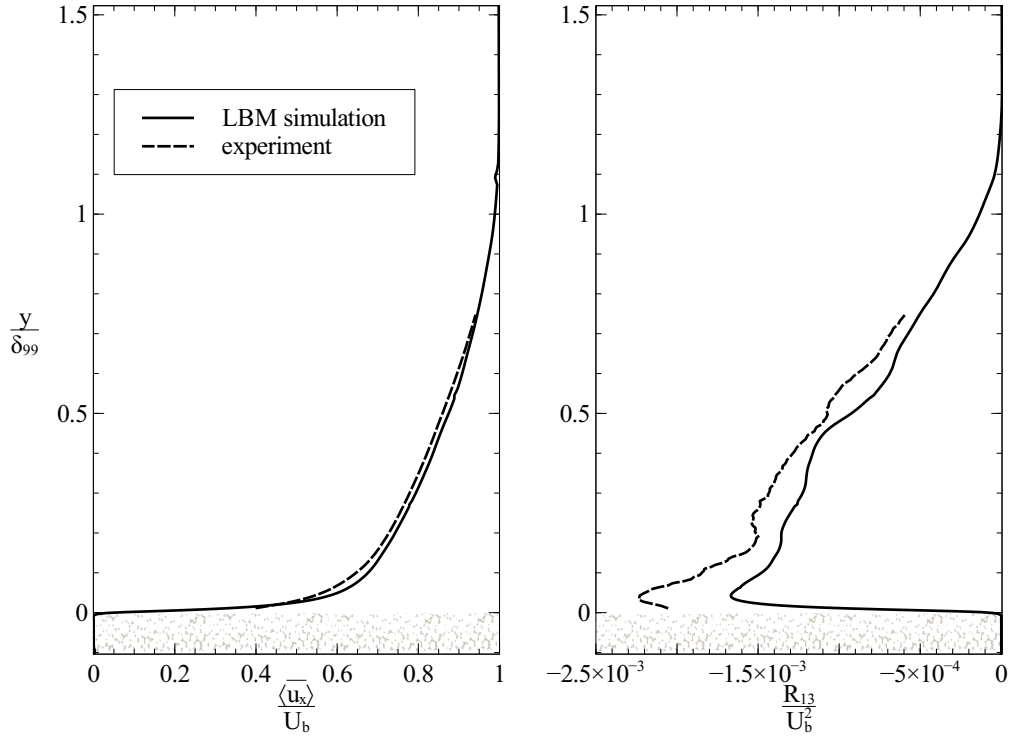


Figure 15: Comparison of normalized velocity profile and Reynolds shear stress between simulation and experiment for the flat plate with porous inlay. The shear stress shows a larger deviation from the reference than the velocity provile which is consistent with the results presented in Figure 7.

massively parallel VirtualFluids LBM framework with its sophisticated grid refinement treatment allows to conduct resolved simulations on the pore scale including the viscous sublayer. We conclude that the cumulant LBM is a suitable model to investigate turbulent flows without the need for an explicit turbulence model.

One of the results of our simulation was the identification of small recirculation in the upper most pores observed when turbulent flow passes over a porous medium. The immanent recirculation causes the averaged flow in a certain shallow depth inside the porous medium to flow in counter stream-wise direction. Current volume averaged RANS models do not capture this effect.

The computed permeability and Forchheimer-coefficient for porous aluminum and sintered bronze are in quantitative agreement with experimental data. The validation results obtained so far motivate the next step which is the aeroacustic analysis of a $1 : 10$ scale wind tunnel model of a wing extended by an anisotropic porous inlay at the downstream edge in order to analyze the detailed mechanism for the noise decrease observed in experiments. This will allow to determine optimal properties of the porous materials under consideration. According to initial setup tests these simulations will require app. 5×10^9 grid nodes resulting in about 1.3×10^{11} degrees of freedom.

