

# A robust lattice Boltzmann method for parallel simulations of multicomponent flows in complex geometries

Jens Zudrop<sup>a,b,\*</sup>, Kannan Masilamani<sup>b</sup>, Sabine Roller<sup>b</sup>, Pietro Asinari<sup>c</sup>

<sup>a</sup> Applied Supercomputing in Engineering, German Research School for Simulation Sciences, RWTH Aachen, Schinkelstrasse 2a, Aachen, Germany

<sup>b</sup> Simulation Techniques and Scientific Computing, University Siegen, Hölderlinstrasse 3, Siegen, Germany

<sup>c</sup> Department of Energy, Politecnico di Torino, Corso Duca degli Abruzzi 24, Torino, Italy

## ARTICLE INFO

### Article history:

Received 28 July 2014

Revised 5 April 2017

Accepted 21 April 2017

Available online 22 April 2017

### Keywords:

Lattice Boltzmann

Multicomponent flow

Maxwell–Stefan diffusion

High performance computing

Massively parallel simulations

Octree

## ABSTRACT

In this work, we are concerned with robust parallel simulations of multicomponent flows in complex geometries by an extended lattice Boltzmann method. In Zudrop et al. (2014) [22] we presented a model for the incompressible Navier–Stokes and Maxwell–Stefan diffusion equations. Here, we prove that the implicit-to-explicit variable transformation of the model is well-posed under weak constraints on the dimensionless Maxwell–Stefan diffusivities. Furthermore, we analyze various boundary conditions for the multicomponent lattice Boltzmann model. These results make the model robust and well suited for complex geometries, which allows us to study realistic, large-scale mass transport applications.

© 2017 Published by Elsevier Ltd.

## 1. Introduction

Multicomponent fluid flows are characterized by a homogeneous mixture of multiple species. Electrochemical and electrodiolytic processes are two of many examples (e.g. [1,2]). Due to the fact that many of these processes involve complex geometries or strongly non-linear processes, analytical predictions fail most often and experimental setups are not always possible. Numerical simulations are often the only approach to achieve a deeper understanding of such processes. However, the simulation of multicomponent flows is a quite extensive task due to the strongly non-linear behavior and requires huge computational resources for realistic setups. Classical numerical schemes, such as the finite volume method, have been applied to simple test cases [1,2]. For flows in complex geometries the Lattice Boltzmann (LB) method offers great potential due to its explicit nature [3–7]. However, the method cannot be applied without modifications to multicomponent flows. To properly take into account the momentum exchange among the species of a mixture in a BGK lattice Boltzmann scheme, several authors considered pseudo-potential interactions [8–11] or defined heuristic free energies in [12–15]. Other models [16,17] adopt a force coupling in the momentum equations, while the models in [18–20] have two collisional operators

(self and cross collisions). All these models have various shortcomings: Some of them work only for binary mixtures, others resolve only Fick's diffusion instead of Maxwell–Stefan diffusion and still others are not parameter free. In [21] another LB scheme for Maxwell–Stefan diffusion has been proposed. It aims to minimize a discrete  $\mathcal{H}$ -functional, but the model recovers the Maxwell–Stefan equations only within the macroscopic mixture averaged diffusion approximation. In [22] the authors presented a lattice Boltzmann method, consistent with gas and liquid multicomponent flows, which remedies these shortcomings. It is based on [20,23], thus it does not rely on the mixture averaged diffusion approximation. The model rigorously includes external forcing terms (on the viscous and mass diffusion scales), as well as thermodynamic factors in the Maxwell–Stefan closure relation (i.e. non-ideal diffusion effects).

The aforementioned publications presented a detailed mathematical analysis of the multicomponent LB methods, but very little attention was paid to computational aspects. In this work we discuss and demonstrate the robustness and efficiency of the multicomponent LB model introduced in [22] for simulations in complex geometries. Although the method is local and works on a single grid for all species, a few open points remain before the method can be considered as “robust”:

- The implicit-to-explicit variable transformation of the LB model [22] necessitates the solution of an element local linear equation system in each LB step. Although this technique has also

\* Corresponding author.

E-mail address: [j.zudrop@grs-sim.de](mailto:j.zudrop@grs-sim.de) (J. Zudrop).

been used in other models, such as [20,21,23], well-posedness (i.e. existence and uniqueness) of this equation system has not been investigated. Thus, it is a-priori unclear if the multicomponent LB iteration itself may fail under certain conditions or not.

- Multicomponent LB boundary conditions have been largely unaddressed in literature. This is a severe shortcoming, as almost all scientifically relevant simulation setups involve boundaries in one form or another. Moreover, locality and simplicity of the boundary handling in the LB method plays a decisive role for complex geometries like porous media.
- The computational workload of an  $N$ -species multicomponent LB scheme should be roughly the same as computing  $N$  standard LB schemes. Furthermore, scalability of the multicomponent LB method should be just as well as for the standard LB scheme. In this sense, the multicomponent scheme should maintain the computational advantages of the LB method. We note that this is in general not true for multicomponent LB methods on different lattice grids for each species which require costly interpolations [21].

In this work, all of the issues listed above are addressed successfully for the model presented in [22]: We prove that the element local equation system is well-posed under weak constraints on the dimensionless Maxwell–Stefan diffusion coefficients. Furthermore, we derive and analyze accurate boundary conditions for various setups. These boundary conditions maintain the efficiency of their single-species counterparts. To demonstrate robustness and efficiency, we apply the multicomponent LB model in the framework [24–26] and present results for parallel computations in complex geometries.

The outline of this paper is as follows: In Section 2, we recall the governing macroscopic equations, which are solved by the LB method. A summary of the LB model is given in Section 3 and in Section 4 we prove well-posedness of the implicit-to-explicit variable transformation. In Section 4.2, we analyze various boundary conditions for the multicomponent lattice Boltzmann method. In Section 5, we present the verification of our model using a well-known stefan tube and taylor dispersion experiments and plausible study of the proposed properties with some numerical experiments. This includes test cases to demonstrate accuracy and stability of the presented boundary conditions, as well as large-scale simulations in porous media. Section 6 briefly presents scalability and performance results. Finally, Section 7 gives a conclusion.

## 2. Governing equations of (non-)ideal mixtures

In this section we briefly recall the governing equations for a fluid consisting of  $N$  species in the low Mach regime. In general it is described by conservation equations for species mass and mixture momentum, plus an additional closure relation for the mass transport relative to the mixture flow. For species  $k \in \{1, \dots, N\}$  mass conservation reads as

$$\partial_t n_k + \nabla \cdot (n_k \mathbf{w}) = -\nabla \cdot \mathbf{J}_k, \quad (1)$$

where  $\mathbf{w} = \sum_k n_k \mathbf{v}_k / n$  is the number density averaged mixture velocity, computed with the species velocities  $\mathbf{v}_k$ .  $n_k = \rho_k / m_k$  and  $n = \sum_k n_k$  are the number density of species  $k$  and the total number density, respectively.  $\rho_k$  is the mass density of the species  $k$  and  $m_k$  the molecular weight of species  $k$ . We use the Maxwell–Stefan closure relation to determine the molar diffusive flux  $\mathbf{J}_k$ , which is defined for (non-)ideal mixtures [27,28] as

$$\mathbf{F}_k + \sum_i \Gamma_{k,i} \nabla \chi_i = \sum_{l \neq k} \frac{1}{n D_{k,l}} (\chi_k \mathbf{J}_l - \chi_l \mathbf{J}_k) \quad \forall k \in \{1, \dots, N\}, \quad (2)$$

where  $\chi_k = n_k / n$  denotes its number density fraction and  $D_{k,l}$  the Maxwell–Stefan diffusion coefficient between species  $k$  and

species  $l$ . The matrix  $\Gamma$  contains the so-called thermodynamic factors, formally derived by the concentration dependent activity of the species in the mixture [29]. For ideal mixtures, e.g. ideal gases, we have  $\Gamma = \mathbf{I}$ . However in non-ideal mixtures, e.g. liquid mixtures as electrolytes, this matrix deviates significantly from the identity matrix [29] and the left hand side of Eq. (2) is not decoupled. The external diffusive forcing terms  $\mathbf{F}_k$  are given by:

$$\mathbf{F}_k = \frac{1}{nRT} \rho_k \left( \mathbf{g} - \frac{z_k \mathcal{F}}{m_k} \mathbf{E} \right) - \frac{y_k}{nRT} \sum_{l=1}^N \rho_l \left( \mathbf{g} - \frac{z_l \mathcal{F}}{m_l} \mathbf{E} \right),$$

where  $\mathcal{F}$  denotes the Faraday constant,  $\mathbf{E}$  the electric field,  $\mathbf{g}$  the gravitational acceleration,  $T$  the mixture temperature,  $R$  the universal gas constant,  $z_k$  the ionic charge of species  $k$ ,  $y_k = \rho_k / \rho$  its mass fraction. By definition of  $\mathbf{F}_k$ , the sum of the external diffusive forces vanishes, i.e.:

$$\sum_k \mathbf{F}_k = 0$$

In general the external diffusive forces  $\mathbf{F}_k$  can be decomposed into two parts: The total external force for species  $k$  (i.e.  $\rho_k (\mathbf{g} - \mathbf{E} z_k \mathcal{F} / m_k) / (nRT)$ ) and the fraction of the total mixture force acting on the species  $k$  (i.e.  $y_k (\sum_l \rho_l (\mathbf{g} - \mathbf{E} z_l \mathcal{F} / m_l)) / (nRT)$ ).

We consider mixture flows in the isothermal low-Mach regime [22], so we obtain an incompressible Navier–Stokes equation for the mass averaged mixture velocity  $\mathbf{v} = \sum_l \rho_l \mathbf{v}_l / \rho$ . The equation reads as

$$\partial_t \rho \mathbf{v} + \nabla \cdot (\rho \mathbf{v} \otimes \mathbf{v}) = -\nabla p + \nu \nabla^2 \rho \mathbf{v} + \underbrace{\rho \mathbf{g} + \rho^e \mathbf{E}}_{=\mathbf{F}}, \quad (3)$$

where  $\mathbf{F}$  denotes the total mixture force per unit volume and  $\rho^e$  the charge density of the mixture. The multicomponent flow can be characterized by the dimensionless Reynolds and Schmidt number [27]

$$Re = \frac{|\mathbf{v}|L}{\nu}, \quad Sc = \frac{\nu}{D_{i,j}}.$$

## 3. Finite discrete velocity model for (non-)ideal mixtures

A standard lattice-Boltzmann scheme is based on a discretization of the finite discrete velocity model of the Boltzmann equation. The latter is continuous in time and space, while the phase space (i.e. the mesoscopic velocities) are discretized. Choosing a discrete set of lattice velocities  $\mathbf{u}^m$  for  $m \in \{0, \dots, M-1\}$ , it is given by

$$\partial_t f^m + \mathbf{u}^m \cdot \nabla f^m = \lambda (f^{eq,m} - f^m) \quad (4)$$

where  $f^m$  denotes the probability density function and  $f^{eq,m}$  a properly defined equilibrium distribution.

In this model the relaxation parameter  $\lambda$  is related to the viscosity of the fluid [30]. Macroscopic quantities are obtained as moments of  $f^m$ , e.g. density  $\rho = \sum_m f^m$  and velocity  $\mathbf{v} = \sum_m \mathbf{u}^m f^m$ . In the following, we denote macroscopic moments of  $f^m$  by

$$\Pi_{x^p, y^q} (f^m) = \langle 1, (u_x^m)^p (u_y^m)^q f^m \rangle = \sum_m (u_x^m)^p (u_y^m)^q f^m,$$

where  $u_x^m, u_y^m$  are the  $x, y$ -components of  $\mathbf{u}^m$  and  $p, q \in \mathbf{N}$  (we refer to  $p+q$  as the order of the moment). Mathematically, a connection between Eq. (4) and the incompressible Navier–Stokes equation can be obtained by an asymptotic analysis in terms of a formal smallness parameter  $\epsilon \rightarrow 0$ , where  $\Delta x \approx \epsilon$  and  $\Delta t \approx \epsilon^2$ . For the details we refer the reader to [30].

