MULTISCALE MODEL. SIMUL.
Vol. 10, No. 3, pp. 766–791

NUMERICAL EXTRACTION OF A MACROSCOPIC PDE AND A LIFTING OPERATOR FROM A LATTICE BOLTZMANN MODEL*

YNTE VANDERHOYDONC† AND WIM VANROOSE†

Abstract. Lifting operators play an important role in starting a lattice Boltzmann model from a given initial density. The density, a macroscopic variable, needs to be mapped to the distribution functions, mesoscopic variables, of the lattice Boltzmann model. Several methods proposed as lifting operators have been tested and discussed in the literature. The most famous methods are an analytically found lifting operator, like the Chapman–Enskog expansion, and a numerical method, like the Constrained Runs algorithm, to arrive at an implicit expression for the unknown distribution functions with the help of the density. This paper proposes a lifting operator that alleviates several drawbacks of these existing methods. In particular, we focus on the computational expense and the analytical work that needs to be done. The proposed lifting operator, a numerical Chapman–Enskog expansion, obtains the coefficients of the Chapman–Enskog expansion numerically. Another important feature of the use of lifting operators is found in hybrid models. There the lattice Boltzmann model is spatially coupled with a model based on a more macroscopic description, for example an advection-diffusion-reaction equation. In one part of the domain, the lattice Boltzmann model is used, while in another part, the more macroscopic model is used. Such a hybrid coupling results in missing data at the interfaces between the different models. A lifting operator is then an important tool since the lattice Boltzmann model is typically described by more variables than a model based on a macroscopic partial differential equation.

Key words. lifting operator, missing data, lattice Boltzmann models, macroscopic partial differential equations, hybrid models, Chapman–Enskog expansion, Constrained Runs, numerical Chapman–Enskog expansion

AMS subject classifications. 76M28, 35K45, 35K57

DOI. 10.1137/110842739

1. Introduction.

A lifting operator is, in a multiscale method, an important tool that maps macroscopic variables to microscopic/mesoscopic variables. In kinetic models, for example, a lifting operator will map low order moments, like the density $\rho(\boldsymbol{x}, t)$ that counts the number of particles in a point $\boldsymbol{x} \in D_{\boldsymbol{x}} \subset \mathbb{R}^n$, $n \in \mathbb{N}_0$, to a distribution function $f(\boldsymbol{x}, \boldsymbol{v}, t)$ that counts the number of particles in a point $(\boldsymbol{x}, \boldsymbol{v})$ in phase space where $\boldsymbol{x} \in D_{\boldsymbol{x}} \subset \mathbb{R}^n$ and the velocity $\boldsymbol{v} \in D_{\boldsymbol{v}} \subset \mathbb{R}^n$. For practical applications one uses $n \in \{1, 2, 3\}$ and time $t \geq 0$.

In these problems, the macroscopic level is typically described by a few low order moments and their evolution is simulated by the use of a macroscopic partial differential equation (PDE). For example, the evolution of the density $\rho(\boldsymbol{x}, t)$ can be represented by an advection-diffusion-reaction equation, while the microscopic/mesoscopic level is typically described by a Boltzmann equation that evolves the distribution function $f(\boldsymbol{x}, \boldsymbol{v}, t)$. A lattice Boltzmann model (LBM) is a special discretization of the Boltzmann equation that is used, for example, to simulate complex fluid systems.

Some examples of complex flows for which LBMs are used are flows in complicated geometries and multiphase and turbulent flows. Applications can be found in [3, 25],

*Received by the editors July 29, 2011; accepted for publication (in revised form) April 5, 2012; published electronically July 17, 2012. This work was supported by research project *Hybrid macroscopic and microscopic modelling of laser evaporation and expansion*, G.017008N, funded by "Fonds Wetenschappelijk Onderzoek" together with an "ID-beurs" of the University of Antwerp.
http://www.siam.org/journals/mms/10-3/84273.html

†Department of Mathematics and Computer Science, Universiteit Antwerpen, 2020 Antwerpen, Belgium (ynte.vanderhoydonc@ua.ac.be, wim.vanroose@ua.ac.be).

and a more recent review is given by Aidun and Clausen [1]. The application of the LBM to multiscale physics in fluids is discussed in [26]. Banda et al. [2] present a high order relaxation system for the multiscale lattice Boltzmann equation to obtain the incompressible Navier–Stokes limit.

Kevrekidis et al. introduced a lifting operator to couple different scales in a dynamical system in the equation-free framework [17]. This allows one to model the dynamics at the macroscopic level by using short bursts of the microscopic simulation.

A problem of lattice Boltzmann methods is the determination of initial conditions, usually given by macroscopic variables. During initialization and spatial coupling in a hybrid model, a one-to-many map needs to be created, known as a lifting operator. In this article, we discuss a hybrid LBM and PDE model that uses an LBM in one part of the domain, while another part of the domain is described by a macroscopic PDE. These different levels of description create missing data at the interfaces that can be resolved with a lifting operator.

Hybrid approaches have been formulated for various flow problems. A Lennard-Jones particle dynamics is coupled with a compressible Navier–Stokes equation in [12]. A Boltzmann model (respectively, an LBM) is coupled with the Navier–Stokes equations in [19, 18]. An LBM is also coupled with a Navier–Stokes equation in Peano, an adaptive mesh refinement framework with spacetree grids [20]. Furthermore, the Boltzmann equation is coupled with the Euler equations in [4].

Coupled models also play an important role in the simulation of materials. A review of atomistic-to-continuum coupling is found in [22]. A more detailed overview for coupling methods in hybrid models can be found in [13, 11]. In [13] adaptive mesh and algorithm refinement is used in parts of the domain where a continuum description is replaced by a particle description. The coupling of molecular dynamics and LBMs based on Schwarz's alternating method is presented in [11]. Dimarco and Pareschi [10] merge deterministic methods for the equilibrium part with particle methods for the nonequilibrium part and present results for the Boltzmann equation with Bhatnagar–Gross–Krook (BGK) approximation.

The coupling—that will be discussed in this article—of LBMs with reaction-diffusion PDEs was earlier considered in [29, 28, 33].

We propose a general lifting operator that maps densities to distribution functions. It is illustrated for an LBM, but we believe that it is applicable to general discretizations of the Boltzmann equation and it can map more moments to the corresponding distribution functions.

The new method will be compared to the Chapman–Enskog expansion [6], a well-known analytical method for the initialization of an LBM, and the Constrained Runs (CR) algorithm [30], a numerical lifting operator. The Chapman–Enskog expansion writes the distribution functions as an analytical series of the density. The CR algorithm is based on the attraction of the dynamics toward the slow manifold and expresses, in an implicit way, the unknown distribution functions with the help of the density in successive grid points. A numerical comparison of these methods is given in [33] for hybrid models that spatially couple a diffusion PDE model and an LBM. Although the CR algorithm is very accurate, its major drawback is the computational expense. It achieves a high accuracy for the coupling of an LBM with a diffusion-reaction PDE [29]. However, lifting in the CR algorithm requires many additional LBM steps. This computational cost is too expensive to be useful in more complex problems in higher dimensions. These drawbacks became clear in [33] when comparing the different methods numerically. The intention of this paper is to alleviate them.

In this paper we propose a numerical Chapman–Enskog expansion that seriously reduces the computational cost of the lifting. It combines the idea of the CR algorithm with the Chapman–Enskog expansion and does not need an analytical derivation as the Chapman–Enskog expansion. This lifting operator is calculated before the simulation and finds the coefficients of the Chapman–Enskog expansion numerically. Once these coefficients are found, the application of the lifting operator is just a stencil computation as cheap as the analytical Chapman–Enskog expansion. As a spin-off it also extracts the macroscopic PDE from the LBM. This allows us to construct the hybrid model without deriving the macroscopic PDE analytically. The numerical results show that the new lifting operator can also reach a high accuracy. Although we illustrate and benchmark the new method on academic model problems, we believe that it is applicable to other discretizations of the Boltzmann equation. Furthermore, for the clarity of the presentation we have kept the boundary between the LBM and PDE domain fixed. In a real application this boundary might be moved adaptively, triggered by an error estimate similar to that in adaptive mesh refinement.

This work is organized as follows. In section 2 the model problem is defined. It focuses, in particular, on a hybrid model that consists of an LBM in one part of the domain and a macroscopic equivalent PDE in another part. Section 3 gives an overview of existing lifting techniques that are used in the literature. The Chapman–Enskog expansion and the CR algorithm, respectively, are considered in sections 3.1 and 3.2. These methods are discussed in section 3.3. We tend to remove these drawbacks by considering a numerical Chapman–Enskog expansion in section 4. Section 5 contains the numerical results. In section 5.1 the proposed lifting operator is tested in a setting of restriction and lifting. The application of the lifting operator to the hybrid LBM and PDE model is considered in sections 5.2 and 5.3. We conclude and give an outlook in section 6.

2. Model problem. Kinetic models make use of the Boltzmann equation [25] that describes the evolution of a distribution function $f(\boldsymbol{x}, \boldsymbol{v}, t)$ (function space $C^2_{\mathbb{R}}(D)$) that counts the number of particles or individuals in point $\boldsymbol{x} \in D_{\boldsymbol{x}} \subset \mathbb{R}^n$, $n \in \mathbb{N}_0$, with a velocity $\boldsymbol{v} \in D_{\boldsymbol{v}} \subset \mathbb{R}^n$, at time $t \geq 0$. The equation is

$$\frac{\partial}{\partial t} f(\boldsymbol{x}, \boldsymbol{v}, t) + \boldsymbol{v} \frac{\partial}{\partial \boldsymbol{x}} f(\boldsymbol{x}, \boldsymbol{v}, t) + F(\boldsymbol{x}, t) \frac{\partial}{\partial \boldsymbol{v}} f(\boldsymbol{x}, \boldsymbol{v}, t) = \Omega. \tag{2.1}$$

This is an evolution law in phase space where $F(\boldsymbol{x}, t)$ is the external force and Ω is an integral operator that models the reorganization of the velocity distribution due to collisions or other interactions.