Acknowledgements

This work was funded by the Deutsche Forschungsgemeinschaft DFG (Deutsche Forschungsgemeinschaft) in the framework of the collaborative research center SFB 880. Computational resources have been provided by the High Performance Computing Center Stuttgart (HLRS) and the North-German Supercomputing Alliance (HLRN).

References

[1] P. Forchheimer, Wasserbewegung durch boden [water movement through soil], Zeitschrift des Vereins Deutscher Ingenieure 45 (1901) 1782.
[2] J. Kozeny, Uber kapillare leitung des wassers in boden, Wiss. Wien Math. Naturwiss. KI. 2 (1927) 271.
[3] H. C. Brinkman, A calculation of the viscous force exerted by a flowing fluid on a dense swarm of particles, Appli. Sci. Res. A1 (1947) 27–34.
[4] S. Ergun, Fluid flow through packed columns, Chemical Engineering. Progress 48 (1952) 89.
[5] F. Engelund, On the laminar and turbulent flows of ground water through homogeneous sand, Akademiet for de tekniske Videnskaber 3 (1953) 1.
[6] J. C. Ward, Turbulent flow in porous media, Journal of Hydraulic Division, ASCE 90 (1964) 110.
[7] N. Ahmed, D. K. Sunada, Nonlinear flow in porous media, Journal of Hydraulic Division, ASCE 95 (1969) 1.
[8] S. M. Hassanizadeh, W. G. Gray, High velocity flow in porous media, Transport in Porous Media 2 (1987) 521–531.
[9] D. W. Barr, Turbulent flow through porous media, Ground Water 39 (5) (2001) 646–650.
[10] E. Skjetne, J.-L. Auriault, High-velocity laminar and turbulent flow in porous media, Transport in Porous Media 36 (2) (1999) 131–147. doi:10.1023/A:1006582211517.
URL http://dx.doi.org/10.1023/A:1006582211517
[11] N. S. Cheng, Z. Y. Hao, S. K. Tan, Comparison of quadratic and power law for nonlinear flow through porous media, Experimental Thermal and Fluid Science 32 (2008) 1538.
[12] F. Kuwahara, Y. Kameyama, S. Yamashita, A. Nakayama, Numerical modeling of turbulent flow in porous media using a spatially periodic array, Journal of Porous Media 1 (1998) 47.
[13] S. Whitaker, The forchheimer equation: A theoretical development, Transport in Porous Media 25 (1996) 27.
[14] H. Wang, E. S. Takle, Boundary-layer flow and turbulence near porous obstacles, Boundary-Layer Meteorology 74 (1995) 73.
[15] V. S. Travkin, I. Catton, L. Gratton, Single-phase turbulent transport in prescribed non-isotropic and stochastic porous media, Heat Transfer in Porous Media, ASME-HTD-240 (1993) 43.
[16] Y. Takatsu, T. Masuoka, Turbulent phenomena in flow through porous media, Journal of Porous Media 3 (1998) 243.
[17] A. Nakayama, F. Kuwahara, A macroscopic turbulence model for flow in a porous medium, Journal of Fluids Engineering 121 (1999) 427.
[18] T. Masuoka, Y. Takatsu, Turbulence model for flow through porous media, International Journal of Heat and Mass Transfer 39 (1996) 2803.
[19] M. J. d. Lemos (Ed.), Turbulence in Porous Media, Elsevier Science Ltd, Oxford, 2006. doi:http://dx.doi.org/10.1016/B978-008044491-8/50000-X.
URL http://www.sciencedirect.com/science/article/pii/B978008044491850000X
[20] F. E. Teruel, Rizwan-Uddin, Characterization of a porous medium employing numerical tools: Permeability and pressure-drop from darcy to turbulence, International Journal of Heat and Mass Transfer 52 (2526) (2009) 5878 – 5888. doi:http://dx.doi.org/10.1016/j.ijheatmasstransfer.2009.07.017.
URL http://www.sciencedirect.com/science/article/pii/S0017931009004505
[21] F. Ferdos, B. Dargahi, A study of turbulent flow in large-scale porous media at high reynolds numbers. part i: numerical validation, Journal of Hydraulic Research 54 (6) (2016) 663–677. arXiv:http://dx.doi.org/10.1080/00221686.2016.1211184, doi:10.1080/00221686.2016.1211184.
URL http://dx.doi.org/10.1080/00221686.2016.1211184