To extend the lattice Boltzmann method to multicomponent flows, an equation of type (4) has to be solved for each individual species. In addition the interaction of the species has to be taken into account, which is usually achieved by a modification

of the equilibrium part and the relaxation towards it. In [22] we presented such a finite discrete velocity model (FDV model) that is able to recover the correct macroscopic Eqs. (1)–(3) in the diffusive, asymptotic limit. It has the following form (where  $m \in \{0, \dots, M-1\}$ ):

$$\partial_t f_k^m + \mathbf{u}^m \cdot \nabla f_k^m = \lambda_k (f_k^{eq,m} - f_k^m) \quad (5)$$

Here, the relaxation parameter  $\lambda_k$  for species  $k$  is defined by:

$$\lambda_k = \frac{KB}{\rho},$$

where  $K$  is the bulk modulus of the mixture and  $B$  is a free parameter to be chosen with respect to stability, later on. As it can be seen, we assign the same relaxation parameter to all the species in our model. If not mentioned otherwise, we choose  $B$  such that  $\lambda_k = 2$ . Furthermore, the equilibrium of our model is given by

$$f_k^{eq,m} = \omega^m \rho_k \left( s_k^m + \frac{1}{c_s^2} (\mathbf{u}^m \cdot \mathbf{v}_k^*) + \frac{1}{2c_s^4} (\mathbf{u}^m \cdot \mathbf{v})^2 - \frac{1}{2c_s^2} \mathbf{v}^2 \right), \quad (6)$$

where  $\omega^m$  are the lattice weights [31]. The equilibrium is defined with a modified velocity  $\mathbf{v}_k^*$  for each species  $k$  in the bilinear equilibrium part

$$\mathbf{v}_k^* (\mathbf{v}_1, \dots, \mathbf{v}_N) = \mathbf{v}_k + \sum_l \frac{B_{k,l}}{B} \phi_k \chi_l (\mathbf{v}_l - \mathbf{v}_k) \quad (7)$$

and mass averaged mixture velocity  $\mathbf{v}$  in the quadratic equilibrium part to recover Maxwell–Stefan diffusion equations for each species  $k$  and incompressible Navier–Stokes equations for the mixture respectively [22].

Similar to the single-species finite discrete velocity model we define the moments of species  $k$  as  $\Pi_{x^p, y^q}^k = \Pi_{x^p, y^q}(f_k^m)$ . The parameters in (6), and (7) for a D2Q9 lattice are defined by [22]

$$\phi_k = \frac{\min_a m_a}{m_k} \leq 1, \quad B_{k,l} = \frac{1}{D_{k,l}}, \quad c_s^2 = \frac{K}{\rho}, \quad \tau = \frac{1}{B},$$

$$s_k^m = \begin{cases} (9 - 5\phi_k)/4 & \text{if } m = 0 \\ \phi_k & \text{else} \end{cases}, \quad (8)$$

where  $\min_a m_a$  is the minimal molecular weight of all species. The extension to other lattices and other spatial dimensions is straight forward.

It should be noted that not all material parameters (i.e.  $\nu$ ,  $D_{k,l}$ ) can be tuned independently in our BGK model, but this restriction can be easily overcome by a multiple relaxation time (MRT) model [20], which uses a specific relaxation for distinct moments and reads as

$$\partial_t f_k^m + \mathbf{u}^m \cdot \nabla f_k^m = \mathbf{M}^{-1} \mathbf{\Lambda}_k \mathbf{M} (f_k^{eq,m} - f_k^m).$$

The matrix  $\mathbf{M}$  converts the mesoscopic state  $f_k^m$  to moments  $\Pi_{x^0, y^0}, \Pi_{x^1, y^0}, \Pi_{x^0, y^1}, \dots$ .

The MRT matrices  $\mathbf{M}$ ,  $\mathbf{\Lambda}_k$  for the D2Q9 and D3Q19 lattice in  $\mathbb{R}^2$  and  $\mathbb{R}^3$  are given in Appendix A and Appendix B respectively.

### 3.1. External forces

External driving forces are often a key part in engineering applications to steer the overall mass transport, e.g. electric fields in an electroanalysis. Therefore it is mandatory to impose them correctly in numerical schemes for mass transport. The finite discrete velocity model with external forcing terms can be written as

$$\partial_t f_k^m + \mathbf{u}^m \cdot \nabla f_k^m = \lambda_k (f_k^{eq,m} - f_k^m) + d_k^m. \quad (9)$$

The mesoscopic source term  $d_k^m$  is related to  $\mathbf{F}_k$  by [22]

$$d_k^m = \omega^m \mathbf{u}^m \cdot \left( n^0 \min_a m_a \mathbf{F}_k + \frac{1}{c_s^2} \rho_k^0 \mathbf{g} + \frac{\mathcal{F}}{c_s^2} n_k^0 z_k \mathbf{E} \right).$$

### 3.2. Thermodynamic factors

Besides the external forcing terms, thermodynamic factors can be included into the finite discrete velocity model [22] to incorporate non-ideal diffusion in the Maxwell–Stefan closure relation. This is accomplished by a modification of the bilinear part of the thermodynamic equilibrium

$$f_k^{eq,m,BL}(\rho_1, \dots, \rho_N, \mathbf{v}_1, \dots, \mathbf{v}_N) = \frac{\omega^m}{c_s^2} \mathbf{u}^m \cdot \left( \rho_k \mathbf{v}_k + \sum_l \Gamma_{k,l}^{-1} \rho_l \sum_\zeta \chi_\zeta \frac{B_{l,\zeta}}{B} \phi_l (\mathbf{v}_\zeta - \mathbf{v}_l) \right).$$

In general any activity model which delivers composition derivatives can be used to compute the thermodynamic factor matrix  $\Gamma$  in our model. Unless otherwise stated, we use the electrolyte-NRTL model. A detailed comparison of the existing models can be found in [32].

### 4. Lattice Boltzmann method

The fully discrete Lattice Boltzmann Model (LBM) is derived in [22] by integration along the characteristics [33]. The particular LB model formulation is obtained by choosing a discretization for integration of the right hand side of Eq. (5); for stability and accuracy a trapezoidal rule is used [22]. Clearly, this seems to be unattractive, because it makes the scheme implicit. Additionally, the future values of  $f_k^{eq,m}$  are nonlinear functions of  $f_k^m$ . To overcome this problem in [22] we followed the idea of [33] and apply an appropriate redefinition of the probability density functions

$$\tilde{f}_k^m = f_k^m + \frac{\delta t \lambda_k}{2} (f_k^m - f_k^{eq,m}). \quad (10)$$

This definition leads to the following LBM like scheme

$$\tilde{f}_k^m(\mathbf{x} + \mathbf{u}^m \delta t, t + \delta t) = \tilde{f}_k^m(\mathbf{x}, t) + \frac{\delta t}{\frac{1}{\lambda_k} + \frac{\delta t}{2}} (f_k^{eq,m}(\mathbf{x}, t) - \tilde{f}_k^m(\mathbf{x}, t)). \quad (11)$$

For convenience we define  $\delta x = \mathbf{u}^m \delta t$ . Similarly, we obtain for the MRT model

$$\tilde{f}_k^m = f_k^m + \frac{\delta t \mathbf{M}^{-1} \mathbf{\Lambda}_k \mathbf{M}}{2} (f_k^m - f_k^{eq,m})$$

and

$$\tilde{f}_k^m(\mathbf{x} + \mathbf{u}^m \delta t, t + \delta t) = \tilde{f}_k^m(\mathbf{x}, t) + (\mathbf{M}^{-1} \mathbf{\Lambda}_k \mathbf{M}) \left( \mathbf{I} + \delta t \frac{\mathbf{M}^{-1} \mathbf{\Lambda}_k \mathbf{M}}{2} \right)^{-1} \times (f_k^{eq,m}(\mathbf{x}, t) - \tilde{f}_k^m(\mathbf{x}, t)). \quad (12)$$

Notice, that the right hand side collision operator contains the untransformed equilibrium probability density function. This is built with the macroscopic moments of  $f_k^m$  instead of  $\tilde{f}_k^m$ , and unfortunately the moments of  $f_k^m$  and  $\tilde{f}_k^m$  are unfortunately not equal [22]. However, the problem of unequal macroscopic moments can be overcome by solving a cell-local linear equation system [22], consequently we refer to (11) as the “semi-implicit lattice Boltzmann scheme”.

Now, we show that the algorithm presented in [22] is well-suited for an efficient implementation, and can be used for application to large-scale problems. In the following Section 4.1 we prove that the linear equation system, arising from the implicit-to-explicit variable transformation (11), is well-posed. Uniqueness and existence of the solution are guaranteed under weak constraints on the dimensionless Maxwell–Stefan diffusion coefficients. In Section 4.2 we present and analyze boundary conditions for the multicomponent lattice Boltzmann model.

#### 4.1. Well-posedness of the implicit-to-explicit variable transformation

By direct calculation it is easy to check that

$$\langle 1, \bar{f}_k^m \rangle = \langle 1, f_k^m \rangle + \frac{\delta t \lambda_k}{2} (\langle 1, f_k^m \rangle - \langle 1, f_k^{eq,m} \rangle) = \langle 1, f_k^m \rangle = \rho_k.$$

Thus, density moments of  $\bar{f}_k^m$  and  $f_k^m$  equal each other. We denote the moments of the redefined probability density function  $\bar{f}_k^m$  with upper bars, i.e. if we consider the first order moment we get

$$\begin{aligned} \bar{\mathbf{j}}_k &= \langle 1, \mathbf{u}^m \bar{f}_k^m \rangle = \mathbf{j}_k + \frac{\delta t \lambda_k}{2} (\mathbf{j}_k - \langle 1, \mathbf{u}^m f_k^{eq,m} \rangle) \\ &= \mathbf{j}_k + \frac{\delta t \lambda_k}{2} \sum_l \frac{B_{k,l}}{B} \phi_k \chi_l \mathbf{j}_l - \frac{\delta t \lambda_k}{2} \sum_l \frac{B_{k,l}}{B} \chi_k \phi_l \mathbf{j}_l. \end{aligned} \quad (13)$$

If  $N$  denotes the number of species in our multicomponent scheme, the upper Eq. (13) specifies an  $N \times N$  equation system for each component of  $\mathbf{j}$  of the following type:

$$\mathbf{A} \cdot \begin{pmatrix} \mathbf{j}_1 \\ \vdots \\ \mathbf{j}_N \end{pmatrix} = \begin{pmatrix} \bar{\mathbf{j}}_1 \\ \vdots \\ \bar{\mathbf{j}}_N \end{pmatrix}$$

To ensure solvability of the upper equation system we check for strict diagonal dominance, i.e. we check if the relation

$$|1 + \frac{\delta t \lambda_k}{2} \sum_{l \neq k} \frac{B_{k,l}}{B} \phi_k \chi_l| > |\frac{\delta t \lambda_k}{2} \sum_{l \neq k} \frac{B_{k,l}}{B} \chi_k \phi_l| \quad \forall k \in \{1, \dots, N\}$$

holds. We can neglect the modulus on both sides, as all terms are always greater or equal than zero. We now consider:

$$\begin{aligned} 1 &> \frac{\delta t \lambda_k}{2} \sum_{l \neq k} \frac{B_{k,l}}{B} (\chi_k \phi_l - \chi_l \phi_k) \\ \Leftrightarrow 1 &> \frac{\delta t \lambda_k}{2} \sum_{l \neq k} \frac{B_{k,l}}{B} \phi_k \phi_l \frac{\rho_k - \rho_l}{\rho} \quad \forall k \in \{1, \dots, N\} \end{aligned}$$

Now, we try to find an upper bound on the relative density difference, i.e. we search for  $C$  such that:

$$|\frac{\rho_k - \rho_l}{\rho}| \leq C \quad \forall k, l$$

If we find this bound, we know by

$$C \frac{\delta t \lambda_k}{2} \sum_{l \neq k} \frac{B_{k,l}}{B} \phi_k \phi_l \geq \frac{\delta t \lambda_k}{2} \sum_{l \neq k} \frac{B_{k,l}}{B} \phi_k \phi_l \frac{\rho_k - \rho_l}{\rho} \quad \forall k \in \{1, \dots, N\}$$

that the condition

$$1 > C \frac{\delta t \lambda_k}{2} \sum_{l \neq k} \frac{B_{k,l}}{B} \phi_k \phi_l \quad \forall k \in \{1, \dots, N\}$$

guarantees solvability. To find this constant  $C$  we consider the following expression:

$$|\frac{\rho_k}{\rho} - \frac{\rho_l}{\rho}| \leq |\frac{\rho_k}{\rho}| + |\frac{\rho_l}{\rho}| \leq 1$$