The collision operator can be approximated by a simpler BGK model $\Omega = \omega(f^{eq}(\boldsymbol{x}, \boldsymbol{v}, t) - f(\boldsymbol{x}, \boldsymbol{v}, t))$ [15] in which the equilibrium distribution $f^{eq}(\boldsymbol{x}, \boldsymbol{v}, t)$ is given by the Maxwell–Boltzmann distribution [25]. The BGK approximation represents a relaxation towards equilibrium with an associated time scale $\tau = 1/\omega$.

At the moment it is still computationally expensive to simulate or analyze a Boltzmann model numerically. The development of efficient numerical methods for models based on the Boltzmann equation is therefore an active research field. A possible increase in efficiency can be obtained by constructing a hybrid model. The kinetic model is then replaced with a macroscopic description in the regions of the spatial domain where this is justified, for example away from reaction fronts. These macroscopic models are cheaper to simulate.

Possible macroscopic models for fluid dynamics are described by the Navier–Stokes and Euler equations. One can derive both the Navier–Stokes and the Euler

equations from the Boltzmann equation [34]. A lifting operator then transforms the variables of the PDE to the distribution function of the Boltzmann equation at the boundaries between the domains.

In this paper we study a simple model that allows a detailed study of the lifting operator both for the initialization of the distribution function and in a hybrid context. The model uses an LBM with an equilibrium distribution function that depends only on the density, while the macroscopic PDE is an advection-diffusion-reaction equation for the density only. This simple model allows a detailed analysis, yet it is general enough to expect that the results can be extended to more realistic Boltzmann models. The correspondence of the lattice Boltzmann equation (LBE) with the Boltzmann equation is discussed in [34, 24].

2.1. LBMs. An LBM [25, 34] is a special discretization of (2.1). It describes the evolution of one-particle distribution functions $f_i(\boldsymbol{x}, t) = f(\boldsymbol{x}, \boldsymbol{v}_i, t)$ discretized in space $\boldsymbol{x}$, time t, and velocity $\boldsymbol{v}_i$. The velocities are taken from a discrete set defined by the geometry of the grid. The functions are represented as $f_i : \mathcal{X} \times \mathcal{T} \to \mathbb{R}$ with $\mathcal{X} \times \mathcal{T}$ the space-time grid with space steps $\Delta \boldsymbol{x}_i$ in the direction of velocity $\boldsymbol{v}_i$, time step Δt, and $\mathcal{T} = \{0, \Delta t, 2\Delta t, \ldots\} \cap [0, T]$. Representation DdQq used for the description of LBMs stands for d dimensions and q velocity directions. D1Q3, for example, considers in a one-dimensional spatial domain only three values for the velocity $v_i = c_i \Delta x / \Delta t$ with $c_i = i$, $i \in \{-1, 0, 1\}$, the dimensionless grid velocities.

The remainder of this section contains the description of the LBE in one dimension but can easily be generalized to more dimensions.

The LBE describing the evolution of the distribution functions with BGK approximation and no external force in (2.1) is

$$f_i(x + c_i \Delta x, t + \Delta t) = (1 - \omega) f_i(x, t) + \omega f_i^{eq}(x, t). \tag{2.2}$$

The equilibrium distributions are given by $f_i^{eq}(x, t) = \frac{1}{3}\rho(x, t)$, $i \in \{-1, 0, 1\}$ [28], in which the particle density $\rho(x, t)$ is defined as the zeroth order moment of the distribution functions $\rho(x, t) = \sum_{i \in \{-1,0,1\}} f_i(x, t)$. These equilibrium distributions correspond to a local diffusive equilibrium.

The focus of this paper concerns the initialization of an LBM. Starting the LBM scheme from a given initial density includes some arbitrariness. The distribution functions $f_i(x, 0)$ at time $t = 0$ need to be constructed from a given density $\rho(x, 0)$. When the initialization is not consistent, it leads to solutions with steep initial layers [21].

2.2. Macroscopic models for LBMs. This section contains descriptions that represent macroscopic equivalent PDEs specific for LBMs.

PDEs model, at a macroscopic scale, the evolution of the moments of the particle distribution functions like density $\rho(x, t) = \sum_i f_i(x, t)$, momentum $\phi(x, t) = \sum_i v_i f_i(x, t)$, or energy $\xi(x, t) = \frac{1}{2} \sum_i v_i^2 f_i(x, t)$.

The transition between the distribution functions and the moments is straightforward since the matrix M below is invertible:

$$\begin{pmatrix} \rho \\ \phi \\ \xi \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 0 & -1 \\ \frac{1}{2} & 0 & \frac{1}{2} \end{pmatrix} \begin{pmatrix} f_1 \\ f_0 \\ f_{-1} \end{pmatrix} = M \begin{pmatrix} f_1 \\ f_0 \\ f_{-1} \end{pmatrix}. \tag{2.3}$$

When we look at these functions in a point x at time t, they can be represented either as $(f_1, f_0, f_{-1})^T \in \mathbb{R}^3$ or as $(\rho, \phi, \xi)^T \in \mathbb{R}^3$. If we focus on the complete discretization in space, with n spatial grid points, the function spaces are $\mathbb{R}^{3 \times n}$.

It can be shown that the diffusion PDE and the LBM are macroscopic equivalent [28] when considering D1Q3 and

$$\frac{\partial \rho}{\partial t} = D\frac{\partial^2 \rho}{\partial x^2}, \quad D = \frac{2-\omega}{3\omega}\frac{\Delta x^2}{\Delta t}, \quad f_i^{eq}(x,t) = \frac{1}{3}\rho(x,t), \quad i \in \{-1,0,1\}. \tag{2.4}$$

This can be checked by using a Chapman–Enskog expansion. Here $f_i(x,t)$, $i \in \{-1,0,1\}$, is written as a series, each term containing higher order derivatives of $\rho(x,t)$ [6, 28, 5]:

$$f_i = f_i^{[0]} + f_i^{[1]}\Delta x + f_i^{[2]}\Delta x^2 + f_i^{[3]}\Delta x^3 + \cdots, \tag{2.5}$$

where

$$f_i^{[0]} = f_i^{eq} = \frac{\rho}{3}, \quad f_i^{[1]} = -\frac{i}{3\omega}\frac{\partial \rho}{\partial x}, \quad f_i^{[2]} = -\frac{1}{18\omega^2}(\omega-2)(3i^2-2)\frac{\partial^2 \rho}{\partial x^2}.$$

The macroscopic diffusion PDE (2.4) is obtained by summing the series of the Chapman–Enskog expansion in (2.5) over the velocities. This considers purely diffusive effects. Results that include advection and more spatial dimensions can be found in [27]. Similarly, advection-diffusion equations are obtained as macroscopic equivalents.

With such analytical expressions from the Chapman–Enskog expansion available, a lifting operator can be constructed. Indeed, $f_i(x,t)$ is then written as a series in function of the given density $\rho(x,t)$. This allows us to construct the distribution functions from a given initial density necessary to initialize the LBM.

We will consider the LBM as the "exact" model and these PDEs of the density as the macroscopic approximations. From this, we can construct a hybrid model problem as outlined in section 2.3.

2.3. Hybrid models. This section deals with the construction of the hybrid model problem. Consider the one-dimensional problem D1Q3, but bear in mind that a similar construction of a hybrid model can be done in higher dimensions. In particular, we couple an LBM with the macroscopic equivalent PDE. The resulting hybrid domain for D1Q3 is presented in Figure 2.1. A one-dimensional domain $[a,b]$ is considered that couples the PDE (2.4) on $[a,l[$ with the LBE (2.2) on $[l,b]$. Furthermore, we assume periodic boundary conditions. Note that in both subdomains the same grid spacings are used in space (Δx) and in time (Δt) and the boundary l remains fixed for all times. Using a different space-time grid is of particular interest for future work, but the same spacings are used to highlight the coupling error. Similarly, the boundary may be moved adaptively as in adaptive mesh refinement. In an actual physical problem it will be important to use the hybrid model that gives a Boltzmann description where shock waves, contact discontinuities, or sharp gradients occur. Since these move in time, it might be useful to work with a moving interface method as in [8, 9]. However, this is not the focus of the current paper.

On the domain $[a,l[$, we discretize the PDE (2.4) with cell centered central differences in space and forward Euler time discretization. The grid points x_j with $j \in \{0,1,\ldots,p\}$ cover this domain, and for these points it holds that

$$\rho(x_j, t+\Delta t) = \rho(x_j,t) + \frac{D\Delta t}{\Delta x^2}\left(\rho(x_{j-1},t) - 2\rho(x_j,t) + \rho(x_{j+1},t)\right). \tag{2.6}$$

For the grid points x_j with $j \in \{p+1,\ldots,n-1\}$ in $[l,b]$ the LBE (2.2) holds.

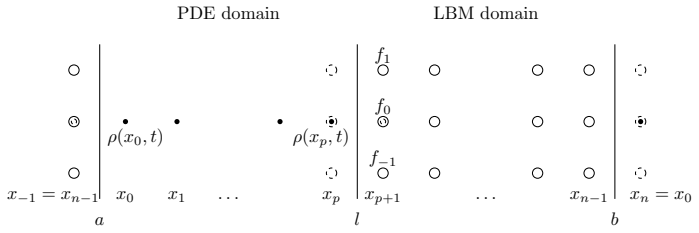

FIG. 2.1. *The domain $[a,b]$ in the hybrid model is split into $[a,l[$, on which we solve the PDE model, and $[l,b]$, on which we solve the LBM. The solid points ($\bullet$) represent the grid for the density ρ of the discrete PDE; the circles ($\circ$) represent the LBM variables $(f_1,f_0,f_{-1})^T$. The periodic boundary conditions and the coupling are implemented with ghostcells which are drawn by dashed circles. The density in the ghostcells of the PDE domain, in x_{-1} and x_{p+1}, are found by taking $\sum_i f_i$ in x_{n-1} and x_{p+1}, respectively. However, the ghostcells for the LBM domain, in x_p and x_n, require a lifting operator that lifts ρ to $(f_1,f_0,f_{-1})^T$ in these points.*

The full domain has an initial condition $\rho(x_j,0)\ \forall j \in \{0,1,\dots,n-1\}$. The lifting operator is required to formulate the initial conditions of the LBM domain $f_i(x_j,0)$ $\forall j \in \{p+1,\dots,n-1\}$.