[22] F. Ferdos, B. Dargahi, A study of turbulent flow in large-scale porous media at high reynolds numbers. part ii: flow physics, Journal of Hydraulic Research 54 (6) (2016) 678–691. arXiv:http://dx.doi.org/10.1080/00221686.2016.1211185, doi:10.1080/00221686.2016.1211185.
URL http://dx.doi.org/10.1080/00221686.2016.1211185

[23] C. C. Edward, L. Fue-Sang, Permeability effects of turbulent flow through a porous insert in a backward-facing-step channel, Transport in Porous Media 59 (2005) 47–71.

[24] P. Kundu, V. Kumar, I. Mishra, Numerical modeling of turbulent flow through isotropic porous media, International Journal of Heat and Mass Transfer 75 (2014) 40 – 57. doi:http://dx.doi.org/10.1016/j.ijheatmasstransfer.2014.03.020.
URL http://www.sciencedirect.com/science/article/pii/S0017931014002269

[25] B. V. Antohe, J. L. Lage, A general two-equation macroscopic turbulence model for incompressible flow in porous media, International Journal of Heat and Mass Transfer 40 (1997) 3013.

[26] D. Getachew, W. J. Minkowycz, J. L. Lage, A modified form of the model for turbulent flows of an incompressible fluid in porous media, International Journal of Heat and Mass Transfer 43 (2000) 2909.

[27] R. B. Kazeroonia, S. K. Hannania, Simulation of turbulent flow through porous media employing a v2f model, AIP Conference Proceedings 963. doi:http://dx.doi.org/10.1063/1.2835977.

[28] E. Fattahi, C. Waluga, B. Wohlmuth, U. Rüde, M. Manhart, R. Helmig, Lattice boltzmann methods in porous media simulations: From laminar to turbulent flow, Computers & Fluids 140 (2016) 247 – 259. doi:http://dx.doi.org/10.1016/j.compfluid.2016.10.007.
URL http://www.sciencedirect.com/science/article/pii/S0045793016303061

[29] D. Trebotich, G. H. Miller, Simulation of flow and transport at the micro (pore) scale, Proceedings of the 2nd International Conference on Porous Media and its Applications in Science and Engineering.

[30] M. Piller, D. Casagrande, G. Schena, M. Santini, Pore-scale simulation of laminar flow through porous media, Journal of Physics: Conference Series 501. doi:10.1088/1742-6596/501/1/012010.

[31] T. Inamuro, M. Yoshino, F. Ogino, Lattice boltzmann simulation of flows in a three-dimensional porous structure, International Journal for Numerical Methods in Fluids 29 (7) (1999) 737–748. doi:10.1002/(SICI)1097-0363(19990415)29:7<737::AID-FLD813>3.0.CO;2-H.
URL http://dx.doi.org/10.1002/(SICI)1097-0363(19990415)29:7<737::AID-FLD813>3.0.CO;2-H

[32] M. Hasert, J. Bernsdorf, S. Roller, Lattice boltzmann simulation of non-darcy flow in porous media, Procedia Computer Science 4 (2011) 1048 – 1057. doi:http://dx.doi.org/10.1016/j.procs.2011.04.111.
URL http://www.sciencedirect.com/science/article/pii/S1877050911001694

[33] N. Manjhi, N. Verma, K. Salem, D. Mewes, Simulation of 3d velocity and concentration profiles in a packed bed adsorber by lattice boltzmann methods, Chemical Engineering Science 61 (2006) 7754.

[34] Z. Tang, S. Tang, Y. Dong, Numerical analysis of porous wall-bounded turbulent flows based on the lattice boltzmann method, Proceedings of the International Conference on Multiphase Flow and Heat Transferdoi:10.11159/icmfht16.108.