So,  $C = 1$ , and it is sufficient to guarantee, that

$$2 > \delta t \lambda_k \cdot \sum_{l \neq k} \frac{B_{k,l}}{B} \phi_k \phi_l \quad (14)$$

and we notice that it is even more restrictive to require

$$2 > \delta t \lambda_k \cdot \sum_{l \neq k} \frac{B_{k,l}}{B} \geq \delta t \lambda_k \cdot \sum_{l \neq k} \frac{B_{k,l}}{B} \phi_k \phi_l,$$

as  $\phi_b \leq 1$  for all the species. By inserting our definition of  $\lambda_k$ , (A.1), we obtain:

$$2 > \delta t \frac{K}{\rho} \cdot \sum_{l \neq k} B_{k,l}$$

Finally, we use (8) and conclude that

$$\frac{1}{c_s^2} > \frac{\delta t}{2} \cdot \sum_{l \neq k} B_{k,l} \quad \forall k \in \{1, \dots, N\}$$

guarantees solvability of our linear equation system. In lattice units the upper condition reads as follows (with  $c_s = 1/\sqrt{3}$ ,  $\delta t = 1$  and the lattice Maxwell–Stefan resistivities  $B'_{k,l}$ ):

$$6 > \sum_{l \neq k} B'_{k,l} \quad \forall k \in \{1, \dots, N\} \quad (15)$$

As all other parameters (like  $\delta t$  and  $c_s$ ) are fixed in lattice units, there is only the possibility to tune for the parameters  $B'_{k,l}$ . It is worth noticing that in physical units the Maxwell–Stefan resistivities are  $B_{k,l} = [s/m^2]$  and due to diffusive scaling, we scale the time step and element size by  $\tilde{\Delta t} = \epsilon^2 \Delta t$  and  $\tilde{\Delta x} = \epsilon \Delta x$  (cf. [22]):

$$B'_{k,l} = B_{k,l} \cdot \frac{(\tilde{\Delta x})^2}{\tilde{\Delta t}} = B_{k,l} \cdot \frac{(\Delta x)^2}{\Delta t}$$

This proves: Once solvability of the linear equation system is satisfied (by means of Eq. (15)) on a mesh with a specific mesh resolution, this directly implies that solvability can be guaranteed for an arbitrarily refined mesh (as long as the refinement is done by a diffusive scaling starting from the mesh where solvability is guaranteed). In contrast, for acoustic scaling we scale for  $\Delta x$  and  $\Delta t$  by  $\Delta x = \epsilon \Delta x$  and  $\Delta t = \epsilon \Delta t$  and therefore we end up with:

$$B'_{k,l} = B_{k,l} \cdot \frac{(\tilde{\Delta x})^2}{\tilde{\Delta t}} = \epsilon B_{k,l} \cdot \frac{(\Delta x)^2}{\Delta t}$$

Thus, for acoustic scaling Eq. (15) will be satisfied for all  $\epsilon < \epsilon_0$  when we consider  $\epsilon \rightarrow 0$ .

A similar solvability condition can be obtained for the multiple relaxation time model in Eq. (12). After some algebra one finds that

$$6 > \sum_{l \neq k} \lambda_k^\delta B'_{k,l} \quad \forall k \in \{1, \dots, N\} \quad (16)$$

is sufficient to guarantee solvability of the implicit-to-explicit variable transformation for the multiple relaxation time lattice Boltzmann scheme (12).

##### 4.1.1. Thermodynamic factors

When thermodynamic factors are included in our lattice Boltzmann model, the linear equation system in equation (13) has to be replaced by the following linear equation system

$$\bar{\mathbf{j}}_k = \mathbf{j}_k + \frac{\delta t \lambda_k}{2} \sum_l \Gamma_{k,l}^{-1} \sum_\zeta \frac{B_{l,\zeta}}{B} \chi_\zeta \phi_l \mathbf{j}_l - \frac{\delta t \lambda_k}{2} \sum_l \Gamma_{k,l}^{-1} \sum_\zeta \frac{B_{l,\zeta}}{B} \chi_l \phi_\zeta \mathbf{j}_\zeta. \quad (17)$$

Following a similar analysis as above, one can easily show that

$$1 > \sum_\zeta \left| \frac{\partial \ln \gamma_k}{\partial \chi_\zeta} \right| + \frac{\delta t \lambda_k}{2} \sum_\zeta \frac{B_{k,\zeta}}{B} \quad \forall k \in \{1, \dots, N\} \quad (18)$$

is a sufficient condition to ensure well-posedness of the linear equation system. Compared to Eq. (14) an additional term appears, which imposes conditions on the thermodynamic factor model to guarantee solvability of the linear equation system (17). As thermodynamic factor models are quite complex, we were not able to prove well-posedness theoretically for this setup. However, in the course of the numerical experiments in Section 5, solvability of the linear equation systems (13), and (17) respectively, was always guaranteed.



#### 4.1.2. External driving forces

In case of a finite discrete velocity model which includes external forces, the left hand side of the model is unmodified. Formally, we integrate the right hand side of the finite discrete velocity model over the time step of length  $\delta t$  in the same way. Applying the variable transformation (10) as in the case without external forces, we obtain:

$$\begin{aligned} \tilde{f}_k^m(\mathbf{x} + \mathbf{u}^m \delta t, t + \delta t) = & \tilde{f}_k^m(\mathbf{x}, t) + \frac{\delta t}{\frac{1}{\lambda_k} + \frac{1}{2}} (f_k^{eq,m}(\mathbf{x}, t) - \tilde{f}_k^m(\mathbf{x}, t)) \\ & + \frac{\delta t}{2} (d_k^m(\mathbf{x} + \mathbf{u}^m \delta t, t + \delta t) + d_k^m(\mathbf{x}, t)) \end{aligned}$$

Hence, the forcing term has to be available at the future point in time  $t + \delta t$  and the current point in time  $t$  to be able to evolve the solution to the next time step by this lattice Boltzmann algorithm. In case that external forcing terms are obtained by other numerical solvers, e.g. in multi-physics simulations, these terms have to be provided for the current and next time step. To overcome this problem, we can approximate the integral for the forcing terms by a (first order) forward Euler scheme. In this case the external forces have to be delivered only at the previous point in time. As the external forces are usually small (to be consistent with the low Mach regime) the limitations with respect to stability are moderate. The solvability conditions of the linear equations systems (13) and (17) are not affected by the external forcing term.

#### 4.2. Boundary conditions

Besides a proper PDE discretization, numerical simulations on bounded domains require a specific set of boundary conditions to truncate the simulation domain (depending on the type of the numerical test case under consideration). Boundary conditions are (in most cases) given in terms of macroscopic moments of the target equation, e.g.  $\rho_k, \mathbf{v}, \mathbf{v}_k, \dots$ . As the number of boundary conditions is usually insufficient to determine the state in mesoscopic space, i.e.  $f_k^m$ , imposing boundary conditions in a LBM is by no means a trivial task. However, once the asymptotic analysis of the kinetic scheme is solved, it is possible to prescribe boundary conditions in accordance with the asymptotic series expansion to recover the expected order of convergence of the numerical scheme. In [34,35] a generalized analysis of boundary conditions is given. Although the analysis in both publications is to certain extent “complete”, the resulting boundary conditions are often complicated to implement in numerical simulation codes, as they require additional neighboring nodes for extra- or interpolations (which is often problematic for complex geometries e.g. porous media).

##### 4.2.1. Moment based boundary conditions

Therefore, we follow the concept of moment based boundary conditions presented in [36]. This type of boundary conditions is robust, can be used for any specific macroscopic boundary condition, and is purely local. Furthermore, it can be applied even in presence of the variable transformation (10). The key concept of the boundary conditions in [36] is to recover the unknown  $f_k^m$  at the boundaries (after streaming) by reconstructing them out of the remaining known states and imposed macroscopic moments. To ensure that this reconstruction is well-posed, only specific groups of macroscopic moments can be imposed. In the following we generalize the boundary conditions of [36] to the model we presented in [22]. In general all elements intersected by an obstacle are marked as boundary elements and reconstruction is applied to the missing  $f_k^m$ , a simplified setup for a circular mesh is shown in Fig. 1. In the specific setup depicted in Fig. 1 the missing links  $f_3, f_7, f_4$  have to be reconstructed according to the given values for  $f_0, f_8, f_1, f_5, f_2, f_6$  and the imposed boundary conditions at the wall. By

checking the linearly independent information, we can prescribe the moments

$$\Pi_{x^1, y^0}^k = \Pi_{x^0, y^1}^k = 0$$

$$\Pi_{x^2, y^1}^k = \Pi_{x^1, y^2}^k = 0$$

at the boundary node. Notice that in our model  $\Pi_{x^1, y^0}(f_k^m) \neq \Pi_{x^1, y^0}(\tilde{f}_k^m)$ , and hence  $\Pi_{x^1, y^0}(f_k^m) \neq \Pi_{x^1, y^0}(\tilde{f}_k^m)$ . Thus, the transformed equation variable  $\tilde{f}_k^m$  is constructed from reconstructed missing links of  $f_k^m$  using Eq. (10).

Due to the moment based nature of the boundary conditions, the presented approach is easily applicable for the multiple relaxation time lattice Boltzmann model (12), too. Furthermore, it also works whenever thermodynamic factors and concentration dependent diffusivities are included.

##### 4.2.2. Bounce-back based boundary conditions

The bounce-back boundary condition is probably the most well-known boundary condition for lattice Boltzmann methods. Even though it can be analyzed in terms of moment based boundary conditions [36] too, it is purely defined in terms of  $f_k^m$  instead of  $\tilde{f}_k^m$ . To adapt the bounce back rule to our LBM, which is defined in terms of  $\tilde{f}_k^m$ , we apply the following strategy: The transformed variable  $\tilde{f}_k^m$  is rewritten (temporarily) as  $f_k^m$ , then simple bounce back is carried out in terms of  $f_k^m$ . Afterwards the resulting state is again written in terms of  $\tilde{f}_k^m$  and the usual lattice Boltzmann algorithm can be carried out. For efficiency reasons we rewrite it as a one step updating rule, without explicitly involving the untransformed variable.

To update an element adjacent to the boundary from time  $t$  to  $t + \delta t$  at point  $\mathbf{x}$  the bounce back rule reads as

$$\begin{aligned} \tilde{f}_k^{c(m)}(\mathbf{x}, t + \delta t) = & (\tilde{f}_k^m(\mathbf{x}, t^p) + \lambda_k \delta t / 2 f_k^{eq,m}(\mathbf{x}, t)) \\ & - \lambda_k \delta t / 2 f_k^{eq,c(m)}(\mathbf{x}, t + \delta t), \end{aligned}$$

where  $c(m)$  denotes the mirrored direction of  $m$ , i.e.  $-\mathbf{u}^m = \mathbf{u}^{c(m)}$ , and  $t^p$  the post-collision state. Therefore, it corresponds to a bounce back in terms of  $\tilde{f}_k^m(\mathbf{x}, t^p) + \lambda_k / 2 f_k^{eq,m}(\mathbf{x}, t)$  followed by an additional summation. We can further simplify the upper boundary conditions by expanding  $f_k^{eq,m}(\mathbf{x}, t)$  in a Taylor series around  $(\mathbf{x}, t + \delta t)$ , we obtain

$$f_k^{eq,m}(\mathbf{x}, t) = f_k^{eq,m}(\mathbf{x}, t + \delta t) + \mathcal{O}(\delta t).$$

According to the definition of the equilibrium distribution, we notice that

$$f_k^{eq,m}(\mathbf{x}, t + \delta t) - f_k^{eq,c(m)}(\mathbf{x}, t + \delta t) = \frac{2\omega^{c(m)}\rho_k}{c_s^2} (-\mathbf{u}^{c(m)} \cdot \mathbf{v}_k^*(t + \delta t)),$$

and therefore the bounce back rule can be rewritten as

$$\tilde{f}_k^{c(m)}(\mathbf{x}, t + \delta t) = \tilde{f}_k^m(\mathbf{x}, t^p) + (\lambda_k \delta t \omega^m \rho_k / c_s^2) (\mathbf{u}^m \cdot \mathbf{v}_k^*(t + \delta t)) + \mathcal{O}(\delta t). \quad (19)$$

Notice that, according to the applied variable transformation, the modified mixture velocity  $\mathbf{v}_k^*$  can be computed for  $t + \delta t$  by solving an element local linear equation system

$$\bar{\mathbf{v}}_k = \mathbf{v}_k + \frac{\delta t \lambda_k}{2} \sum_l \frac{B_{k,l}}{B} \phi_k \chi_l \mathbf{v}_k - \frac{\delta t \lambda_k}{2} \frac{1}{\rho_k} \sum_l \frac{B_{k,l}}{B} \chi_k \phi_l \rho_l \mathbf{v}_l. \quad (20)$$

for  $\mathbf{v}_k$  and plugging it in Eq. (7). Hence, the scheme remains fully explicit. A slightly different choice is to expand  $f_k^{eq,c(m)}(\mathbf{x}, t + \delta t)$  around  $t$ . However, due to the extrapolatory nature of the boundary term, the resulting equation, equivalent to Eq. (19) but build with  $\mathbf{v}_k^*$  at time  $t$ , might be stable only for small  $\delta t$ .