The periodic boundary conditions lead to the following boundary conditions for the PDE domain: $\forall t : \rho(x_{-1},t) = \sum_i f_i(x_{n-1},t)$ and $\rho(x_{p+1},t) = \sum_i f_i(x_{p+1},t)$.

The aim is to construct the boundary conditions of the LBM domain in such a way that $\forall t > 0$ and $\forall j \in \{0,1,\dots,n-1\}$ the macroscopic density defined as

$$\rho(x_j,t) = \begin{cases} \rho(x_j,t) & \text{if} \quad j \in \{0,\dots,p\}, \\ \sum_i f_i(x_j,t) & \text{if} \quad j \in \{p+1,\dots,n-1\} \end{cases} \tag{2.7}$$

behaves as the density of an LBM solved on the full domain.

To formulate these boundary conditions, a lifting operator is required that maps the density $\rho(x,t)$ in the ghost points x_0 and x_p, the unknown of the PDE, to the distribution functions $f_i(x,t)$, $i \in \{-1,0,1\}$, of the LBM.

This can be generalized by considering a higher-dimensional spatial domain. The remaining derivations in this paper focus on one dimension, although they can also be generalized to more dimensions.

3. Review of existing lifting operators. This section gives an overview of existing lifting operators that map densities to distribution functions. An analytical expansion that expresses the distribution functions as a series of the density and its spatial derivatives is given in section 3.1, while a numerical method is presented in section 3.2. Section 3.3 discusses these methods, which results in the motivation to propose a lifting operator based on the combined ideas of the analytical and numerical methods.

3.1. Chapman–Enskog expansion. The Chapman–Enskog expansion [6, 28, 5], already discussed in section 2.3, can be used as a lifting operator. It constructs a mapping from $\rho(x,t)$ to $f_i(x,t)$, $i \in \{-1,0,1\}$ (D1Q3).

3.2. CR algorithm. An alternative numerical procedure is the CR algorithm discussed in this section. It is well known that in phase space the dynamics are quickly attracted toward a slow manifold [14]. For the problem we are studying the dynamics on the slow manifold can be parameterized by the density $\rho(x,t)$. The distribution functions are then of the form $\{f_1(\rho(x,t)), f_0(\rho(x,t)), f_{-1}(\rho(x,t))\}$.

Due to (2.3), it is equivalent to determine $\{f_1(\rho), f_0(\rho), f_{-1}(\rho)\}$ or $\{\rho, \phi(\rho), \xi(\rho)\}$. The missing distribution functions $\{f_1, f_0, f_{-1}\}$ can be found by determining ϕ and ξ for a given ρ such that ϕ, ξ, and ρ lie on the slow manifold. This is the basic idea of the CR algorithm that was proposed by Gear et al. [14] for stiff singularly perturbed ordinary differential equations (ODEs) to map macroscopic initial data to missing microscopic variables. It uses the numerical simulator to find the missing data such that the evolution is close to the slow manifold.

This CR algorithm can be applied to LBMs [30]. The state of the LBM can be split into

$$u = (\rho) \quad \text{and} \quad v = \begin{pmatrix} \phi \\ \xi \end{pmatrix},$$

where $u \in \mathbb{R}^n$ and $v \in \mathbb{R}^{2n}$ for an LBM with n spatial grid points. The density ρ is known, so u is given, while v is unknown since ϕ and ξ are missing. Denote the known initial conditions as $u(0) = u_0 \in \mathbb{R}^n$.

The idea is now to initialize v such that the evolution of v under the LBM is smooth of order m. The smoothness condition is defined by

$$\left.\frac{d^{m+1}v(t)}{dt^{m+1}}\right|_{t=0} = 0, \qquad \text{which is approximated by} \quad \Delta^{m+1}v(t) \approx \Delta t^{m+1}\frac{\mathrm{d}^{m+1}v(t)}{\mathrm{d}t^{m+1}},$$

where Δ^m is the well-known forward finite difference stencil on $v(t)$, $v(t+\Delta t)$, ... For $m = 0$, the converged v satisfies the smoothness condition, up to a certain tolerance, and it is an approximation to the point of intersection with the slow manifold. This is schematically represented in Figure 3.1. This iteration is always stable, and the point of intersection is found to first order accuracy compared to the Chapman–Enskog expansion for the LBM with BGK collisions for one-dimensional reaction-diffusion problems [30]. For $m \geq 1$, multiple LBM steps are necessary to estimate the derivative. For $m = 1$, this is often interpreted as a backward linear extrapolation in time [32].

The number of LBM steps used in the backward extrapolation determines the accuracy of the scheme. Higher order schemes increase the accuracy, but they can become unstable. In [32] this instability is circumvented by formulating the point of intersection as a fixed point

$$v^{k+1} = \mathcal{C}_m(u_0, v^k), \tag{3.1}$$

where $\mathcal{C}_m$ denotes one step of the CR algorithm and m is related to the order of the time derivative that is set to zero in the backward extrapolation in time. In general, (3.1) is nonlinear and the fixed point can be found by a Newton–Krylov iteration. However, this requires many additional LBM evaluations to construct the Jacobian. Similarly, matrix-free methods like GMRES still require many matrix-vector products since the spectrum is unfavorable for fast convergence [31]. In [33] the CR algorithm is combined with Newton's method by performing local updates at the ghost points of the hybrid model to reduce the size of the Jacobian.

3.3. Discussion of existing lifting operators. The methods discussed in sections 3.1 and 3.2 are well-known methods for constructing a lifting operator for LBMs. However, each of these methods has some drawbacks. As noted earlier, a drawback of the use of the Chapman–Enskog expansion (section 3.1) is the necessity to construct the expressions analytically. Therefore its use is limited to a few examples where the

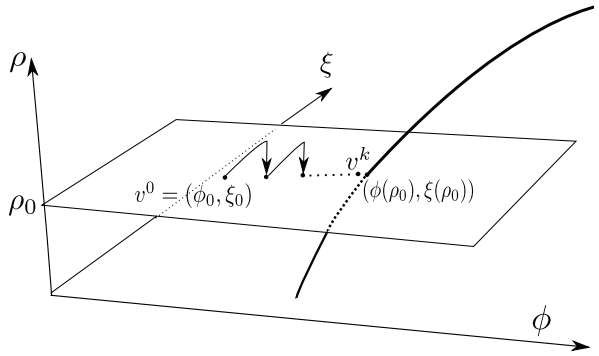

FIG. 3.1. *Sketch of the first few steps of the CR algorithm for the LBM with a constant backward extrapolation in time. The solid line shows the evolution of* $\{\rho(x,t), \phi(x,t), \xi(x,t)\}$ *along the slow manifold for a given grid point* x*. For a given* ρ_0 *we search for the intersection of the plane with the slow manifold. We start iterating with* ρ_0*, the known density, and initial guesses* ϕ_0 *and* ξ_0 *for the missing moments (*v^0*). After each step of the LBM, the density is reset to its initial value* ρ_0*, but the moments evolve during the LBM time simulation. This results in* v^k*, the* k*th iterate of the CR algorithm. This algorithm finds an approximation for the missing values* ϕ *and* ξ *on the slow manifold.*

expansion is known. However, its computational cost is limited to the calculation of the numerical approximation of the derivatives. The Chapman–Enskog expansion then becomes a stencil operator, and the cost of the application grows linearly with the number of points where lifting is required.

The CR scheme (section 3.2) can be used to approximate these expressions numerically. However, the lifting method can become computationally expensive since it requires many evaluations of the underlying LBM to construct the Jacobian matrix. Even with the matrix-free methods and local updates discussed at the end of section 3.2 it still remains computationally expensive to use in practice, especially in higher-dimensional problems.

As an advantage of the CR algorithm, we should note that the lifting error can be smaller than the modeling error, the difference in density between the LBM and its PDE approximation, by using the CR algorithm in the hybrid model discussed in section 2.3 [29]. Section 5.4 contains a comparison of the computational cost of these existing methods in the sense of hybrid models.

The focus of this paper is to obtain an alternative lifting operator that reduces the computational cost but holds the advantage of achieving the modeling error.

4. Numerical Chapman–Enskog expansion. In this section, we construct a lifting operator that alleviates the computational expense of the CR algorithm. It combines the ideas of CR and the Chapman–Enskog expansion. Instead of using CR to find for each grid point the missing moments ϕ and ξ of the distribution functions, we use CR to find the unknown coefficients of the Chapman–Enskog expansion. This has several advantages that will be discussed at the end of section 4.5.

The derivations in this section are again based on one-dimensional problems but can easily be generalized to more dimensions.

4.1. Distribution functions as a series of the density. This section shows that the solution $f_i(x,t)$, $i \in \{-1,0,1\}$, of an LBM with an infinite domain and parameters Δx, Δt, and ω can be written as a series of $\rho(x,t)$, the macroscopic

density. We initially characterize distribution functions f_i as smooth functions that are sufficiently differentiable functions in time and space, which implies that the same holds for the density, a sum of these distribution functions. The smoothness condition will be specified below. This condition can be justified when the lattice spacing Δx is much bigger than the mean free path [16, 7]. Then the distribution functions can be written as

$$f_i(x,t) = f_i^{eq}(x,t) + \alpha_i \frac{\partial \rho}{\partial x} + \beta_i \frac{\partial^2 \rho}{\partial x^2} + \cdots + \gamma_i \frac{\partial \rho}{\partial t} + \zeta_i \frac{\partial^2 \rho}{\partial t^2} + \cdots + \eta_i \frac{\partial^2 \rho}{\partial x \partial t} + \cdots, \tag{4.1}$$

where

$$\alpha = \begin{pmatrix} \alpha_1 \\ \alpha_0 \\ \alpha_{-1} \end{pmatrix} \in \mathbb{R}^3, \quad \beta = \begin{pmatrix} \beta_1 \\ \beta_0 \\ \beta_{-1} \end{pmatrix} \in \mathbb{R}^3, \ldots \tag{4.2}$$

are fixed constants that depend only on ω, Δx, and Δt. The derivation of this expansion is outlined in the remainder of this section.