[35] B. Ahrenholz, J. Tlke, M. Krafczyk, Lattice-boltzmann simulations in reconstructed parametrized porous media, International Journal of Computational Fluid Dynamics 20 (6) (2006) 369–377. arXiv:http://dx.doi.org/10.1080/10618560601024694, doi:10.1080/10618560601024694.
URL http://dx.doi.org/10.1080/10618560601024694

[36] B. Ahrenholz, J. Niessner, R. Helmig, M. Krafczyk, Pore-scale determination of parameters for macroscale modeling of evaporation processes in porous media, Water Resources Research 47 (7), w07543. doi:10.1029/2010WR009519.
URL http://dx.doi.org/10.1029/2010WR009519

[37] X. Yang, Y. Mehmani, W. A. Perkins, A. Pasquali, M. Schönherr, K. Kim, M. Perego, M. L. Parks, N. Trask, M. T. Balhoff, et al., Intercomparison of 3d pore-scale flow and solute transport simulation methods, Advances in Water Resources 95 (2016) 176–189.

[38] M. Herr, K.-S. Rossignol, J. Delfs, N. Lippitz, M. Mößner, Specification of porous materials for low-noise trailing-edge applications, in: 20th AIAA/CEAS Aeroacoustics Conference, 2014, p. 3041.

[39] P. Kumar, K. Kutscher, M. Mößner, R. Radespiel, M. Krafczyk, M. Geier, Validation of a vrans-model for turbulent flow over a porous flat plate by cumulant-lattice-boltzmann dns/les and experiments, Journal of Porous MediaAccepted for publication.

[40] M. Mößner, R. Radespiel, Modelling of turbulent flow over porous media using a volume averaging approach and a reynolds stress model, Computers & Fluids 108 (2015) 25–42.

[41] S. Uphoff, M. Krafczyk, K. Rurkowska, S.Langer, N. Lippitz, B. Famann, A hierarchical approach to determining acoustic absorption properties of porous media combining pore-resolved and macroscopic models, accepted for publication in Journal of Porous Media.

[42] M. Geier, M. Schönherr, A. Pasquali, M. Krafczyk, The cumulant lattice boltzmann equation in three dimensions: Theory and validation, Computers & Mathematics with Applications 70 (4) (2015) 507 – 547. doi:http://dx.doi.org/10.1016/j.camwa.2015.05.001.
URL http://www.sciencedirect.com/science/article/pii/S0898122115002126

[43] M. Krafczyk, K. Kucher, Y. Wang, M. Geier, Dns/les studies of turbulent flows based on the cumulant lattice boltzmann approach, in: High Performance Computing in Science and Engineering 14, Springer, 2015, pp. 519–531.

[44] A. Pasquali, M. Schnherr, M. Geier, M. Krafczyk, Simulation of external aerodynamics of the drivaer model with the lbm on gpgpus, Parallel Computing: On the Road to Exascale, Advances in Parallel Computing 27 (2015) 391–400.

[45] E. Kian Far, M. Geier, K. Kutscher, M. Krafczyk, Distributed cumulant lattice boltzmann simulation of the dispersion process of ceramic agglomerates, Journal of Computational Methods in Sciences and Engineering 16 (2) (2016) 231–252.

[46] E. Kian Far, M. Geier, K. Kutscher, M. Krafczyk, Simulation of micro aggregate breakage in turbulent flows by the cumulant lattice boltzmann method, Computers & Fluids 140 (2016) 222–231.

[47] E. Kian Far, M. Geier, K. Kutscher, M. Krafczyk, Implicit large eddy simulation of flow in a micro-orifice with the cumulant lattice boltzmann method, Computation 5 (2) (2017) 23.