In the asymptotic continuum limit we scale  $\delta t \sim \epsilon^\alpha \rightarrow 0$  (with  $\alpha = 1$  for the acoustic continuum limit and  $\alpha = 2$  for the diffusive

continuum limit [30,37]), hence the bounce back rule (19) is equivalent to

$$\bar{f}_k^{c(m)}(\mathbf{x}, t + \delta t) = \bar{f}_k^m(\mathbf{x}, t^p) + \mathcal{O}(\epsilon^\alpha)$$

in the continuum limit, i.e. bounce-back in terms of the transformed variable. The upper simple bounce back boundary condition can be implemented very efficiently, even for complex geometrical setups.

By construction of the bounce back condition, the boundary is located at the boundary of the lattice Boltzmann element. However, we can apply the concept of  $q$ -values, as presented in [38] for the single-species lattice Boltzmann algorithm, in our multicomponent model to improve accuracy in case of curved boundaries. The corresponding equation for a link at a boundary reads as

$$\begin{aligned} \bar{f}_k^{c(m)}(\mathbf{x}, t + \delta t) = & 2q \left( \bar{f}_k^m(\mathbf{x}, t^p) + \frac{\lambda_k \delta t}{2} f_k^{eq,m}(\mathbf{x}, t) \right) \\ & + (1 - 2q) \left( \bar{f}_k^m(\mathbf{x} - \mathbf{u}^m \delta t, t^p) + \frac{\lambda_k \delta t}{2} f_k^{eq,m}(\mathbf{x} - \mathbf{u}^m \delta t, t) \right) \\ & - \frac{\lambda_k \delta t}{2} f_k^{eq,c(m)}(\mathbf{x}, t + \delta t) \quad , \text{ if } q < 1/2 \end{aligned} \quad (21a)$$

$$\begin{aligned} \bar{f}_k^{c(m)}(\mathbf{x}, t + \delta t) = & \frac{1}{2q} \left( \bar{f}_k^m(\mathbf{x}, t^p) + \frac{\lambda_k \delta t}{2} f_k^{eq,m}(\mathbf{x}, t) \right) \\ & + \frac{2q - 1}{2q} \left( \bar{f}_k^{c(m)}(\mathbf{x}, t^p) + \frac{\lambda_k \delta t}{2} f_k^{eq,c(m)}(\mathbf{x}, t) \right) \\ & - \frac{\lambda_k \delta t}{2} f_k^{eq,c(m)}(\mathbf{x}, t + \delta t) \quad , \text{ if } q \geq 1/2. \end{aligned} \quad (21b)$$

After some algebra we find that the upper equation is asymptotically equivalent to

$$\begin{aligned} \bar{f}_k^{c(m)}(\mathbf{x}, t + \delta t) &= \begin{cases} 2q \bar{f}_k^m(\mathbf{x}, t^p) + (1 - 2q) \bar{f}_k^m(\mathbf{x} - \mathbf{u}^m \delta t, t^p) + \mathcal{O}(\epsilon^\alpha), & \text{if } q < 1/2, \\ \frac{1}{2q} \bar{f}_k^m(\mathbf{x}, t^p) + \frac{2q-1}{2q} \bar{f}_k^{c(m)}(\mathbf{x}, t^p) + \mathcal{O}(\epsilon^\alpha), & \text{if } q \geq 1/2. \end{cases} \end{aligned} \quad (22)$$

Similar to [38], velocity boundary conditions can be imposed by adding a scaled antisymmetric equilibrium part, built with the imposed velocity  $\mathbf{v}_k^{bc}$ , to the right hand side of Eq. (22). The additional terms are

$$\delta \bar{f}_k^{c(m)} = \begin{cases} \frac{2\omega_k^m}{c_s^2} \mathbf{u}^{c(m)} \cdot \rho_k \mathbf{v}_k^{bc}, & \text{if } q < 1/2, \\ \frac{\omega_k^m}{q c_s^2} \mathbf{u}^{c(m)} \cdot \rho_k \mathbf{v}_k^{bc}, & \text{if } q \geq 1/2. \end{cases} \quad (23)$$

Notice that, according to the asymptotic analysis of [22], we have  $f_k^m = f_k^0 + \epsilon \frac{\omega_k^m}{c_s^2} \mathbf{u}^m \cdot \rho_k \mathbf{v}_k + \mathcal{O}(\epsilon^2)$  and hence, we use  $\mathbf{v}_k^{bc}$  instead of  $\mathbf{v}_k^*$  for the correction term in Eq. (23).

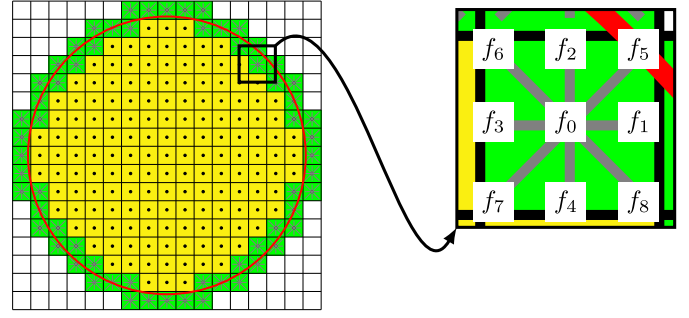
As reported in [36], the simple bounce back rule (without  $q$ -values) can be analyzed in terms of the moment based boundary conditions and for our multicomponent lattice Boltzmann scheme, it is equivalent to imposing the moments

$$\Pi_{x^0, y^1}^k = 0$$

$$\Pi_{x^1, y^2}^k = 0$$

$$\Pi_{x^2, y^1}^k = 0$$

at the wall (the upper relations are derived for a horizontal wall in two dimensional space). The momentum in  $x$  direction along the wall is small  $\Pi_{x^1, y^0}^k \in \mathcal{O}(\epsilon)$ , hence it recovers the no-slip boundary condition in the continuum limit [36]. Most other boundary conditions, e.g. non-equilibrium bounce-back [39], can be analyzed and used in the same way.



**Fig. 1.** Moment-boundary element (green) and fluid element (yellow) distribution in a 2D circular mesh. The missing links at the boundary elements are recovered after the streaming step by the moment based boundary conditions. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

In a computational implementation, moment based boundary conditions are usually prescribed by additional boundary elements, a schematic mesh setup is shown in Fig. 1 (in fact boundary elements may exist only virtually in a code implementation to improve efficiency for complex geometries). After execution of the streaming step, the boundary nodes are reconstructed according to the imposed boundary condition and actual orientation of the boundary. After filling these boundary elements, the remaining steps of the algorithm can be carried out, e.g. collision step, evaluation of the material parameters like diffusivities and activities, cf. Fig. 2. We apply an operator splitting approach to compute quasi-steady state external driving forces, e.g. gravitational forces, electric potentials. In contrast to moment based boundary conditions, bounce-back boundaries are implemented without additional boundary elements. This makes them more efficient with regards to memory and computational cost.

## 5. Numerical experiments

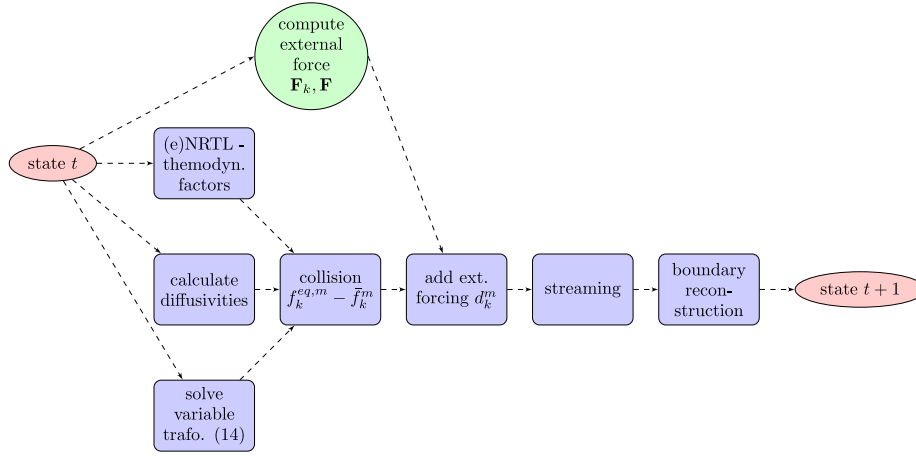
In the following we present simulation results of the LBM proposed in Section 4. For our numerical experiments we use the D2Q9 and D3Q19 stencil. We measure all errors in relative  $L_2$ -norm, e.g. for velocity we compute

$$\|e\|_{L_2} = \sqrt{\frac{\int \|\mathbf{v}_{ref} - \mathbf{v}\|_{L_2}^2 dV}{\int \|\mathbf{v}_{ref}\|_{L_2}^2 dV}}.$$

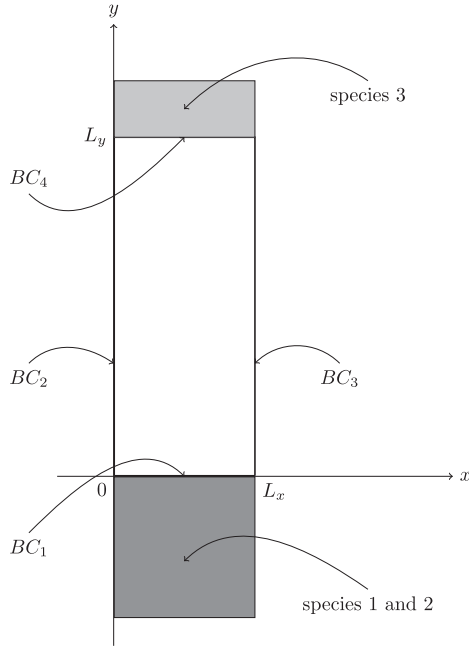
The numerical experiments are organized as follows: In Section 5.1 we study the Stefan tube experiment. This diffusion dominated test case shows that the lattice Boltzmann scheme (and its boundary conditions) recovers the Maxwell–Stefan diffusion equations. We compare our simulation results to reference results obtained by a shooting method [40] for the steady state mass transport equations. In Section 5.2 we study the Taylor dispersion experiment. This test case demonstrates that our model works in the convection dominated regime and the proposed boundary conditions are accurate and stable. Finally, in Section 5.3, we consider a three dimensional mixing process of three species inside a porous medium. Overall the numerical simulation has more than 1.2 billion species elements, solved on more than 20,000 cores. These results, in combination with the computational efficiency of the model (as presented in the following Section 6), show that the model is able to handle large scale, complex simulation setups.

### 5.1. Stefan tube

We consider a two-dimensional tube (aligned in  $y$  direction) of dimension  $[0; L_x] \times [0; L_y]$ , cf. Fig. 3), with three species. We prescribe the following boundary conditions:



**Fig. 2.** Algorithmic steps of the mixture lattice Boltzmann scheme (with moment based boundary conditions). Here, a simple operator splitting approach is used for quasi-steady state external driving forces.



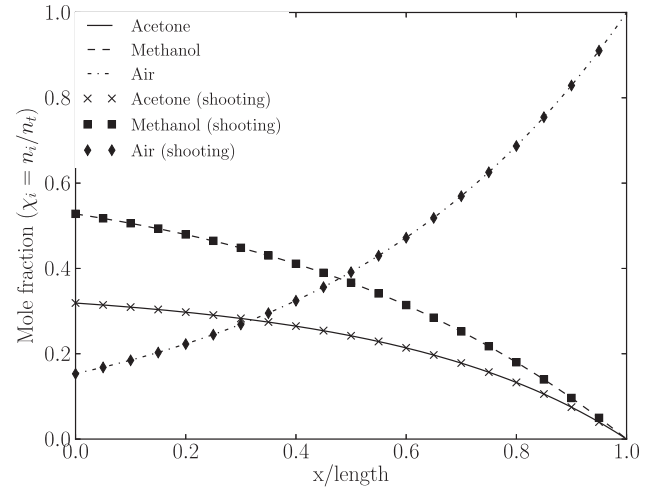
**Fig. 3.** Setup of the Stefan tube test case.

$BC_1$  - inlet:] At the bottom of the tube, we assume that species 3 cannot be dissolved in the mixture of species 1 and 2, i.e. no mole flux,  $\mathbf{N}_3 = 0$ , resulting in  $\mathbf{v}_3 = 0$ , i.e. no fluid flow for species 3. Furthermore, we assume constant mole fraction  $\chi_1 = \chi_1^{BC_1} = 0.319$ ,  $\chi_2 = \chi_2^{BC_1} = 0.528$  for species 1 and 2 (due to the infinitely large reservoir below  $BC_1$ ).