Since the functions $f_i(x,t)$ are infinitely differentiable, a Taylor expansion can be constructed. The distribution functions in point $x + i\Delta x$, $i \in \{-1, 0, 1\}$, at time $t + \Delta t$ are given by

$$\begin{aligned} f_i(x + i\Delta x, t + \Delta t) = f_i(x,t) + \frac{\partial f_i}{\partial x} i\Delta x + \frac{\partial^2 f_i}{\partial x^2} \frac{i^2 \Delta x^2}{2} + \frac{\partial f_i}{\partial t} \Delta t \\ + \frac{\partial^2 f_i}{\partial t^2} \frac{\Delta t^2}{2} + \frac{\partial^2 f_i}{\partial x \partial t} i \Delta x \Delta t + \cdots. \end{aligned}$$

Combined with the assumption that f_i is a solution of the LBE (2.2) on an infinite domain, we end up with

$$f_i(x,t) = f_i^{eq}(x,t) - \frac{i\Delta x}{\omega} \frac{\partial f_i}{\partial x} - \frac{i^2 \Delta x^2}{2\omega} \frac{\partial^2 f_i}{\partial x^2} - \frac{\Delta t}{\omega} \frac{\partial f_i}{\partial t} - \frac{\Delta t^2}{2\omega} \frac{\partial^2 f_i}{\partial t^2} - \frac{i \Delta x \Delta t}{\omega} \frac{\partial^2 f_i}{\partial x \partial t} - \cdots.$$

With the notation $\mathcal{L}_i$ for the functional

$$\mathcal{L}_i = -\frac{i\Delta x}{\omega} \frac{\partial}{\partial x} - \frac{i^2 \Delta x^2}{2\omega} \frac{\partial^2}{\partial x^2} - \frac{\Delta t}{\omega} \frac{\partial}{\partial t} - \frac{\Delta t^2}{2\omega} \frac{\partial^2}{\partial t^2} - \frac{i \Delta x \Delta t}{\omega} \frac{\partial^2}{\partial x \partial t} - \cdots, \tag{4.3}$$

we can rewrite the LBE into a set of three coupled PDEs for the distribution functions

$$(1 - \mathcal{L}_i) f_i(x,t) = f_i^{eq}(x,t) \quad \forall i \in \{-1, 0, 1\} \tag{4.4}$$

that holds for $x \in]-\infty, \infty[$ and $t \in [0, \infty[$.

The solution can be found by performing a Picard or fixed point iteration

$$f_i^{(n+1)} = \mathcal{L}_i f_i^{(n)} + f_i^{eq}, \tag{4.5}$$

with initial guess $f_i^{(-1)} = 0$ that results in

$$\begin{aligned} f_i^{(0)} &= f_i^{eq} =: g_i, \\ f_i^{(1)} &= g_i + \mathcal{L}_i f_i^{(0)} = g_i + \mathcal{L}_i g_i, \\ f_i^{(2)} &= g_i + \mathcal{L}_i g_i + \mathcal{L}_i (\mathcal{L}_i g_i), \\ &\cdots \\ f_i^{(n)} &= \sum_{k=0}^{n} \mathcal{L}_i^k g_i, \end{aligned}$$

with $f_i^{(n)}$ the nth iterate. This iteration converges if the error between subsequent iterations goes to zero.

In contrast to traditional iterations, which require convergence for any initial guess, (4.5) is a fixed point iteration with initial guess zero and a smooth right-hand side. It is necessary to show convergence only for this particular case. To discuss this convergence we introduce the 2-norm, $\|.\|$, to show what happens between subsequent iterations. The absolute difference of subsequent iterations is given by

$$\|f_i^{(n+1)} - f_i^{(n)}\| = \|\mathcal{L}_i^{n+1} g_i\|.$$

This goes to zero if f_i is smooth enough, implying smoothness on ρ and $g_i = f_i^{eq}(x,t)$ such that $\lim_{n\to\infty} \|\mathcal{L}_i^{n+1} g_i\| = 0$. This smoothness condition depends on the parameters of the LBM, Δx, Δt, and ω, and the derivatives of g_i. For example, when g_i can be described by a polynomial, we have that there exists a k such that $\forall n > k$ it applies that $\|\mathcal{L}_i^{n+1} g_i\| = 0$.

We end up with the series (4.1) that consists of the vectors of constants given in (4.2), the density, and its derivatives. Once the constants are determined, the lifting operator—that is necessary to initialize the LBM and to determine the ghost points in the hybrid model—can be constructed. How these constants are found is discussed in sections 4.2 and 4.3. Section 4.2 deals with the analytical derivation, while section 4.3 is concerned with the numerical procedure. As a surplus, it allows us to find the corresponding macroscopic PDE as outlined in section 4.4.

4.2. Derivation of a lifting operator. With the help of expansion (4.1) it is possible to build a lifting operator that constructs the distribution functions for a given density. The focus of this section lies in the determination of the vectors of constants (4.2), the coefficients of such a lifting operator (4.1). To simplify the discussion and notation we limit ourselves to a truncated series

$$f(x,t) = f^{eq}(x,t) + \alpha \frac{\partial \rho}{\partial x} + \beta \frac{\partial^2 \rho}{\partial x^2} + \gamma \frac{\partial \rho}{\partial t}, \tag{4.6}$$

where α, β, and γ are the vectors containing the constants. The method is easily generalized to include higher order terms which will be considered in section 4.3.2.

Using the fact that (4.6) is valid for every possible grid point, we can consider three grid points x_j, x_k, and x_l and set up a linear system for the nine unknowns, namely three vectors each containing three constants, where j, k, and l are certain indices determined in a later stage of the paper:

$$\left(\begin{array}{ccc|ccc|ccc}
\frac{\partial\rho(x_j)}{\partial x} & & & \frac{\partial^2\rho(x_j)}{\partial x^2} & & & \frac{\partial\rho(x_j)}{\partial t} & & \\
 & \frac{\partial\rho(x_j)}{\partial x} & & & \frac{\partial^2\rho(x_j)}{\partial x^2} & & & \frac{\partial\rho(x_j)}{\partial t} & \\
 & & \frac{\partial\rho(x_j)}{\partial x} & & & \frac{\partial^2\rho(x_j)}{\partial x^2} & & & \frac{\partial\rho(x_j)}{\partial t} \\
\hline
\frac{\partial\rho(x_k)}{\partial x} & & & \frac{\partial^2\rho(x_k)}{\partial x^2} & & & \frac{\partial\rho(x_k)}{\partial t} & & \\
 & \frac{\partial\rho(x_k)}{\partial x} & & & \frac{\partial^2\rho(x_k)}{\partial x^2} & & & \frac{\partial\rho(x_k)}{\partial t} & \\
 & & \frac{\partial\rho(x_k)}{\partial x} & & & \frac{\partial^2\rho(x_k)}{\partial x^2} & & & \frac{\partial\rho(x_k)}{\partial t} \\
\hline
\frac{\partial\rho(x_l)}{\partial x} & & & \frac{\partial^2\rho(x_l)}{\partial x^2} & & & \frac{\partial\rho(x_l)}{\partial t} & & \\
 & \frac{\partial\rho(x_l)}{\partial x} & & & \frac{\partial^2\rho(x_l)}{\partial x^2} & & & \frac{\partial\rho(x_l)}{\partial t} & \\
 & & \frac{\partial\rho(x_l)}{\partial x} & & & \frac{\partial^2\rho(x_l)}{\partial x^2} & & & \frac{\partial\rho(x_l)}{\partial t}
\end{array}\right)
\left(\begin{array}{c}
\alpha_1 \\ \alpha_0 \\ \alpha_{-1} \\ \hline \beta_1 \\ \beta_0 \\ \beta_{-1} \\ \hline \gamma_1 \\ \gamma_0 \\ \gamma_{-1}
\end{array}\right)$$

$$= \begin{pmatrix} f_1(x_j,t) - f_1^{eq}(x_j,t) \\ f_0(x_j,t) - f_0^{eq}(x_j,t) \\ f_{-1}(x_j,t) - f_{-1}^{eq}(x_j,t) \\ \hline f_1(x_k,t) - f_1^{eq}(x_k,t) \\ f_0(x_k,t) - f_0^{eq}(x_k,t) \\ f_{-1}(x_k,t) - f_{-1}^{eq}(x_k,t) \\ \hline f_1(x_l,t) - f_1^{eq}(x_l,t) \\ f_0(x_l,t) - f_0^{eq}(x_l,t) \\ f_{-1}(x_l,t) - f_{-1}^{eq}(x_l,t) \end{pmatrix}. \tag{4.7}$$

For a given $f_i(x,t)$, where $i \in \{-1,0,1\}$, the linear system (4.7) will give the coefficients α, β, and γ. However, linear system (4.7) delivers the correct coefficients only if f_i is smooth enough such that f_i^{eq} satisfies the smoothness condition. This is the case when f_i lies on the slow manifold. The CR algorithm offers a way to reach the slow manifold in an iterative way.

We combine the ideas of the CR algorithm to reach the slow manifold and the Chapman–Enskog expansion to find the unknown constants on this slow manifold. The numerical procedure to do so is given in section 4.3.