[48] M. Geier, A. Pasquali, M. Schnherr, Parametrization of the cumulant lattice boltzmann method for fourth order accurate diffusion part i:

Derivation and validation, Journal of Computational Physicsdoi:http://dx.doi.org/10.1016/j.jcp.2017.05.040.
URL http://www.sciencedirect.com/science/article/pii/S0021999117304230
[49] M. Geier, A. Pasquali, M. Schnherr, Parametrization of the cumulant lattice boltzmann method for fourth order accurate diffusion part ii: Application to flow around a sphere at drag crisis, Journal of Computational Physicsdoi:http://dx.doi.org/10.1016/j.jcp.2017.07.004.
URL http://www.sciencedirect.com/science/article/pii/S0021999117305065
[50] D. d'Humières, Generalized lattice boltzmann equations, Rarefied Gas Dynamics: Theory and Simulations 159 (1992) 450–458.
[51] P. Lallemand, L.-S. Luo, Theory of the lattice Boltzmann method: dispersion, dissipation, isotropy, Galilean invariance, and stability, Physical review. E, Statistical physics, plasmas, fluids, and related interdisciplinary topics 61 (6 Pt A) (2000) 6546–62.
[52] D. d'Humières, I. Ginzburg, M. Krafczyk, P. Lallemand, L.-S. Luo, Multiple-relaxation-time lattice Boltzmann models in three dimensions, Phil. Trans. R. Soc. A 360 (2002) 437–451.
[53] P. Asinari, Multiple-relaxation-time lattice Boltzmann scheme for homogeneous mixture flows with external force, Phys. Rev. E 77 (2008) 056706. doi:10.1103/PhysRevE.77.056706.
URL http://link.aps.org/doi/10.1103/PhysRevE.77.056706
[54] M. Geier, A. Greiner, J. G. Korvink, Cascaded digital lattice Boltzmann automata for high reynolds number flow, Phys. Rev. E 73 (2006) 066705. doi:10.1103/PhysRevE.73.066705.
[55] P. Asinari, Generalized local equilibrium in the cascaded lattice Boltzmann method, Phys. Rev. E 78 (2008) 016701. doi:10.1103/PhysRevE.78.016701.
URL http://link.aps.org/doi/10.1103/PhysRevE.78.016701
[56] M. Geier, De-aliasing and stabilization formalism of the cascaded lattice boltzmann automaton for under-resolved high reynolds number flow, INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS 56 (8) (2007) 1249–1254. doi:10.1002/fld.1634.
[57] M. GEIER, A. GREINER, J. G. KORVINK, Properties of the cascaded lattice boltzmann automaton, International Journal of Modern Physics C 18 (04) (2007) 455–462. arXiv:http://www.worldscientific.com/doi/pdf/10.1142/S0129183107010681, doi:10.1142/S0129183107010681.
URL http://www.worldscientific.com/doi/abs/10.1142/S0129183107010681
[58] Y. Ning, K. N. Premnath, D. V. Patil, Numerical study of the properties of the central moment lattice boltzmann method, International Journal for Numerical Methods in Fluids.
[59] A. De Rosis, E. Lévêque, Central-moment lattice boltzmann schemes with fixed and moving immersed boundaries, Computers & Mathematics with Applications 72 (6) (2016) 1616–1628.
[60] M. Geier, A. Greiner, J. G. Korvink, A factorized central moment lattice Boltzmann method, The European Physical Journal Special Topics 171 (1) (2009) 55–61. doi:10.1140/epjst/e2009-01011-1.
[61] K. N. Premnath, S. Banerjee, Incorporating forcing terms in cascaded lattice boltzmann approach by method of central moments, Physical Review E 80 (3) (2009) 036702.
[62] K. N. Premnath, S. Banerjee, On the three-dimensional central moment lattice boltzmann method, Journal of Statistical Physics 143 (4) (2011) 747–794.
[63] S. Geller, S. Uphoff, M. Krafczyk, Turbulent jet computations based on MRT and cascaded lattice Boltzmann models, Computers and Mathematics with Applications 65 (12) (2013) 1956–1966.
[64] D. Lycett-Brown, K. H. Luo, Multiphase cascaded lattice boltzmann method, Computers & Mathematics with Applications 67 (2) (2014) 350 – 362, mesoscopic Methods for Engineering and Science (Proceedings of ICMMES-2012, Taipei, Taiwan, 2327 July 2012). doi:http://dx.doi.org/10.1016/j.camwa.2013.08.033.
URL http://www.sciencedirect.com/science/article/pii/S0898122113005403
[65] D. Lycett-Brown, K. H. Luo, R. Liu, P. Lv, Binary droplet collision simulations by a multiphase cascaded lattice boltzmann method, Physics of Fluids 26 (2) (2014) 023303.
[66] S. Leclaire, N. Pellerin, M. Reggio, J.-Y. Trépanier, Multiphase flow modeling of spinodal decomposition based on the cascaded lattice boltzmann method, Physica A: Statistical Mechanics and its Applications 406 (2014) 307–319.
[67] L. Fei, K. Luo, Thermal cascaded lattice boltzmann method, arXiv preprint arXiv:1610.07114.
[68] F. Hajabdollahi, K. N. Premnath, Improving the low mach number steady state convergence of the cascaded lattice boltzmann method by preconditioning, Computers & Mathematics with Applications.
[69] R. A. Fisher, The derivation of the pattern formulae of two-way partitions from those of simpler patterns, Proceedings of the London Mathematical Society 33 (1931) 195 – 208.
[70] X. He, L.-S. Luo, Lattice boltzmann model for the incompressible navier–stokes equation, Journal of statistical Physics 88 (3) (1997) 927–944.
[71] M. Geier, A. Pasquali, Fourth order galilean invariance for the lattice boltzmann method, submitted to Computers and Fluids (2017).
[72] C. Peng, Z. Guo, L.-P. Wang, Lattice boltzmann model capable of mesoscopic vorticity computation, Physical Review E 96 (5) (2017) 053304.
[73] W. H. Miller, A treatise on crystallography, For J. & JJ Deighton, 1839.
[74] O. Filippova, D. Hänel, Grid refinement for lattice-bgk models, Journal of computational Physics 147 (1) (1998) 219–228.
[75] O. Filippova, D. Hänel, Boundary-fitting and local grid refinement for lattice-bgk models, International Journal of Modern Physics C 9 (08) (1998) 1271–1279.
[76] D. Lagrava, O. Malaspinas, J. Latt, B. Chopard, Advances in multi-domain lattice boltzmann grid refinement, Journal of Computational Physics 231 (14) (2012) 4808–4822.
[77] P. Valero-Lara, J. Jansson, A non-uniform staggered cartesian grid approach for lattice-boltzmann method, Procedia Computer Science 51 (2015) 296–305.
[78] Y. Kuwata, K. Suga, Imbalance-correction grid-refinement method for lattice boltzmann flow simulations, Journal of Computational Physics 311 (2016) 348–362.