$BC_2$  and  $BC_3$  - periodic:  $BC_2$  and  $BC_3$  represent periodic boundary conditions.

$BC_4$  - outlet: At the top of the tube, we assume zero mole fraction of species 1 and 2 (infinitely large reservoir of species 3 alone). To avoid division by zero, we use a small parameter  $\kappa$  and impose  $\chi_1, \chi_2 = \kappa$ . Furthermore, we assume a given mole fraction  $\chi_3 = \chi_3^{BC_4} = 1 - 2\kappa$  for species 3.

In addition, we assume zero kinematic pressure difference between  $BC_1$  and  $BC_4$  (i.e. we prescribe a given, arbitrary kinematic pressure there) such that no convective species transport should occur (assuming that the initial condition has zero velocity). The



**Fig. 4.** Numerical Stefan tube experiment. The concentration profiles are obtained by running the simulation until steady state on a mesh of 240 elements in  $y$  direction.

simulation starts with the initial conditions

$$n(\mathbf{x}) = n^C$$

$$n_1(\mathbf{x}) = \chi_1^{BC_1} n^C - \frac{\chi_1^{BC_1} n^C}{L_y} y + \kappa$$

$$n_2(\mathbf{x}) = \chi_2^{BC_1} n^C - \frac{\chi_2^{BC_1} n^C}{L_y} y + \kappa$$

$$n_3(\mathbf{x}) = \frac{n^C}{L_y} y - 2\kappa$$

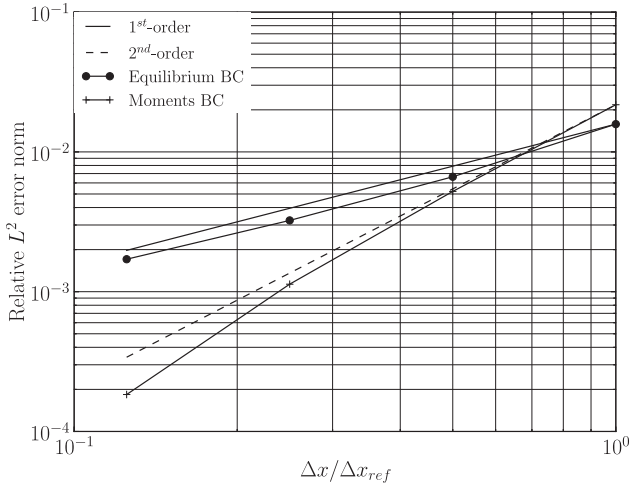
$$\mathbf{v}_1(\mathbf{x}) = \mathbf{v}_2(\mathbf{x}) = \mathbf{v}_3(\mathbf{x}) = 0.$$

The Schmidt number for this setup is

$$Sc = 3.09.$$

This numerical experiment is solved by the BGK multicomponent lattice Boltzmann scheme (11) on D2Q9 stencil. A complete list of all parameters is given in Appendix C.

To validate the accuracy of our model we use a shooting method [40] to solve Eq. (2). Fig. 4 shows the concentration profiles for the proposed LBM and the shooting method at steady state. As can be seen, both results are in good agreement to each



**Fig. 5.** Error plot of mole fraction profile of species 3 with moments and equilibrium based boundary conditions.

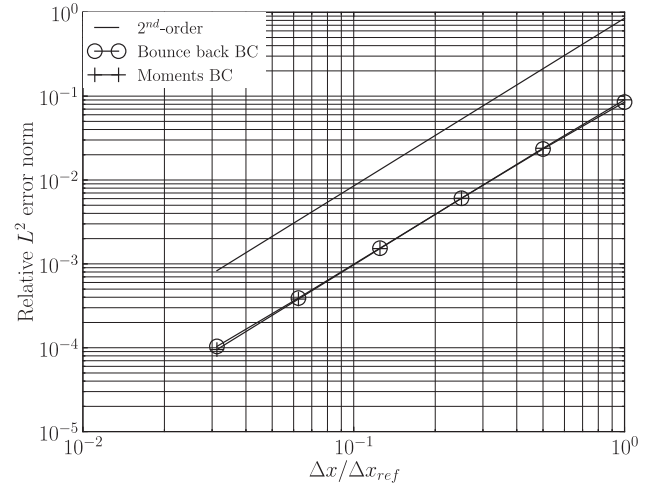
other. Furthermore, the presented moment based boundary conditions, with imposed mole fraction for species 1 and 2, and no flux for species 3 at  $BC_1$  and imposed mole fraction for all species at  $BC_4$ , provide sufficient accuracy to recover the correct concentration profiles. In addition the results point out that our model is able to recover the full Maxwell–Stefan equations. In addition, the norm of the relative  $L_2$  error of the concentration profile of species 3 is shown in Fig. 5 for moment based boundary conditions. For convenience we also give the convergence curve for equilibrium based boundary conditions [41], which are expected to be rough and of first order accuracy (for concentration and velocity) in the diffusive asymptotic limit.

Fig. 5 shows second order convergence for moments based boundary conditions and first order convergence for the equilibrium based boundary conditions in the Stefan tube setup, which is in good agreement with the theoretical analysis. Overall, moment based boundary conditions deliver more accurate simulation results.

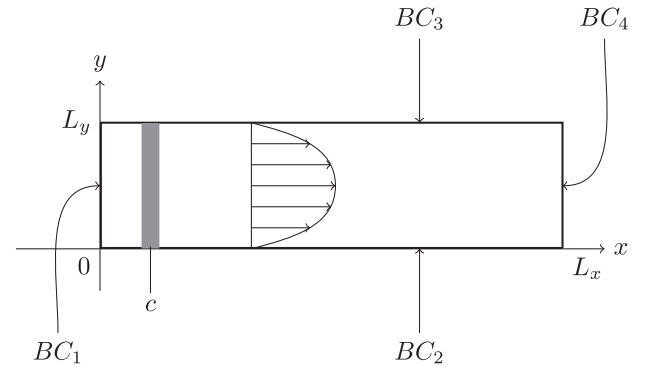
## 5.2. Taylor dispersion

In this section we investigate convection dominated multicomponent flow problems. We consider a simple fluid channel of dimensions  $[0; L_x] \times [0; L_y]$  with a laminar Poiseuille flow profile for the mixture. The direction of flow is the positive  $x$  direction. Before we look at the actual Taylor dispersion experiment, we briefly consider a channel homogeneously filled with two distinct species. For this experiment the exact solution is given by the Poiseuille flow profile, thus this setup allows an analysis of the numerical accuracy of the LB method and its boundary conditions. Fig. 6 shows the numerically obtained error convergence of the velocity field. We observe second order convergence for moment and bounce-back based boundary conditions, which is in perfect agreement with the theoretical predictions.

After this short parenthesis on the accuracy of boundary conditions, we now consider the actual Taylor dispersion experiment. The Taylor dispersion is a test case that is well known in the field of multicomponent flows [42,43] and is widely used to test mass transport models. It includes laminar hydrodynamics as well as pure diffusion phenomena. Initially, the first of two species is equally distributed in the flow channel, while the second one has non-zero concentration only in a small stripe perpendicular to the flow direction (i.e. aligned in the  $y$  direction). The setup for the Taylor dispersion experiment is shown in Fig. 7. We assume the



**Fig. 6.** Error plot of velocity for Poiseuille flow with moments and bounce back based boundary conditions.



**Fig. 7.** Setup of the Taylor dispersion experiment for the numerical simulations. The gray region depicts the region with high concentration of species 2. The center of the stripe is  $x = c$  and its width is  $\Delta c$ . The velocity distribution is a laminar Poiseuille pipe flow profile.

following initial conditions (with  $k \in \mathbb{N}$  and  $\kappa = -\ln(10^{-12})$ ):

$$n_1(\mathbf{x}) = n^c - n_2(\mathbf{x})$$

$$n_2(\mathbf{x}) = \kappa + 0.1 \cdot n^c \cdot \exp\left(-\kappa \left(\frac{x-c}{\Delta c/2}\right)^{2k}\right)$$

$$\mathbf{v}(\mathbf{x}) = \frac{\Delta p}{L_x} \cdot \frac{L_y^2}{2\nu} \cdot \left(1 - \left(\frac{y-L_y/2}{L_y/2}\right)^2\right)$$

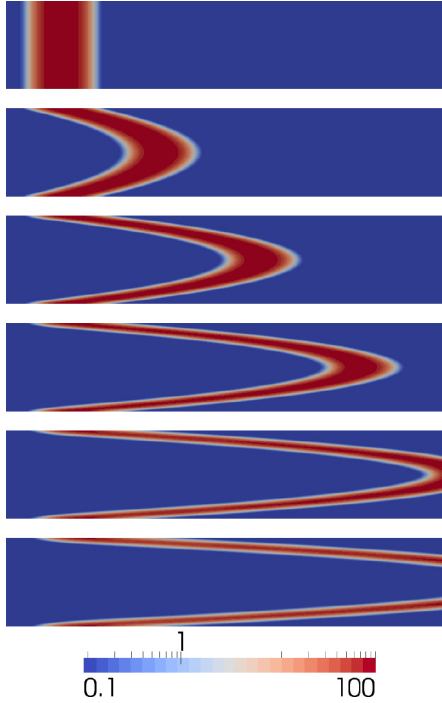
$$\mathbf{v}_1(\mathbf{x}) = \mathbf{v}_2(\mathbf{x}) = \mathbf{v}(\mathbf{x})$$

$$p(\mathbf{x}) = \rho \cdot \left(-\frac{\Delta p}{L_x} \cdot x + \Delta p\right)$$

The constant  $\kappa$  represents again a small parameter to avoid divisions by zeros,  $k$  determines the sharpness of the transition zone of the concentration stripe for species 2 (larger  $k$  leads to sharper profile),  $c$  the center of the concentration stripe and  $\Delta c$  the total width of the stripe, cf. Fig. 7. The pressure difference between inlet and outlet is controlled by  $\Delta p$ . Without loss of generality we assume zero kinematic pressure at the end of the channel. Furthermore, we impose the following boundary conditions at the walls (cf. Fig. 7):

$BC_1$  - inlet: For the inlet boundary we prescribe the species velocities  $\mathbf{v}_1 = \mathbf{v}_2 = \mathbf{v}$  (Poiseuille profile) and the species num-





**Fig. 8.** Temporal evolution of the diffusion of the second species (isocontours of the concentration) in the numerical Taylor dispersion experiment for  $t = 0, 10, 20, 30, 40, 50$  [s] (from top to bottom). The simulation was carried out on a two-dimensional mesh with 800,000 elements (i.e. 400 elements along the height of the channel).

ber densities  $n_1, n_2$ . The mixture kinematic pressure  $p/\rho$  is extrapolated.

$BC_2$  and  $BC_3$  - no-slip wall: At the solid walls of the channel we assume zero velocities, i.e.  $\mathbf{v}_1 = \mathbf{v}_2 = \mathbf{v} = 0$ . This type of boundary condition is covered by the simple bounce-back rule.

$BC_4$  - outlet: At the outlet a given kinematic pressure  $p/\rho$  is prescribed (without loss of generality we assume  $p/\rho = 0$ ). All other quantities ( $\rho_1, \rho_2, \mathbf{v}_1, \mathbf{v}_2$ ) are extrapolated.

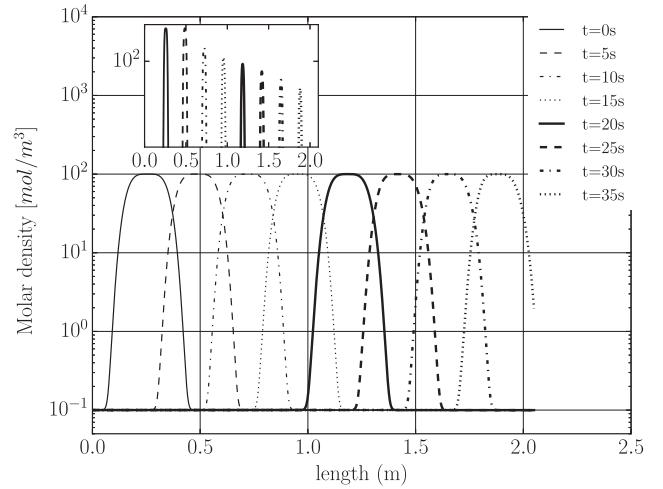
The Reynolds and Schmidt number for this setup are

$$Re = 26.77$$

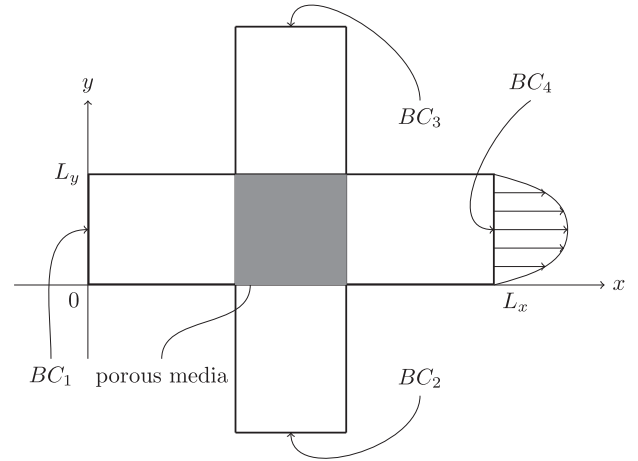
$$Sc = 85.18.$$

The test case is solved by the MRT multicomponent lattice Boltzmann scheme (12) on D2Q9 stencil. A complete list of all parameters for the numerical setup is given in Appendix C.