If a PDE in closed form exists that describes the evolution of ρ in the form of $\rho_t + a\rho_x = D\rho_{xx}$, then the linear system (4.7) will be singular. Indeed, the PDE will give a relation between ρ_t, ρ_x, and ρ_{xx} in each of the grid points x_j, x_k, and x_l. As a result, every element in the last three columns of the linear system (4.7) can be written as a linear combination of the first six columns. In practice, however, the PDE is only an approximation and the system will be close to singular.

This clarifies why we do not solve the linear system in (4.7) and first focus on one that is not close to singular in section 4.3. Section 4.4 explains why such a PDE exists.

4.3. Numerical procedure for constructing the lifting operator. From the previous discussion it is clear that the coefficients of the lifting operator can be extracted from a linear system once f approaches the slow manifold. Next, the extraction of the coefficients is combined with the CR algorithm to bring f close to the slow manifold.

4.3.1. Coefficients of the Chapman–Enskog expansion as a fixed point. This discussion is limited, as in section 4.2, to the first few terms of the expansion. The singular system can be avoided by taking a series that contains only spatial derivatives. Such a series can represent the same state since the time derivative $\partial_t\rho$ is often related to the spatial derivatives through a macroscopic PDE. For example, suppose that a PDE of the form $\rho_t + a\rho_x = D\rho_{xx}$ describes the behavior of ρ. It is then possible to eliminate $\partial_t\rho$ from the expansion. The coefficients are then $\tilde{\alpha} = \alpha - \gamma a$ and $\tilde{\beta} = \beta + \gamma D$. The distribution functions are now series with only spatial derivatives:

$$f(x,t) = f^{eq}(x,t) + \tilde{\alpha}\frac{\partial\rho}{\partial x} + \tilde{\beta}\frac{\partial^2\rho}{\partial x^2}. \tag{4.8}$$

REMARK 1. *Rewrite $\tilde{\alpha}$ and $\tilde{\beta}$ as α and β, but bear in mind that it considers different coefficients in* (4.6) *and* (4.8).

Again, once the distribution functions are close to the slow manifold, we can

extract the coefficients α and β from the linear system

$$\left(\begin{array}{ccc|ccc} \frac{\partial\rho(x_j)}{\partial x} & & & \frac{\partial^2\rho(x_j)}{\partial x^2} & & \\ & \frac{\partial\rho(x_j)}{\partial x} & & & \frac{\partial^2\rho(x_j)}{\partial x^2} & \\ & & \frac{\partial\rho(x_j)}{\partial x} & & & \frac{\partial^2\rho(x_j)}{\partial x^2} \\ \hline \frac{\partial\rho(x_k)}{\partial x} & & & \frac{\partial^2\rho(x_k)}{\partial x^2} & & \\ & \frac{\partial\rho(x_k)}{\partial x} & & & \frac{\partial^2\rho(x_k)}{\partial x^2} & \\ & & \frac{\partial\rho(x_k)}{\partial x} & & & \frac{\partial^2\rho(x_k)}{\partial x^2} \end{array}\right)\left(\begin{array}{c} \alpha_1 \\ \alpha_0 \\ \alpha_{-1} \\ \hline \beta_1 \\ \beta_0 \\ \beta_{-1} \end{array}\right)$$

$$= \left(\begin{array}{c} f_1(x_j,t) - f_1^{eq}(x_j,t) \\ f_0(x_j,t) - f_0^{eq}(x_j,t) \\ f_{-1}(x_j,t) - f_{-1}^{eq}(x_j,t) \\ \hline f_1(x_k,t) - f_1^{eq}(x_k,t) \\ f_0(x_k,t) - f_0^{eq}(x_k,t) \\ f_{-1}(x_k,t) - f_{-1}^{eq}(x_k,t) \end{array}\right). \tag{4.9}$$

To reach the slow manifold we combine CR with the extraction of the coefficients. Consider a numerical function $h(\alpha, \beta; \rho, m)$ as described in Function 1. This function takes as input α and β and as parameters a fixed density ρ and an integer m, the order of the smoothness condition. It first constructs, with this input, a state f with the help of series (4.8). This state is then used to perform multiple LBM steps. For each of these steps we can find the corresponding moments ϕ and ξ. On the moments, we can use the CR algorithm (section 3.2) to find new moments that are closer to the slow manifold by considering the finite difference approximations of the mth order smoothness condition,

$$\frac{d^{m+1}\phi}{dt^{m+1}} = 0 \quad \text{and} \quad \frac{d^{m+1}\xi}{dt^{m+1}} = 0. \tag{4.10}$$

These new moments result in new coefficients α and β, by applying the linear system (4.9) on the distribution functions f_i, corresponding to the new ϕ and ξ and the given ρ.

The idea is now to determine α and β such that they are invariant under this numerical function $h(\alpha, \beta; \rho, m)$. Indeed, if the initial and final states can be described by the same α and β, then the lifted f is close to the slow manifold since it is a fixed point of the underlying CR iteration.

Instead of performing a regular fixed point iteration with $h(\alpha, \beta; \rho, m)$, a Newton iteration is used that finds $a := (\alpha, \beta) \in \mathbb{R}^6$ such that $a = h(a; \rho, m)$. This reduces the computational cost significantly because the size of the Jacobian system with α and β is much smaller than the Jacobian of the original CR algorithm. The latter involves the moments in every grid point, and this becomes very large.

The numerical function $h(\alpha, \beta; \rho, m)$ has a density $\rho(x, 0)$ as a parameter, and the solution for the coefficients is independent of its choice of ρ. The coefficients are determined by functional $\mathcal{L}_i$ (4.3) that depends only on constants ω, the spatial grid size Δx, and time step Δt. Since the coefficients do not depend on time, we can choose an arbitrary density $\rho(x, 0)$ such that (4.9) is easily solvable.

The choice of the grid points x_j and x_k in (4.9) should be such that the condition number of the matrix is optimal. In addition, the spatial derivatives that are considered need to exist and should not become zero during the LBM evolution since otherwise we would end up with singular linear systems.

FUNCTION 1. $h(\alpha, \beta; \rho, m)$

Require: Guess on coefficients α, β, given density ρ, order m to use in (4.10).

1: Construct lifting operator $f = f^{eq} + \alpha\frac{\partial \rho}{\partial x} + \beta\frac{\partial^2 \rho}{\partial x^2}$ (see (4.8)).
2: Compute corresponding moments ϕ and ξ by applying (2.3).
3: Perform $m+1$ LBM time steps to compute $\frac{d^{m+1}\phi}{dt^{m+1}} = 0$ and $\frac{d^{m+1}\xi}{dt^{m+1}} = 0$ by using forward finite difference formulas. This results in new moments ϕ and ξ. {find moments closer to slow manifold}
4: Revert back to distribution functions f by applying (2.3).
5: Select grid points x_j and x_k to construct linear system (4.9).
6: Solve the system for α and β.
7: **return** α, β.

Furthermore, the test domain used in the LBM inside the function $h(\alpha, \beta; \rho, m)$ can be significantly smaller than the domain of the original LBM problem. A smaller test domain will not affect the constant coefficients of the lifting operator. However, it should use the same Δx and Δt as the LBM of interest since the coefficients depend on the chosen spacings in space and time. The choices for the test domain, density, and indices are further discussed in section 5.1 for the considered model problem.

4.3.2. Higher order versions. There are two ways to increase the accuracy. First, more terms in the expansion can be considered such that more derivatives of the density are taken into account. Second, we can enforce a higher order smoothness in the CR algorithm. Both methods are outlined below.

The proposed method can easily be extended by considering more terms with higher order derivatives in the truncated series (4.8). For example, consider the expansion

$$f = f^{eq} + \alpha\frac{\partial \rho}{\partial x} + \beta\frac{\partial^2 \rho}{\partial x^2} + \delta\frac{\partial^3 \rho}{\partial x^3} + \epsilon\frac{\partial^4 \rho}{\partial x^4} \tag{4.11}$$

that now requires the determination of more coefficients that are found by considering —in addition to x_j and x_k—additional grid points x_l and x_m. This leads to a larger system of unknowns but will give better results.

Higher order smoothness can be enforced on the moments ϕ and ξ as in the CR algorithm by considering a higher order m in (4.10). This requires more LBM steps and uses a higher order finite difference formula to estimate the derivatives in time.

For further conclusions and results, higher order derivatives and higher order smoothness are taken into account.

4.4. Derivation of macroscopic PDE. Next, we derive from (4.1) the macroscopic PDE by summing over the velocities. Using $\sum_i f_i(x,t) = \rho(x,t) = \sum_i f_i^{eq}(x,t)$ results in a macroscopic PDE for the density:

$$\begin{aligned}\left(\sum_i \gamma_i\right)\frac{\partial \rho}{\partial t} + \left(\sum_i \zeta_i\right)\frac{\partial^2 \rho}{\partial t^2} + \cdots + \left(\sum_i \eta_i\right)\frac{\partial^2 \rho}{\partial x \partial t} + \cdots \\ = -\left(\sum_i \alpha_i\right)\frac{\partial \rho}{\partial x} - \left(\sum_i \beta_i\right)\frac{\partial^2 \rho}{\partial x^2} - \left(\sum_i \delta_i\right)\frac{\partial^3 \rho}{\partial x^3} - \left(\sum_i \epsilon_i\right)\frac{\partial^4 \rho}{\partial x^4} - \cdots .\end{aligned}$$

Series (4.1) derived in this setting leads to the classical Chapman–Enskog expan-

sion [28, 5] that we obtained in section 2.2. Indeed, (4.1) is written as

$$f_i(x,t) = g_i + \mathcal{L}_i g_i + \mathcal{L}_i^2 g_i + \cdots$$

because of the application of the fixed point iteration. When the series is truncated after the second order spatial derivative and the first order time derivative, we end up with

$$f_i(x,t) = \frac{1}{3}\rho - \frac{i\Delta x}{3\omega}\frac{\partial \rho}{\partial x} + \left(\frac{i^2\Delta x^2}{3\omega^2} - \frac{i^2\Delta x^2}{6\omega}\right)\frac{\partial^2 \rho}{\partial x^2} - \frac{\Delta t}{3\omega}\frac{\partial \rho}{\partial t}.$$

Summing $f_i(x,t)$ over $i \in \{-1,0,1\}$ we obtain the same macroscopic diffusion PDE (2.4). Substituting this PDE in the series to remove the time derivative leads to the classical Chapman–Enskog expansion in (2.5).