[79] F. Schornbaum, U. Rude, Massively parallel algorithms for the lattice boltzmann method on nonuniform grids, SIAM Journal on Scientific Computing 38 (2) (2016) C96–C126.
[80] X. Shi, J. Lin, Z. Yu, Discontinuous galerkin spectral element lattice boltzmann method on triangular element, International Journal for Numerical Methods in Fluids 42 (11) (2003) 1249–1261.
[81] Y. Li, E. J. LeBoeuf, P. Basu, Least-squares finite-element lattice boltzmann method, Physical Review E 69 (6) (2004) 065701.
[82] M. Mirzaei, A. Poozesh, Simulation of fluid flow in a body-fitted grid system using the lattice boltzmann method, Phys. Rev. E 87 (2013) 063312. doi:10.1103/PhysRevE.87.063312.
URL http://link.aps.org/doi/10.1103/PhysRevE.87.063312
[83] K. Qu, C. Shu, Y. T. Chew, An isoparametric transformation-based interpolation-supplemented lattice boltzmann method and its application, Modern Physics Letters B 24 (13) (2010) 1315–1318.
[84] D. V. Patil, K. Lakshmisha, Finite volume tvd formulation of lattice boltzmann simulation on unstructured mesh, Journal of Computational Physics 228 (14) (2009) 5262–5279.
[85] T. Hübner, S. Turek, Efficient monolithic simulation techniques for the stationary lattice boltzmann equation on general meshes, Computing and Visualization in Science 13 (3) (2010) 129–143. doi:10.1007/s00791-009-0132-6.
URL http://dx.doi.org/10.1007/s00791-009-0132-6
[86] A. Zadehgol, M. Ashrafizaadeh, S. Musavi, A nodal discontinuous galerkin lattice boltzmann method for fluid flow problems, Computers & Fluids 105 (2014) 58 – 65. doi:http://dx.doi.org/10.1016/j.compfluid.2014.09.015.
URL http://www.sciencedirect.com/science/article/pii/S0045793014003521
[87] S. S. Patel, M. Min, K. C. Uga, T. Lee, A spectral-element discontinuous galerkin lattice boltzmann method for simulating natural convection heat transfer in a horizontal concentric annulus, Computers & Fluids 95 (2014) 197 – 209. doi:http://dx.doi.org/10.1016/j.compfluid.2014.02.021.
URL http://www.sciencedirect.com/science/article/pii/S004579301400084X
[88] K. C. Uga, M. Min, T. Lee, P. F. Fischer, Spectral-element discontinuous galerkin lattice boltzmann simulation of flow past two cylinders in tandem with an exponential time integrator, Computers & Mathematics with Applications 65 (2) (2013) 239 – 251, special Issue on Mesoscopic Methods in Engineering and Science (ICMMES-2010, Edmonton, Canada). doi:http://dx.doi.org/10.1016/j.camwa.2011.12.059.
URL http://www.sciencedirect.com/science/article/pii/S0898122111011308
[89] M. D. Mazzeo, Fast discontinuous galerkin lattice-boltzmann simulations on gpus via maximal kernel fusion, Computer Physics Communications 184 (3) (2013) 537–549.
[90] A. Fakhari, T. Lee, Finite-difference lattice boltzmann method with a block-structured adaptive-mesh-refinement technique, Phys. Rev. E 89 (2014) 033310. doi:10.1103/PhysRevE.89.033310.
URL http://link.aps.org/doi/10.1103/PhysRevE.89.033310
[91] A. Fakhari, M. Geier, T. Lee, A mass-conserving lattice boltzmann method with dynamic grid refinement for immiscible two-phase flows, Journal of Computational Physics 315 (2016) 434–457.