Due to the parabolic velocity profile the second species is transported much faster along the center than at the channel boundaries (where zero velocity is assumed). Fig. 8 shows the results of the numerical experiment by means of isocontour lines for the number density of the second species. The concentration profile follows the parabolic velocity profile of the background flow and the no-slip boundary condition along the horizontal boundaries is accurately resolved. Notice that the concentration profile leaves the channel along outlet boundary without numerical reflections. The evolution of the concentration profile for the second species along the center line for different points in time is shown in Fig. 9. Both results show that the expected advection-diffusion mechanism is accurately recovered by the LB scheme. The proposed boundary conditions resolve no-slip as well as in- and outlet conditions accurately.



**Fig. 9.** Diffusion of the second species in the numerical Taylor dispersion experiment along the axial line of the channel over time. The subplot shows the decay of the concentration profile peaks over time.



**Fig. 10.** Setup of the porous media test case of Section 5.3. The length of the porous medium is denoted by  $c$ . The outlet velocity profile is a laminar Poiseuille pipe flow profile.

### 5.3. Multicomponent mass transport in porous media

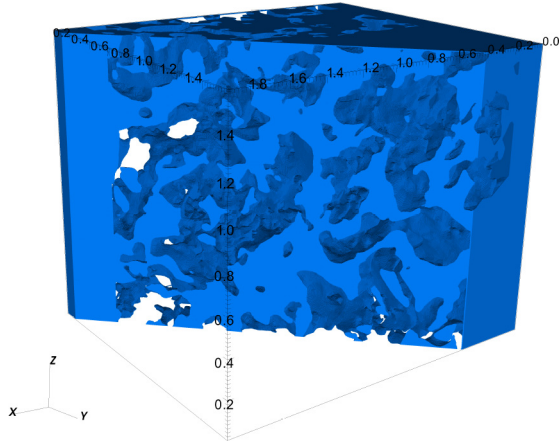
In this section we consider multicomponent mass transport of three species in a three-dimensional complex porous media. It is demonstrated that the multicomponent LB scheme maintains the efficiency of standard LB schemes for very complex geometries. The geometrical setup of the test case is shown in Fig. 10. A plot of the clipped mesh around the porous media is shown in Fig. 11. In total, the mesh consists of more than 450 million fluid elements and complex boundary shapes occur inside the porous media. The boundary conditions are assigned as follows:

$BC_1, BC_2, BC_3$  - inlet: For the inlet boundary  $BC_i$ , we prescribe the species mole fraction  $\chi_i$ . For all other species  $k \neq i$  we have no-flux boundary conditions.

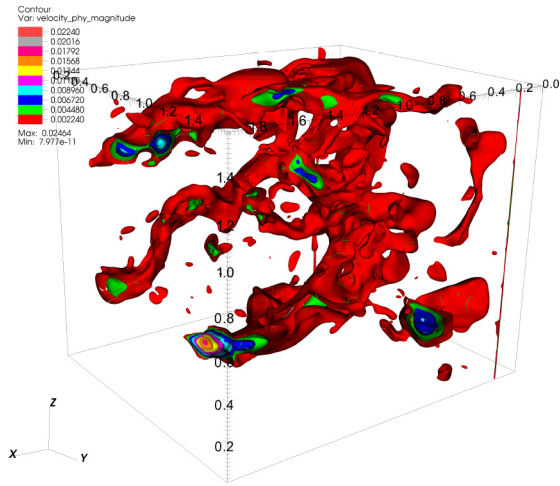
$BC_4$  - outlet: At the outlet a given mixture velocity profile  $\mathbf{v}$  is prescribed. Furthermore, we assume that  $\mathbf{v}_i = \mathbf{v}$  for all  $i$ .

other: All other boundaries of the channels and the porous media are no-slip boundary conditions.

Initially, we fill the channel in  $x$  direction (including the porous media) with species 1. The attached channels in  $y$  direction (excluding the porous media) are filled with species 2 and 3, respectively. Due to the concentration gradients in the porous media the



**Fig. 11.** Clipped mesh of the porous media test case of Section 5.3. Overall the mesh consists of more than 450 million fluid elements. Overall more than 1.2 billion species elements are handled by more than 20,400 cores.



**Fig. 12.** Clipped isosurfaces of mixture velocity magnitude of the porous media test case of Section 5.3 (same clipping plane and viewing angle as in Fig. 11).

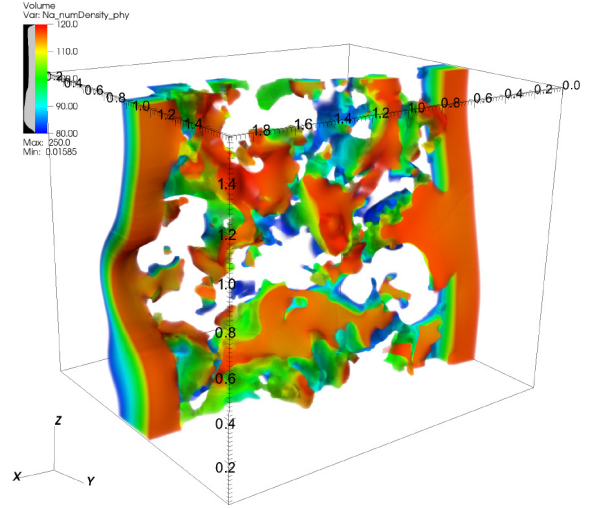
species start to diffuse, while at the same time the mean flow convects the species in the positive  $x$  direction. In combination with the complex structure of the porous media, a complex mass transport evolves over time. The Reynolds and Schmidt number for this setup are

$$Re = 0.02$$

$$Sc = 35668.$$

The test case is solved by the MRT multicomponent lattice Boltzmann scheme (12) on D3Q19 stencil. Furthermore, we improved the accuracy of the boundary conditions within the porous media by using the  $q$ -value bounce-back rules (22).

Fig. 12 shows the mixture velocity isosurfaces within the porous media at the final simulation time. The flow profile of the mixture follows the shape of the porous media. Fig. 13 shows the concentration distribution of the second species in the porous medium at the same point in time. According to the initially imposed concentration gradient the species diffuses into the  $y$ -direction. Overall the simulation of this test case has more than 1.2 billion species elements. We used 20,480 cores to solve the test case in less than 48 hours of total computing time. Setups similar to test case of Section 5.3 appear in many engineering applications, e.g. fuel cells, electrodialysis [1]. The simulation results demonstrate that



**Fig. 13.** Volume rendering of concentration distribution (species 2) of the porous media test case of Section 5.3 (same viewing angle as in Fig. 11).

the LBM and the proposed boundary conditions are well suited for complex simulation setups.

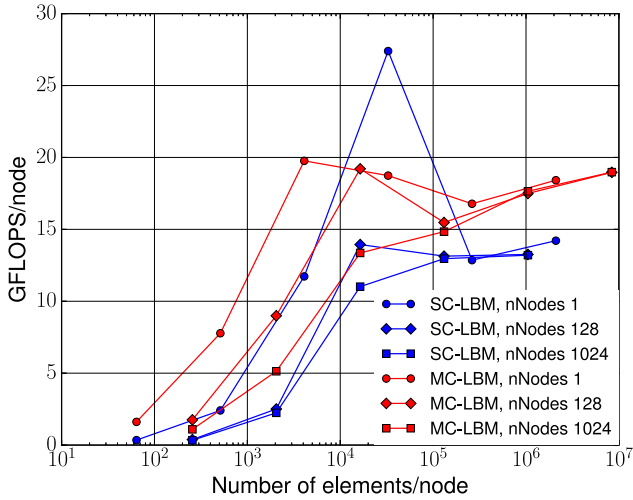
## 6. Computational aspects, parallelization and scalability

The multicomponent LB model presented in this paper was implemented in our in-house lattice Boltzmann solver, *Musubi* [24] which is part of the octree based massively parallel framework [25]. This model can be implemented in any existing LB frameworks like waLBerla [44] or VirtualFluids [45] with minimal effort and at the same time maintaining the advantages of those framework. In this section we present computational performance results: We achieved a high sustained performance and it is demonstrated that the model scales excellently. The results are compared to the single component LB implementation in *Musubi* [24] and it was optimized with same optimization techniques used in [46]. Performance and scalability results show that the model inherits the computational advantages from the standard LB method.<sup>1</sup>

To measure the scalability of our implementation, we consider a fully periodic, cubic simulation domain, i.e. the full octree. The number of elements is given by  $n_{elems} = 8^{level}$  where  $level \in \mathbb{N}$ . Furthermore, we use  $n_{procs} = 8^k$  number of processes with  $k \in \mathbb{N}$  for our simulations, such that the simulation domain is decomposed into self-similar domains independent of the number of elements and processes. This allows us to study the pure scalability of the implementation (without any influence of the actual domain decomposition). Although this setup seems to be of no relevance for complex geometries at first glance, it represents the worst case scenario for the scalability check as the ratio of communication surface and computation volume is maximized. Actually, with complex geometries scalability improves due to a lower amount of communication.

To measure the relative performance of a computing node for different  $n_{elems}$ ,  $n_{procs}$  combinations, we define GFLOPS/node as the number of giga( $10^9$ ) floating point operations per second per node. Ideally, the GFLOPS should be independent of the number of processes and the problem size. The measured data on the Hermit system at HLRS Stuttgart (Cray XE6) for both single component LBM (SC-LBM) and multicomponent LBM (MC-LBM) are shown in Fig. 14. The system consists of 3552 nodes, each node has 32 cores.

<sup>1</sup> The presented results are obtained for a three dimensional setting with the D3Q19 stencil.

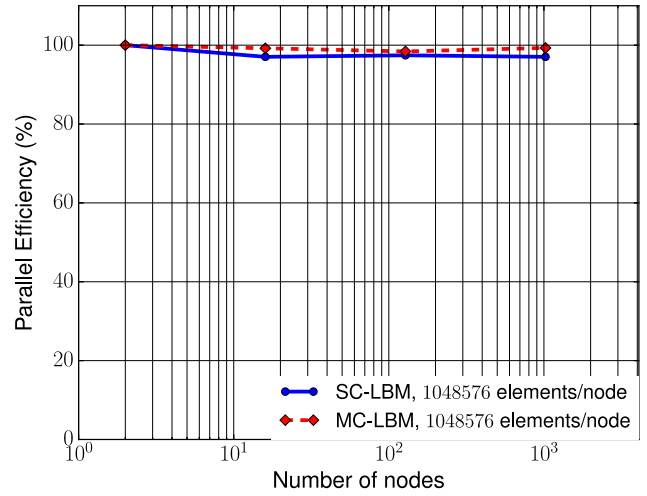


**Fig. 14.** Performance map of the multicomponent model for a setup with three species and single component LBM on the Hermit system at HLRS (Cray XE6). Each node has 32 cores, hence the largest computation was carried out on more than 32,000 cores.