This macroscopic PDE was used for the removal of the time derivative in (4.6) and its replacement with (4.8). Furthermore, it is important to note that the macroscopic PDE is not necessarily of the reaction-diffusion prototype. Truncating (4.1) after more terms and taking more derivatives into account results in a better output, but it will lead to the term $\frac{\partial^2 \rho}{\partial t^2}$, which gives a less comfortable macroscopic PDE.

This relation to the macroscopic PDE can now be integrated in the numerical Chapman–Enskog method. Once the fixed point described in section 4.3.1 is found, we have α and β, which lifts ρ to the distribution functions close to the slow manifold. By performing two more LBM steps, $\frac{\partial \rho}{\partial t}$ can be calculated by using a forward finite difference formula. System (4.7) can be applied to find the vectors of constants of this larger system that include the time derivative. There are now two possibilities: (1) the resulting system is nonsingular, and it can be solved for the coefficients and only an approximate PDE can be found, as is considered above; (2) the resulting system is too singular to be solved accurately, but then the PDE can be extracted from the nullspace of the system.

Let us first discuss the situation where the matrix in (4.7) is nonsingular. The system can then be solved for α, β, and γ. The approximate PDE can be determined by summing over the obtained coefficients:

$$\frac{\partial \rho}{\partial t} = -\frac{\sum_i \alpha_i}{\sum_i \gamma_i}\frac{\partial \rho}{\partial x} - \frac{\sum_i \beta_i}{\sum_i \gamma_i}\frac{\partial^2 \rho}{\partial x^2}.$$

This PDE is only approximate. Otherwise, if it would hold exactly, the system would be singular, as expected.

For a singular system, we know that one or more of the eigenvalues will be zero with a corresponding null eigenvector. Focusing on the null eigenvector $v = \{v_1, v_2, \ldots, v_9\}$, we know that $Av = 0$ with A the matrix in system (4.7). Using this, we obtain

$$\frac{\partial \rho(x_j)}{\partial x}v_1 + \frac{\partial^2 \rho(x_j)}{\partial x^2}v_4 + \frac{\partial \rho(x_j)}{\partial t}v_7 = 0,$$

from which we conclude that the resulting PDE looks like

$$\frac{\partial \rho(x_j)}{\partial t} = -\frac{v_1}{v_7}\frac{\partial \rho(x_j)}{\partial x} - \frac{v_4}{v_7}\frac{\partial^2 \rho(x_j)}{\partial x^2}.$$

Remark that the same PDE will be found when considering this equation in grid points x_k and x_l instead of x_j.

4.5. Algorithm for lifting operator and macroscopic PDE. The results of the previous sections are now combined in an algorithm that delivers a lifting operator and an approximate macroscopic PDE. This can be used, for example, to construct a hybrid model. The pseudocode is presented in Algorithm 2, while the complete algorithm is presented below. The algorithm starts by searching for the lifting operator on the basis of the spatial derivatives. Thereafter, it inserts time derivatives and calculates the coefficients of the macroscopic PDE.

Start with an initial guess for $\{\alpha, \beta, \delta, \epsilon\}$ in (4.11). Then apply Function 1 $h(\alpha, \beta, \delta, \epsilon; \rho, m)$ for a given ρ and a certain m for the order of smoothness. This results in coefficients $\{\alpha, \beta, \delta, \epsilon\}$ that represent distribution functions closer to the slow manifold. The lifting operator is constructed at this point.

When these distribution functions are found based on the spatial derivatives only, we still need to determine the corresponding PDE by considering the null eigenvector or by a summation of the coefficients as discussed in section 4.4. By performing two extra LBM steps—to estimate the time derivative with a forward finite difference formula—the coefficient γ belonging to the time derivative of the expansion below can be numerically calculated:

$$f = f^{eq} + \alpha \frac{\partial \rho}{\partial x} + \beta \frac{\partial^2 \rho}{\partial x^2} + \delta \frac{\partial^3 \rho}{\partial x^3} + \epsilon \frac{\partial^4 \rho}{\partial x^4} + \gamma \frac{\partial \rho}{\partial t}.$$

Since the PDE can be obtained from the numerically constructed distribution functions, the PDE obtained through the Chapman–Enskog expansion does not need to be obtained analytically.

REMARK 2. *Note that one can also consider* $f^{eq}(x,t) = \kappa\rho(x,t)$ *and determine the constants of vector* κ *in a similar setting by using an extra grid point to obtain a larger system of unknowns.*

REMARK 3. *In this paper we have chosen to use the same* Δx *and* Δt *in the LBM as in the PDE. However, their stability properties may be different. Our specific LBM simulation is stable in the* 2*-norm when* $0 \leq \omega \leq 2$ [23]. *However, it is not necessary that the macroscopic PDE, when it is discretized with the same* Δx *and* Δt *and forward Euler, is also stable. Indeed, when* $\omega \to 0$ *for a fixed* Δx *and* Δt *the resulting diffusion coefficient* D *grows (see* (2.4)*), and this can lead to an instability.*

ALGORITHM 2. Pseudocode numerical Chapman–Enskog expansion

Require: Test domain that defines $\rho(x,0)$, initial guess $a^0 = \{\alpha, \beta, \delta, \epsilon\} = \text{zeros}(12,1)$ and a user-defined tolerance tol, parameter m for the order of smoothness.
repeat
$a^{k+1} = a^k - \big(J(a^k)\big)^{-1} h(a^k; \rho, m)$ with h defined in Function 1.
until $\|a^{k+1} - a^k\| <$ tol.
Result for coefficients $\{\alpha, \beta, \delta, \epsilon\}$ belonging to the spatial derivatives.
$f = f^{eq} + \alpha\frac{\partial \rho}{\partial x} + \beta\frac{\partial^2 \rho}{\partial x^2} + \delta\frac{\partial^3 \rho}{\partial x^3} + \epsilon\frac{\partial^4 \rho}{\partial x^4}$ closer to the slow manifold.
Perform two more LBM steps to determine $\frac{\partial \rho}{\partial t}$ numerically.
Construct system (4.7)—including higher order spatial derivatives—to achieve coefficients $\{\alpha, \beta, \delta, \epsilon, \gamma\}$.
Determine coefficients of PDE (section 4.4).
return Lifting operator $f = f^{eq} + \alpha\frac{\partial \rho}{\partial x} + \beta\frac{\partial^2 \rho}{\partial x^2} + \delta\frac{\partial^3 \rho}{\partial x^3} + \epsilon\frac{\partial^4 \rho}{\partial x^4}$ and macroscopic PDE.

The pseudocode of the numerical Chapman–Enskog expansion as a lifting opera-

tor is given in Algorithm 2 together with the determination of the transport coefficients of the PDE to construct a hybrid model. The proposed algorithm has several advantages. In contrast to the Chapman–Enskog expansion no analytical work is required. Compared to the CR algorithm it significantly reduces the number of unknowns in the lifting since only the coefficients (vectors of constants) are needed rather than the full state of the distribution functions. Furthermore, it can be done off-line before the calculations. Indeed, once the coefficients are found they can be reused every time step to realize the lifting. As an extra surplus, the PDE can be determined to construct hybrid models.

5. Numerical results. The new lifting operator is now illustrated in several examples. First we benchmark its accuracy against a reference solution that is reconstructed. This is done in section 5.1. In section 5.2 we recall the one-dimensional hybrid model of Figure 2.1. A two-dimensional problem is discussed in section 5.3. The important comparison of the additional required LBM steps to perform the lifting in a hybrid model is presented in section 5.4.

5.1. Numerical comparison of different lifting operators. The proposed lifting operator can be tested against a reference distribution function f_c. This reference solution is calculated by performing 1000 lattice Boltzmann steps starting from an initial state that corresponds to the equilibrium distribution function of a given density ρ.

The lifting operator can now be evaluated when we restrict the reference distribution function f_c to its density $\rho = \sum_i f_i(x,t)$ and lift it back to a distribution function f by using the proposed lifting operator. The resulting f will be compared with f_c with the help of the 2-norm $\|f - f_c\|$.

EXAMPLE 1. *The considered model problem has the following parameters for a one-dimensional domain of length L:*

$$L = 10, \quad n = 200, \quad \Delta x = \frac{L}{n} = 0.05, \quad \Delta t = 0.001,$$
$$\rho(x,0) = e^{-(x-\frac{L}{2})^2}, \quad \omega = 0.9091.$$

For these parameters the classical Chapman–Enskog expansion predicts a diffusion coefficient $D = 1$ (see (2.4)).

To reproduce the numerical results linked to Example 1 we include some extra information on how to determine the indices of (4.9). As mentioned in section 4.3.1 we can choose an arbitrary initial density $\rho(x,0)$ and a test domain to determine the constants of the lifting operator (4.8). For example, consider $\rho(x,0) = x + 1/2x^2$ for unknowns α, β and $\rho(x,0) = x + 1/2x^2 + 1/6x^3$ for unknowns α, β, δ. x is defined by the spatial nodes of the test domain defined below. Furthermore, the test domain reduces the computational expense compared to the actual spatial domain.

Consider the domain parameters of Example 1. Since we know that the constants are affected only by these space and time steps, we should consider—together with $\rho(x,0)$—a test domain with the same step sizes since the vectors of constants (4.2) are affected by these choices. The test domain is of length $L_{test} = 3$ such that $n_{test} = 60$ since $\Delta x = 0.05$. This number of grid points will make it possible to choose the indices x_j, x_k, ... such that system (4.9) is not close to singular. We can return now to the question of which indices should be used in system (4.9). Focus on the fact that we do not want an effect of wrongly chosen boundary conditions in the smaller test domain. The grid points should be taken far enough from the edges and in points

TABLE 5.1

The error $\|f - f_c\|$ is presented to test the exact Chapman–Enskog expansion as a lifting operator. The reference solution is obtained after 1000 *LBM steps in the model problem from Example* 1.