[92] A. Dupuis, B. Chopard, Theory and applications of an alternative lattice boltzmann grid refinement algorithm, Physical Review E 67 (6) (2003) 066707.
[93] B. Crouse, E. Rank, M. Krafczyk, J. Tölke, A lb-based approach for adaptive flow simulations, International Journal of Modern Physics B 17 (01n02) (2003) 109–112.
[94] D. Lagrava, O. Malaspinas, J. Latt, B. Chopard, Automatic grid refinement criterion for lattice boltzmann method, arXiv preprint arXiv:1507.06767.
[95] H. Chen, O. Filippova, J. Hoch, K. Molvig, R. Shock, C. Teixeira, R. Zhang, Grid refinement in lattice boltzmann methods based on volumetric formulation, Physica A: Statistical Mechanics and its Applications 362 (1) (2006) 158–167.
[96] M. Rohde, D. Kandhai, J. Derksen, H. Van den Akker, A generic, mass conservative local grid refinement technique for lattice-boltzmann schemes, International journal for numerical methods in fluids 51 (4) (2006) 439–468.
[97] C. E. Shannon, Communication in the presence of noise, Proceedings of the IRE 37 (1) (1949) 10–21.
[98] M. Geier, A. Greiner, J. Korvink, Bubble functions for the lattice boltzmann method and their application to grid refinement, The European Physical Journal Special Topics 171 (1) (2009) 173–179.
[99] J. Tölke, M. Krafczyk, Second order interpolation of the flow field in the lattice boltzmann method, Computers & Mathematics with Applications 58 (5) (2009) 898–902.
[100] M. Schönherr, K. Kucher, M. Geier, M. Stiebler, S. Freudiger, M. Krafczyk, Multi-thread implementations of the lattice Boltzmann method on non-uniform grids for cpus and gpus, Comput. Math. Appl. 61 (12) (2011) 3730–3743. doi:10.1016/j.camwa.2011.04.012.
[101] M. Geier, M. Schönherr, Esoteric twist: An efficient in-place streaming algorithmus for the lattice boltzmann method on massively parallel hardware, Computation 5 (2) (2017) 19.
[102] J. Qi, H. Klimach, S. Roller, Implementation of the compact interpolation within the octree based lattice boltzmann solver musubi, Computers & Mathematics with Applications.
[103] M. Geier, A. Greiner, J. G. Korvink, Bubble functions for the lattice Boltzmann method and their application to grid refinement, The European Physical Journal Special Topics 171 (1) (2009) 173–179. doi:10.1140/epjst/e2009-01026-6.
[104] C. De Waard, Le manuscrit perdu de Snellius sur la refraction, Brill, 1935.
[105] K. Hentschel, Das brechungsgesetz in der fassung von snellius, Archive for history of exact sciences 55 (4) (2001) 297–344.
[106] R. W. Johnson, Handbook of Fluid Dynamics, CRC Press, 1998.
[107] H. Teng, T. Zhao, An extension of darcy's law to non-stokes flow in porous media, Chemical engineering science 55 (14) (2000) 2727–2735.
[108] W. Breugem, B. Boersma, Direct numerical simulations of turbulent flow over a permeable wall using a direct and a continuum approach, Physics of Fluids (1994-present) 17 (2) (2005) 025103.
[109] J. Hunt, A. Wray, P. Moin, Eddies, streams, and convergence zones in turbulent flow, in: Center for Turbulence Research Proceedings of the Summer Program 1988, 1988.