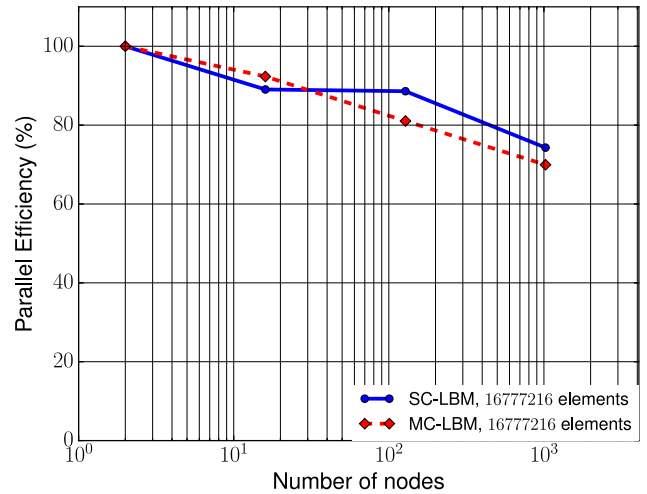
Hence the largest computation was carried out on 32,768 cores i.e. 1024 nodes. In general, the graph can be separated into two regions: In the first one, where small problem sizes per node are considered, the cache is accelerating the computation (for very small problem sizes the communication overhead is actually decreasing the performance). In the second one, where problem sizes are larger than the cache, a continuous access of the main memory can be observed, cf. Fig. 14. The performance Fig. 14, both LBM shows the same trend i.e. there is a gain in performance when problem size fit into cache region and after cache region the performance gets saturated. The cache regions of MC-LBM are shifted to left of SC-LBM due to its increase in memory requirement. Also, the cache regions of MC-LBM are smaller due to increase in communication load per element. Nevertheless, the GFLOPS of MC-LBM are greater than SC-LBM since MC-LBM performs more floating point operations per lattice element per process clock frequency.

In addition, the performance map delivers strong and weak scalability information. In a weak scaling, the problem size per process/node is fixed and the number of processes is varied, leading to connections at a fixed point on the horizontal axis (in the vertical direction) of the performance map, cf. Fig. 14. The smaller the vertical distance of these points the better the weak scalability. A separate graph for the weak scaling with 1 million elements per compute node (corresponding to 32,768 elements per process) is provided in Fig. 15, depicting a very good parallel efficiency above 99.3 and 97.04% for MC-LBM and SC-LBM respectively.

The strong scalability is obtained by a diagonal connection of points in the performance map, Fig. 14, in the following sense: The total problem size is fixed and in each step of a strong scaling we increase the number of processes by a factor of eight. So, for the strong scaling we start at a certain point of the performance map and go to the next point with eight times more processes and an eight times smaller problem size per processes. The resulting line represents the strong scaling graph. A dedicated plot of the strong scaling is shown in Fig. 16 pointing out that both presented model and standard LB model provides more than 70% of parallel efficiency whenever the problem size per process does not decrease significantly below 512 elements. (please notice that the last point in the strong scaling graph corresponds to 512 elements per process for the 16 million element line as 1024 compute nodes correspond to 32,768 processes). The strong scaling of both mod-



**Fig. 15.** Weak scalability of the presented model (three species) and single component LB for 1 million elements per node on the Hermit system at HLRS (Cray XE6). 1 node = 32 cores.



**Fig. 16.** Strong scalability of the presented model (three species) for 16 and 134 million elements on the Hermit system at HLRS (Cray XE6). 1 node = 32 cores.

els decreases with increase in number of nodes due to increase in communication between processes. For SC-LBM with 4096 elements per process on 4096 processes (128 nodes), there is no drop performance due to cache effect. Fig. 16 shows no such cache effect for MC-LBM since to fit into cache it requires roughly 200 elements per process. Overall, the strong scaling is very good and allows us to speed up realistic simulation setups with million of elements by scaling them down to small element per process ratios.

A single lattice updated per species involves 261 floating point operations. In comparison, a single species BGK lattice Boltzmann implementation requires 169 floating point operations per lattice update, showing that the additional overhead introduced by the multicomponent model is quite small. Fixing at total problem size of 16 million elements and 128 nodes, a single species BGK lattice Boltzmann implementation updates 78.44 million lattices per second per node, while our multicomponent model implementation (with 3 species) updates 22.34 million lattices per second per node (corresponding to 67.02 million species-lattice updates per second per node).

All in all, our multicomponent lattice Boltzmann implementation achieves a sustained performance of around 6% (with re-

spect to the theoretical peak performance of the Hermit system), i.e. 17.66 (with 18087.3 GFLOPS on 1024 nodes) of 294 GFLOPS per node, which is a high value for computational fluid dynamics codes. The standard LB method achieves 13.2 GFLOPS per node, i.e. 4.5% sustained performance. Together with the good scalability, the measured performance emphasizes that the multicomponent LB model is well suited for massively parallel simulations. Furthermore, it shows that multicomponent LB model inherits the efficiency from the standard LB method.

## 7. Conclusion

In this paper we studied the computational robustness and performance of a lattice Boltzmann scheme for multicomponent flows. We showed that the cell-local equation system of the implicit-to-explicit variable transformation is well-posed (uniqueness and existence) under weak constraints on the dimensionless Maxwell–Stefan diffusivities. We analyzed moment and bounce-back based boundary conditions for complex geometries. These boundary conditions maintain the locality of the LB scheme and are robust for complex simulation setups. We demonstrated the capabilities of the model and its boundary conditions in various numerical experiments. Furthermore, we demonstrated that the multicomponent model is ideally suited for massively parallel simulations in complex geometries. The obtained parallel performance and scalability are excellent and maintain the computational advantages of the standard LB method.

## Acknowledgment

This work was funded by the German Federal Ministry of Education and Research (Bundesministerium für Bildung und Forschung, BMBF) in the framework of the HPC software initiative in the project HISEEM. PA acknowledges the support of the Italian Ministry of Research (FIRB grant RBFR10VZUG). We thank the Gauss Alliance of German Supercomputing Centers for the provided computing time. We are grateful for the ongoing kind support by HLRS and Cray in Stuttgart and LRZ in Munich.

## Appendix A. MRT matrices for the D2Q9 lattice

In this section, the MRT matrices [20] for the two-dimensional D2Q9 lattice are presented. The moment matrix  $\mathbf{M}$  is given by

$$\mathbf{M} = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & -1 & 0 & 1 & -1 & -1 & 1 \\ 0 & 0 & 1 & 0 & -1 & 1 & 1 & -1 & -1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & 1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix},$$

and

$$\mathbf{A}_k = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \lambda_k^\delta & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \lambda_k^\delta & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{\lambda_k^\zeta + \lambda_k^v}{2} & \frac{\lambda_k^\zeta - \lambda_k^v}{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{\lambda_k^\zeta - \lambda_k^v}{2} & \frac{\lambda_k^\zeta + \lambda_k^v}{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \lambda_k^v & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

denotes the relaxation matrix for the specific moments. The relaxation parameters are given by

$$\lambda_k^\delta = \frac{KB}{\rho}, \lambda_k^v = \frac{1}{3\nu}, \lambda_k^\zeta = \frac{2 - \phi_k}{3\zeta}. \quad (\text{A.1})$$

## Appendix B. MRT matrices for the D3Q19 lattice

In this section, we present the MRT matrices for the three-dimensional D3Q19 lattice. The moment matrix  $\mathbf{M}$  is given by

$$\mathbf{M} = \begin{pmatrix} 1 \\ \mathbf{U}^1 \\ \mathbf{U}^2 \\ \mathbf{U}^3 \\ \mathbf{U}^1 * \mathbf{U}^1 \\ \mathbf{U}^2 * \mathbf{U}^2 \\ \mathbf{U}^3 * \mathbf{U}^3 \\ \mathbf{U}^1 * \mathbf{U}^2 \\ \mathbf{U}^1 * \mathbf{U}^3 \\ \mathbf{U}^2 * \mathbf{U}^3 \\ \mathbf{U}^1 * \mathbf{U}^1 * \mathbf{U}^2 \\ \mathbf{U}^1 * \mathbf{U}^2 * \mathbf{U}^2 \\ \mathbf{U}^1 * \mathbf{U}^1 * \mathbf{U}^3 \\ \mathbf{U}^1 * \mathbf{U}^3 * \mathbf{U}^3 \\ \mathbf{U}^2 * \mathbf{U}^2 * \mathbf{U}^3 \\ \mathbf{U}^2 * \mathbf{U}^3 * \mathbf{U}^3 \\ \mathbf{U}^1 * \mathbf{U}^1 * \mathbf{U}^2 * \mathbf{U}^2 \\ \mathbf{U}^1 * \mathbf{U}^1 * \mathbf{U}^3 * \mathbf{U}^3 \\ \mathbf{U}^2 * \mathbf{U}^2 * \mathbf{U}^3 * \mathbf{U}^3 \end{pmatrix} \quad (\text{B.1a})$$

In the previous definition  $\mathbf{U}^i$  denotes the  $\mathbb{R}^{1 \times 19}$ -vector with the  $i$ th components of the lattice velocities  $\mathbf{u}^m$  ( $m = 0, \dots, 18$ ). The operator “\*” denotes component-wise vector multiplication. The relaxation matrix  $\mathbf{A}_k$  for species  $k$  is defined as the following block-matrix

$$\mathbf{A}_k = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & \mathbf{A}_k & 0 & 0 & 0 \\ 0 & 0 & \mathbf{B}_k & 0 & 0 \\ 0 & 0 & 0 & \mathbf{C}_k & 0 \\ 0 & 0 & 0 & 0 & \mathbf{D}_k \end{pmatrix}. \quad (\text{B.1b})$$

The matrices  $\mathbf{A}_k, \mathbf{B}_k, \mathbf{C}_k, \mathbf{D}_k$  are given by

$$\mathbf{A}_k = \text{diag}\{\lambda_k^\delta, \lambda_k^\delta, \lambda_k^\delta\} \in \mathbb{R}^{3 \times 3} \quad (\text{B.2a})$$

and

$$\mathbf{B}_k = \begin{pmatrix} \frac{\lambda_k^\zeta + 2\lambda_k^v}{3} & \frac{\lambda_k^\zeta - \lambda_k^v}{3} & \frac{\lambda_k^\zeta - \lambda_k^v}{3} \\ \frac{\lambda_k^\zeta - \lambda_k^v}{3} & \frac{\lambda_k^\zeta + 2\lambda_k^v}{3} & \frac{\lambda_k^\zeta - \lambda_k^v}{3} \\ \frac{\lambda_k^\zeta - \lambda_k^v}{3} & \frac{\lambda_k^\zeta - \lambda_k^v}{3} & \frac{\lambda_k^\zeta + 2\lambda_k^v}{3} \end{pmatrix} \in \mathbb{R}^{3 \times 3} \quad (\text{B.2b})$$

and

$$\mathbf{C}_k = \text{diag}\{\lambda_k^v, \lambda_k^v, \lambda_k^v\} \in \mathbb{R}^{3 \times 3} \quad (\text{B.2c})$$

and

$$\mathbf{D}_k = \text{diag}\{1, 1, 1, 1, 1, 1, 1, 1, 1\} \in \mathbb{R}^{9 \times 9}. \quad (\text{B.2d})$$

## Appendix C. Simulation parameters

Table C.1 provides all parameters for the numerical experiments shown in Section 5.



**Table C1**

Simulation parameters for the test cases of Section 5.

Param./Sec.	Section 5.1	Section 5.2	Section 5.3
$L_x[m]$	–	2.05	–
$L_y[m]$	0.238	0.41	–
$n^c[mol/m^3]$	1000	1000	1000
$m_1[kg/mol]$	$58.08 \cdot 10^{-3}$	$18.01 \cdot 10^{-3}$	$18.01 \cdot 10^{-3}$
$m_2[kg/mol]$	$32.04 \cdot 10^{-3}$	$22.98 \cdot 10^{-3}$	$22.98 \cdot 10^{-3}$
$m_3[kg/mol]$	$28.86 \cdot 10^{-3}$	–	$32.45 \cdot 10^{-3}$
$B_{1,2}[s/m^2]$	$(1/8.48) \cdot 10^6$	$(1/1.315) \cdot 10^6$	$(1/1.315) \cdot 10^9$
$B_{1,3}[s/m^2]$	$(1/13.72) \cdot 10^6$	–	$(1/2.097) \cdot 10^9$
$B_{2,3}[s/m^2]$	$(1/19.91) \cdot 10^6$	–	$(1/2.954) \cdot 10^{11}$
$\kappa$	0.0001	0.0001	–
$c[m]$	–	2.05/8	–
$\Delta c[m]$	–	2.05/4	–
$\Delta p[m^2/s^2]$	–	$10^{-4}$	–
$k$	–	2	–