Construction lifting operator: Exact Chapman–Enskog expansion	$\|f - f_c\|$
$f = f^{eq}$	0.0388
$f = f^{eq} + \alpha_{exact}\frac{\partial \rho}{\partial x}$	5.2341e-004
$f = f^{eq} + \alpha_{exact}\frac{\partial \rho}{\partial x} + \beta_{exact}\frac{\partial^2 \rho}{\partial x^2}$	2.7570e-005
$f = f^{eq} + \alpha_{exact}\frac{\partial \rho}{\partial x} + \beta_{exact}\frac{\partial^2 \rho}{\partial x^2} + \delta_{exact}\frac{\partial^3 \rho}{\partial x^3}$	1.2439e-005

TABLE 5.2

The error $\|f-f_c\|$ with the CR algorithm (combined with Newton's method) for various orders of accuracy as a lifting operator. The reference solution is obtained for the model problem in Example 1 *by performing* 1000 *LBM time steps before restricting and lifting. The last column contains results when an extra advection effect of $a = 0.66$ is included—which changes the used equilibrium distribution functions.*

Construction lifting operator: Extrapolation CR algorithm	$\|f - f_c\|$ pure diffusion $D = 1$	$\|f - f_c\|$ plus advection $a = 0.66$
Constant	0.0010	0.0014
Linear	1.3578e-006	1.7927e-006
Quadratic	2.9359e-009	3.9069e-009
Cubic	9.0125e-012	1.1898e-011

such that the system (4.9) does not become singular. The indices can be, for example, 10, 20, 30, ... spread over the test domain of 60 grid points.

Note that one can also focus on local updates around the considered grid points x_j, x_k, ... When the number of iterations needed in Newton's method are known, one knows how many LBM steps will be performed to find the new coefficients α and β. Then the size of the test domain can be shrunk to a smaller domain around x_j, x_k, ... This is the same idea as used in [33] to perform local updates for the CR algorithm.

To compare the proposed lifting operator with the existing ones discussed in section 3, the results for $\|f - f_c\|$ of the different lifting operators are included in Tables 5.1, 5.2, and 5.3.

Table 5.1 contains results obtained with the analytical Chapman–Enskog expansion as a lifting operator. The first column gives the order of the expansion, and the second column shows $\|f - f_c\|$. As expected a better accuracy is obtained with higher order expansions. For example, by taking the third derivative of the density into account an error of 1.24e-5 is achieved.

In Table 5.2 we use the CR algorithm of various orders of accuracy to numerically lift the density to distribution functions. Different types of backward extrapolation are listed in the first column of the table, while the corresponding 2-norm $\|f - f_c\|$ is described in the second column. There we see that very accurate results can be found for the higher order versions. Note that these methods find for each grid point the moments ϕ and ξ of the distribution functions. Together with ρ, the corresponding distribution functions are found by (2.3). Since this gives a local solution, it can give accurate results. The last column contains $\|f - f_c\|$ when some extra advection effect is included, which shows results similar to those in the pure diffusion problem.

Table 5.3 shows the results with the proposed numerical Chapman–Enskog expansion as a lifting operator. We clearly see that taking more terms in the expansion,

TABLE 5.3

The error $\|f - f_c\|$ with the numerical Chapman–Enskog expansion as a lifting operator. The reference distribution function f_c is obtained by performing 1000 *LBM time steps with parameters listed in Example* 1*. Each of the lifting operators is calculated as a fixed point for the coefficients. The first table shows the results with one LBM step before updating the moments, implying an update of the coefficients. There are results listed for an increasing number of terms in the expansion, implying an increasing number of considered coefficients. The second table shows the same results where two LBM steps are used to estimate the smoothness. The third and fourth tables show the results with a quadratic (respectively, cubic) computation of the finite difference approximation in* (4.10)*. The constant coefficients found via the numerical procedure are compared to those found exactly with the classical Chapman–Enskog expansion in columns* 3*,* 4*, and* 5*.*

Construction lifting operator: Numerical Chapman–Enskog constant computation of fixed point	$\|f - f_c\|$	$\|\alpha - \alpha_{exact}\|$	$\|\beta - \beta_{exact}\|$	$\|\delta - \delta_{exact}\|$
$f = f^{eq} + \alpha\frac{\partial\rho}{\partial x}$	5.2341e-004	1.9981e-016	/	/
$f = f^{eq} + \alpha\frac{\partial\rho}{\partial x} + \beta\frac{\partial^2\rho}{\partial x^2}$	0.0010	3.7153e-016	8.9815e-004	/
$f = f^{eq} + \alpha\frac{\partial\rho}{\partial x} + \beta\frac{\partial^2\rho}{\partial x^2} + \delta\frac{\partial^3\rho}{\partial x^3}$	0.0010	2.0200e-015	8.9815e-004	2.0310e-005
$f = f^{eq} + \alpha\frac{\partial\rho}{\partial x} + \beta\frac{\partial^2\rho}{\partial x^2} + \delta\frac{\partial^3\rho}{\partial x^3} + \epsilon\frac{\partial^4\rho}{\partial x^4}$	0.0010	5.0363e-015	8.9815e-004	2.0310e-005
$f = f^{eq} + \alpha\frac{\partial\rho}{\partial x} + \beta\frac{\partial^2\rho}{\partial x^2} + \delta\frac{\partial^3\rho}{\partial x^3} + \epsilon\frac{\partial^4\rho}{\partial x^4} + \theta\frac{\partial^5\rho}{\partial x^5}$	0.0010	1.2979e-013	8.9815e-004	2.0310e-005
$f = f^{eq} + \alpha\frac{\partial\rho}{\partial x} + \beta\frac{\partial^2\rho}{\partial x^2} + \delta\frac{\partial^3\rho}{\partial x^3} + \epsilon\frac{\partial^4\rho}{\partial x^4} + \theta\frac{\partial^5\rho}{\partial x^5} + \iota\frac{\partial^6\rho}{\partial x^6}$	0.0010	9.0311e-013	8.9815e-004	2.0310e-005
Linear computation of fixed point	$\|f - f_c\|$	$\|\alpha - \alpha_{exact}\|$	$\|\beta - \beta_{exact}\|$	$\|\delta - \delta_{exact}\|$
$f = f^{eq} + \alpha\frac{\partial\rho}{\partial x}$	5.2341e-004	9.0005e-017	/	/
$f = f^{eq} + \alpha\frac{\partial\rho}{\partial x} + \beta\frac{\partial^2\rho}{\partial x^2}$	2.7570e-005	2.3990e-015	5.3202e-015	/
$f = f^{eq} + \alpha\frac{\partial\rho}{\partial x} + \beta\frac{\partial^2\rho}{\partial x^2} + \delta\frac{\partial^3\rho}{\partial x^3}$	6.8677e-007	9.1594e-015	2.1865e-014	1.0803e-005
$f = f^{eq} + \alpha\frac{\partial\rho}{\partial x} + \beta\frac{\partial^2\rho}{\partial x^2} + \delta\frac{\partial^3\rho}{\partial x^3} + \epsilon\frac{\partial^4\rho}{\partial x^4}$	1.3591e-006	7.2493e-015	1.6524e-014	1.0803e-005
$f = f^{eq} + \alpha\frac{\partial\rho}{\partial x} + \beta\frac{\partial^2\rho}{\partial x^2} + \delta\frac{\partial^3\rho}{\partial x^3} + \epsilon\frac{\partial^4\rho}{\partial x^4} + \theta\frac{\partial^5\rho}{\partial x^5}$	1.3591e-006	1.4579e-013	3.6404e-013	1.0803e-005
$f = f^{eq} + \alpha\frac{\partial\rho}{\partial x} + \beta\frac{\partial^2\rho}{\partial x^2} + \delta\frac{\partial^3\rho}{\partial x^3} + \epsilon\frac{\partial^4\rho}{\partial x^4} + \theta\frac{\partial^5\rho}{\partial x^5} + \iota\frac{\partial^6\rho}{\partial x^6}$	1.3578e-006	2.6277e-012	6.4797e-012	1.0803e-005
Quadratic computation of fixed point	$\|f - f_c\|$	$\|\alpha - \alpha_{exact}\|$	$\|\beta - \beta_{exact}\|$	$\|\delta - \delta_{exact}\|$
$f = f^{eq} + \alpha\frac{\partial\rho}{\partial x}$	5.2341e-004	1.2647e-016	/	/
$f = f^{eq} + \alpha\frac{\partial\rho}{\partial x} + \beta\frac{\partial^2\rho}{\partial x^2}$	2.7570e-005	7.2802e-016	1.1928e-015	/
$f = f^{eq} + \alpha\frac{\partial\rho}{\partial x} + \beta\frac{\partial^2\rho}{\partial x^2} + \delta\frac{\partial^3\rho}{\partial x^3}$	6.8677e-007	1.1455e-014	2.7241e-014	1.0803e-005
$f = f^{eq} + \alpha\frac{\partial\rho}{\partial x} + \beta\frac{\partial^2\rho}{\partial x^2} + \delta\frac{\partial^3\rho}{\partial x^3} + \epsilon\frac{\partial^4\rho}{\partial x^4}$	4.1047e-008	2.9450e-014	6.4892e-014	1.0803e-005
$f = f^{eq} + \alpha\frac{\partial\rho}{\partial x} + \beta\frac{\partial^2\rho}{\partial x^2} + \delta\frac{\partial^3\rho}{\partial x^3} + \epsilon\frac{\partial^4\rho}{\partial x^4} + \theta\frac{\partial^5\rho}{\partial x^5}$	1.4994e-009	3.3586e-013	8.3590e-013	1.0803e-005
$f = f^{eq} + \alpha\frac{\partial\rho}{\partial x} + \beta\frac{\partial^2\rho}{\partial x^2} + \delta\frac{\partial^3\rho}{\partial x^3} + \epsilon\frac{\partial^4\rho}{\partial x^4} + \theta\frac{\partial^5\rho}{\partial x^5} + \iota\frac{\partial^6\rho}{\partial x^6}$	2.9449e-009	1.3229e-011	3.3252e-011	1.0803e-005
Cubic computation of fixed point	$\|f - f_c\|$	$\|\alpha - \alpha_{exact}\|$	$\|\beta - \beta_{exact}\|$	$\|\delta - \delta_{exact}\|$
$f = f^{eq} + \alpha\frac{\partial\rho}{\partial x}$	5.2341e-004	7.9768e-016	/	/
$f = f^{eq} + \alpha\frac{\partial\rho}{\partial x} + \beta\frac{\partial^2\rho}{\partial x^2}$	2.7570e-005	4.1708e-015	1.0779e-014	/
$f = f^{eq} + \alpha\frac{\partial\rho}{\partial x} + \beta\frac{\partial^2\rho}{\partial x^2} + \delta\frac{\partial^3\rho}{\partial x^3}$	6.8677e-007	1.2825e-014	3.5932e-014	1.0803e-005
$f = f^{eq} + \alpha\frac{\partial\rho}{\partial x} + \beta\frac{\partial^2\rho}{\partial x^2} + \delta\frac{\partial^3\rho}{\partial x^3} + \epsilon\frac{\partial^4\rho}{\partial x^4}$	4.1047e-008	3.9984e-014	7.9750e-014	1.0803e-005
$f = f^{eq} + \alpha\frac{\partial\rho}{\partial x} + \beta\frac{\partial^2\rho}{\partial x^2} + \delta\frac{\partial^3\rho}{\partial x^3} + \epsilon\frac{\partial^4\rho}{\partial x^4} + \theta\frac{\partial^5\rho}{\partial x^5}$	1.4995e-009	6.9980e-013	1.6787e-012	1.0803e-005
$f = f^{eq} + \alpha\frac{\partial\rho}{\partial x} + \beta\frac{\partial^2\rho}{\partial x^2} + \delta\frac{\partial^3\rho}{\partial x^3} + \epsilon\frac{\partial^4\rho}{\partial x^4} + \theta\frac{\partial^5\rho}{\partial x^5} + \iota\frac{\partial^6\rho}{\partial x^6}$	9.4492e-011	2.2682e-011	5.5927e-011	1.0803e-005