[110] J. Jeong, F. Hussain, On the identification of a vortex, Journal of fluid mechanics 285 (1995) 69–94.
[111] M. Mößner, Volume-averaged RANS-simulation of Turbulent Flow Over Porous Media, TU Braunschweig, Niedersächsisches Forschungszentrum für Luftfahrt, 2016.
[112] M. Mößner, R. Radespiel, Numerical simulations of turbulent flow over porous media, AIAA2013-2963, 21st AIAA Computational Fluid.
[113] M. Quintard, S. Whitaker, Transport in ordered and disordered porous media ii: Generalized volume averaging, Transport in porous media 14 (2) (1994) 179–206.
[114] Y. Kuwata, K. Suga, Direct numerical simulation of turbulence over anisotropic porous media, Journal of Fluid Mechanics 831 (2017) 41–71.
[115] T. von Kármán, Mechanische ähnlichkeit und turbulenz, nachr. ges. wiss gottingen, math. phys. klasse; 1930, 5, English translation, NACA TM 611 (1931) 58–76.
[116] S. Hoyas, J. Jiménez, Scaling of the velocity fluctuations in turbulent channels up to re τ= 2003, Physics of fluids 18 (1) (2006) 011702.
[117] M. Lee, R. D. Moser, Direct numerical simulation of turbulent channel flow up to $Re_\tau \approx 5200$, arXiv preprint arXiv:1410.7809.
[118] H. Schlichting, K. Gersten, Boundary-layer theory, Springer Science & Business Media, 2003.
[119] Y. Kuwata, K. Suga, Lattice boltzmann direct numerical simulation of interface turbulence over porous and rough walls, International Journal of Heat and Fluid Flow 61 (2016) 145–157.
[120] C. Fureby, F. F. Grinstein, Large eddy simulation of high-reynolds-number free and wall-bounded flows, Journal of Computational Physics 181 (1) (2002) 68–97.