## References

- [1] Novaresio V, García-Camprubí M, Izquierdo S, Asinari P, Fueyo N. An open-source library for the numerical modeling of mass-transfer in solid oxide fuel cells. *Comput Phys Commun* 2012;183(1):125–46. doi:10.1016/j.cpc.2011.08.003. URL <http://linkinghub.elsevier.com/retrieve/pii/S001046551100275X>.
- [2] Buoni M, Petzold L. An algorithm for simulation of electrochemical systems with surface-bulk coupling strategies. *J Comput Phys* 2010;229(2):379–98. doi:10.1016/j.jcp.2009.09.032. URL <http://linkinghub.elsevier.com/retrieve/pii/S0021999109005245>.
- [3] Chen S, Doolen GD. Lattice Boltzmann method for fluid flows. *Annu Rev Fluid Mech* 1998;30(1):329–64. doi:10.1146/annurev.fluid.30.1.329. URL <http://www.annualreviews.org/doi/abs/10.1146/annurev.fluid.30.1.329>.
- [4] Bernsdorf J, Jaekel U, Zeiser T, Takei T, Matsumoto H, Nishizawa K. Lattice Boltzmann simulation and visualisation of adsorption processes in complex geometries. *Comput Sci - ICCS 2003* 2003;665–6. URL <http://www.springerlink.com/index/MD6U8VFJ787E3EX.pdf>.
- [5] Aidun CK, Clausen JR. Lattice-Boltzmann method for complex flows. *Annu Rev Fluid Mech* 2010;42(1):439–72. doi:10.1146/annurev-fluid-121108-145519. URL <http://www.annualreviews.org/doi/abs/10.1146/annurev-fluid-121108-145519>.
- [6] Geller S, Krafczyk M, Tolke J, Turek S, Hron J. Benchmark computations based on lattice-Boltzmann, finite element and finite volume methods for laminar flows. *Comput Fluids* 2006;35(8–9):888–97. doi:10.1016/j.compfluid.2005.08.009. URL <http://linkinghub.elsevier.com/retrieve/pii/S0045793005001519>.
- [7] Succi S. *The lattice Boltzmann equation for fluid dynamics and beyond*, 1. Clarendon Press; 2001.
- [8] Flekkoy EG. Lattice bhatnagar-gross-krook models for miscible fluids. *Phys Rev E* 1993;47:4247–57. doi:10.1103/PhysRevE.47.4247. URL <http://link.aps.org/doi/10.1103/PhysRevE.47.4247>.
- [9] Shan X, Chen H. Lattice Boltzmann model for simulating flows with multiple phases and components. *Phys Rev E* 1993;47:1815–19. doi:10.1103/PhysRevE.47.1815. URL <http://link.aps.org/doi/10.1103/PhysRevE.47.1815>.
- [10] Shan X, Chen H. Simulation of nonideal gases and liquid-gas phase transitions by the lattice boltzmann equation. *Phys Rev E* 1994;49:2941–8. doi:10.1103/PhysRevE.49.2941. URL <http://link.aps.org/doi/10.1103/PhysRevE.49.2941>.
- [11] Shan X. Multicomponent lattice boltzmann model from continuum kinetic theory. *Phys Rev E* 2010;81:045701. doi:10.1103/PhysRevE.81.045701. URL <http://link.aps.org/doi/10.1103/PhysRevE.81.045701>.
- [12] Orlandini E, Swift MR, Yeomans JM. A lattice Boltzmann model of binary-fluid mixtures. *Europhys Lett* 1995;32(6):463. URL <http://stacks.iop.org/0295-5075/32/i=6/a=001>.
- [13] Osborn WR, Orlandini E, Swift MR, Yeomans JM, Banavar JR. Lattice Boltzmann study of hydrodynamic spinodal decomposition. *Phys Rev Lett* 1995;75:4031–4. doi:10.1103/PhysRevLett.75.4031. URL <http://link.aps.org/doi/10.1103/PhysRevLett.75.4031>.
- [14] Swift MR, Orlandini E, Osborn WR, Yeomans JM. Lattice Boltzmann simulations of liquid-gas and binary fluid systems. *Phys Rev E* 1996;54:5041–52. doi:10.1103/PhysRevE.54.5041. URL <http://link.aps.org/doi/10.1103/PhysRevE.54.5041>.
- [15] Lamura A, Gonnella G, Yeomans JM. A lattice Boltzmann model of ternary fluid mixtures. *EPL (Europhys Lett)* 1999;45:314–20. URL <http://iopscience.iop.org/0295-5075/45/3/314>.
- [16] Luo L-S, Girimaji SS. Theory of the lattice Boltzmann method: two-fluid model for binary mixtures. *Phys Rev E* 2003;67:036302. doi:10.1103/PhysRevE.67.036302. URL <http://link.aps.org/doi/10.1103/PhysRevE.67.036302>.
- [17] Xu A. Two-dimensional finite-difference lattice Boltzmann method for the complete Navier-Stokes equations of binary fluids. *Europhys Lett* 2005;69(2):214. URL <http://stacks.iop.org/0295-5075/69/i=2/a=214>.
- [18] Facin PC, Philippi PC, dos Santos LOE. A non-linear lattice-boltzmann model for ideal miscible fluids. *Future Generation Comp Syst* 2004;20(6):945–9. URL <http://dx.doi.org/10.1016/j.future.2003.12.006>.
- [19] Asinari P. Viscous coupling based lattice Boltzmann model for binary mixtures. *Phys Fluids* 2005;17(6). doi:10.1063/1.1927567. URL <http://link.aps.org/link/?PHF/17/067102/1>.
- [20] Asinari P. Semi-implicit-linearized multiple-relaxation-time formulation of lattice Boltzmann schemes for mixture modeling. *Phys Rev E* 2006;73:056705. doi:10.1103/PhysRevE.73.056705. URL <http://link.aps.org/doi/10.1103/PhysRevE.73.056705>.
- [21] Arcidiacono S, Karlin IV, Mantzaras J, Frouzakis CE. Lattice Boltzmann model for the simulation of multicomponent mixtures. *Phys Rev E* 2007;76:046703. doi:10.1103/PhysRevE.76.046703. URL <http://link.aps.org/doi/10.1103/PhysRevE.76.046703>.
- [22] Zudrop J, Roller S, Asinari P. Lattice Boltzmann scheme for electrolytes by an extended Maxwell-Stefan approach. *Phys Rev E* 2014;89:053310. doi:10.1103/PhysRevE.89.053310. URL <http://link.aps.org/doi/10.1103/PhysRevE.89.053310>.
- [23] Asinari P. Multiple-relaxation-time lattice Boltzmann scheme for homogeneous mixture flows with external force. *Phys Rev E* 2008;77:056706. doi:10.1103/PhysRevE.77.056706. URL <http://link.aps.org/doi/10.1103/PhysRevE.77.056706>.
- [24] Hasert M, Masilamani K, Zimny S, Klimach H, Qi J, Bernsdorf J, et al. Complex fluid simulations with the parallel tree-based lattice Boltzmann solver musubi. *J Comput Sci* 2013. <http://dx.doi.org/10.1016/j.jocs.2013.11.001>. URL <http://www.sciencedirect.com/science/article/pii/S187750313001270>.
- [25] Roller S, Bernsdorf J, Klimach H, Hasert M, Harlacher D, Cakircali M, et al. An adaptable simulation framework based on a linearized octree. In: Resch M, Wang X, Bez W, Focht E, Kobayashi H, Roller S, editors. *High Performance Computing on Vector Systems 2011*. Springer Berlin Heidelberg; 2012. p. 93–105. ISBN 978-3-642-22244-3. doi:10.1007/978-3-642-22244-3\_7.
- [26] Zudrop J, Klimach H, Hasert M, Masilamani K, Roller S. A fully distributed CFD framework for massively parallel systems. *Cray User Group* 2012. Stuttgart, Germany; 2012.
- [27] Bird RB, Stewart WE, Lightfoot EN. *Transport phenomena*. John Wiley & Sons; 1976.
- [28] Taylor R, R K. *Multicomponent mass transfer*, 1. Wiley-Interscience; 1993.
- [29] Liu X, Vlught TJ, Bardow A. Maxwell-Stefan diffusivities in liquid mixtures: using molecular dynamics for testing model predictions. *Fluid Phase Equilib* 2011;301(1):110–17. doi:10.1016/j.fluid.2010.11.019. URL <http://linkinghub.elsevier.com/retrieve/pii/S0378381210005856>.
- [30] Junk M, Klar A, Luo L. Asymptotic analysis of the lattice Boltzmann equation. *J Comput Phys* 2005;210(2):676–704. doi:10.1016/j.jcp.2005.05.003. URL <http://linkinghub.elsevier.com/retrieve/pii/S0021999105002573>.
- [31] Wolf-Gladrow DA. *Lattice-Gas cellular automata and lattice Boltzmann models: an introduction*. Springer; 2000.
- [32] Taylor R, Kooijman HA. Composition derivatives of activity coefficient models (for the estimation of thermodynamic factors in diffusion). *Chem Eng Commun* 1991;102(1):87–106. doi:10.1080/00986449108910851. URL <http://www.tandfonline.com/doi/abs/10.1080/00986449108910851>.
- [33] He X, Luo L-S. Theory of the lattice boltzmann method: from the Boltzmann equation to the lattice boltzmann equation. *Phys Rev E* 1997;56:6811–17. doi:10.1103/PhysRevE.56.6811. URL <http://link.aps.org/doi/10.1103/PhysRevE.56.6811>.
- [34] Junk M, Yang Z. Pressure boundary condition for the lattice boltzmann method. *Comput Math Appl* 2009;58:922–9. doi:10.1016/j.camwa.2009.02.006. URL <http://linkinghub.elsevier.com/retrieve/pii/S0898122109000972>.
- [35] Yang Z. Analysis of lattice Boltzmann boundary conditions. University Konstanz; 2007. URL <http://kops.ub.uni-konstanz.de/handle/urn:nbn:de:bsz:352-opus-35079>.
- [36] Bennett S, Asinari P, Dellar PJ. A lattice Boltzmann model for diffusion of binary gas mixtures that includes diffusion slip. *Int J Numer Methods Fluids* 2012;69:171–89. doi:10.1002/fld.2549. URL <http://dx.doi.org/10.1002/fld.2549>.
- [37] Sone Y. Asymptotic theory of a steady flow of a rarefied gas past bodies for small Knudsen numbers. In: Gatignol R, Soubbaramayer, editors. *Advances in Kinetic Theory and Continuum Mechanics*. Springer Berlin Heidelberg; 1991. p. 19–31. ISBN 978-3-642-50237-8. doi:10.1007/978-3-642-50235-4\_3.
- [38] Bouzidi M, Firdaouss M, Lallemand P. Momentum transfer of a Boltzmann-lattice fluid with boundaries. *Phys Fluids* 2001;13(11):3452. doi:10.1063/1.1399290. URL <http://scitation.aip.org/content/aip/journal/pof2/13/11/10.1063/1.1399290>.
- [39] Zou Q, He X. On pressure and velocity boundary conditions for the lattice Boltzmann bgk model. *Phys Fluids* 1997;9(6):1591–8. <http://dx.doi.org/10.1063/1.869307>. URL <http://scitation.aip.org/content/aip/journal/pof2/9/6/10.1063/1.869307>.
- [40] Binous H. Solution of a Maxwell-Stefan diffusion problem. *Math-Works; 2004*. URL <https://de.mathworks.com/matlabcentral/fileexchange/6106-solution-of-a-maxwell-stefan-diffusion-problem>.
- [41] Mohamad AA, Succi S. A note on equilibrium boundary conditions in lattice boltzmann fluid dynamic simulations. *Eur Phys J Spec Topics* 2009;171(1):213–21. doi:10.1140/epjst/e2009-01031-9. URL <http://www.springerlink.com/index/10.1140/epjst/e2009-01031-9>.
- [42] Taylor G. Dispersion of soluble matter in solvent flowing slowly through a tube. *Proc R Soc Lond Ser A Math Phys Sci* 1953;219(1137):186–203. doi:10.1098/rspa.1953.0139. URL <http://rspa.royalsocietypublishing.org/content/219/1137/186.abstract>.
- [43] Aris R. On the dispersion of a solute in a fluid flowing through a tube. *Proc R Soc Lond Ser A Math Phys Sci* 1956;235(1200):67–77. doi:10.1098/rspa.1956.0065. URL <http://rspa.royalsocietypublishing.org/content/235/1200/67.abstract>.



- [44] Schornbaum F, Rde U. Massively parallel algorithms for the lattice boltzmann method on nonuniform grids. *SIAM J Scient Comput* 2016;38(2):C96–C126. doi:10.1137/15M1035240. URL <http://dx.doi.org/10.1137/15M1035240>.
- [45] Tölke J, Freudiger S, Krafczyk M. An adaptive scheme using hierarchical grids for lattice Boltzmann multi-phase flow simulations. *Comput Fluids* 2006;35(8–9):820–30. doi:10.1016/j.compfluid.2005.08.010. URL <http://linkinghub.elsevier.com/retrieve/pii/S0045793005001441>.
- [46] Wellein G, Zeiser T, Hager G, Donath S. On the single processor performance of simple lattice Boltzmann kernels 2006;35:910–19. doi:10.1016/j.compfluid.2005.02.008.