i.e., more derivatives of $\rho(x,t)$ in the lifting, leads to a better lifting operator. In the same table we show the results with higher order smoothness conditions by using (4.10) with higher order m. As in the CR algorithm, higher order smoothness does result in a significant improvement. The accuracy increases to 9.45e-11 when up to the sixth spatial derivative is taken into account. Including advection in this table will also show similar results, but these are not added.

A comparison of Tables 5.1, 5.2, and 5.3 shows that the proposed numerical lifting operator leads to better results than the analytically found Chapman–Enskog expansion. As can be seen, the CR algorithm is a good lifting method, but, as will be discussed in section 5.4, the computational expense of this method brings down the beauty of it. Table 5.4 of section 5.4 contains a comparison of the number of

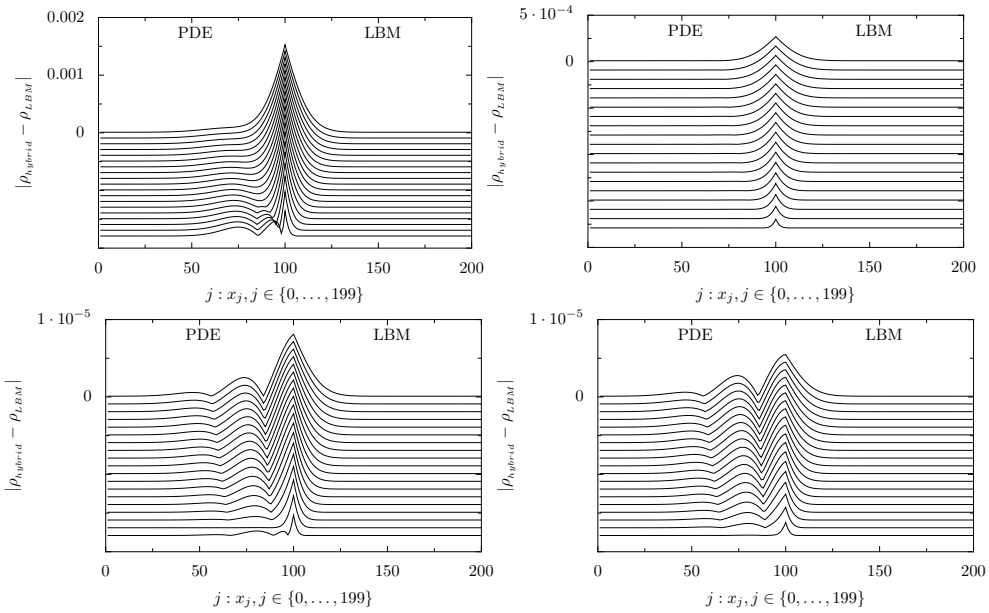

FIG. 5.1. $|\rho_{hybrid} - \rho_{LBM}|$ *after* 200 *time steps. The difference is also shown at earlier time slots but shifted down for clarity. The lines represent time steps between one and* 200*. The top line corresponds to time step* 191*, while the bottom line represents time step* 11*. The lines in between correspond to jumps with* 10 *time steps from* 11 *to* 21*, . . . ,* 181 *to* 191*. The domain that is considered is shown in Figure* 2.1*. The lifting operator is* $f_i(x,t) = 1/3\,\rho(x,t)$ *(top left), first order (top right), second order (bottom left), and third order Chapman–Enskog (bottom right), respectively. The lifting operator is used both to find the ghost points of the LBM domain and for the creation of the initial state for the LBM region. The model problem parameters are listed in Example* 1*.*

additional LBM steps required for each of the lifting operators. In the CR algorithm this additional cost can be attributed to the construction of the Jacobian matrix. These additional LBM steps make the method computationally very expensive. In two dimensions this method becomes prohibitive. This makes the numerical Chapman–Enskog lifting operator a good alternative to the CR algorithm that gives a similar accuracy at a limited computational cost.

5.2. One-dimensional test problem. To compare the results of the numerical Chapman–Enskog expansion with the earlier proposed lifting operators discussed in section 3, Figures 5.1 and 5.2 show the results of the absolute difference $|\rho_{\text{hybrid}} - \rho_{\text{LBM}}|$ for the exact Chapman–Enskog expansion and those obtained with the CR algorithm. ρ_{hybrid} is the density of the hybrid model and ρ_{LBM} the density of a full LBM. ρ_{LBM} is the reference solution to compare the hybrid solution with. It considers an LBM on the whole spatial domain $[a, b]$ with the parameters outlined in Example 1 and the domain represented in Figure 2.1. The lifting operators are used both to initialize the LBM and to find the ghost points of the LBM domain. Figure 5.1 shows the absolute differences by using as lifting operator the exact Chapman–Enskog expansion up to zeroth (top left), first (top right), second (bottom left), and third order (bottom right), respectively. Figure 5.2 shows the absolute differences with the lifting operator based on the CR algorithm in combination with Newton's method for a constant (top left), linear (top right), quadratic (bottom left), and cubic (bottom right) extrapolation in time, respectively. These results were obtained in [33] by considering local updates

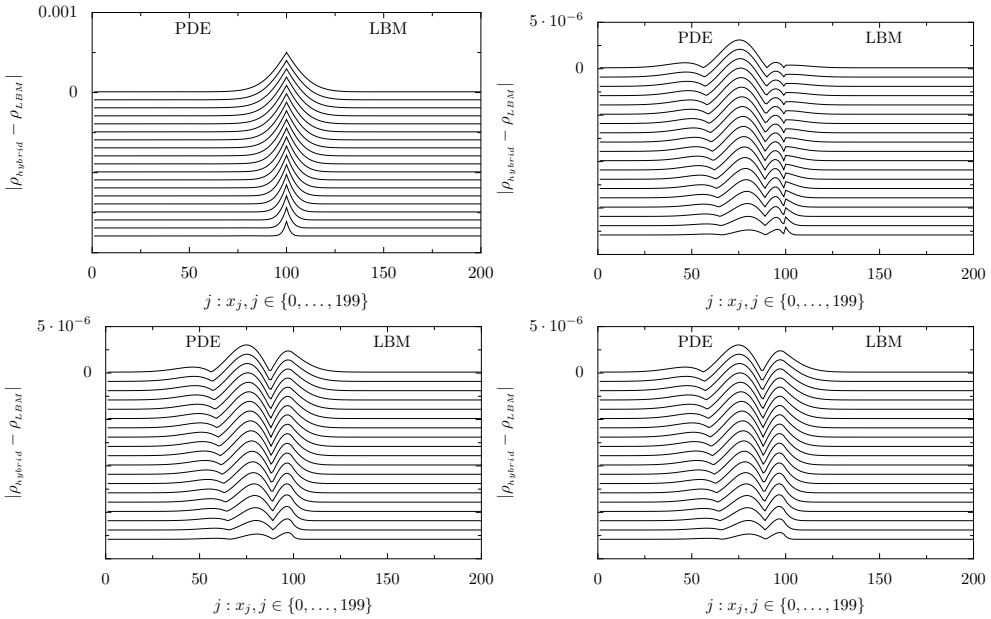

FIG. 5.2. $|\rho_{hybrid} - \rho_{LBM}|$ *after* 200 *time steps where the lifting operator based on the CR algorithm is used in combination with the method of Newton. We show results for constant (top left), linear (top right), quadratic (bottom left), and cubic (bottom right) backward extrapolation, respectively. The difference is also shown at earlier time slots but shifted down for clarity. The domain that is considered is shown in Figure* 2.1. *The model problem parameters are listed in Example* 1.

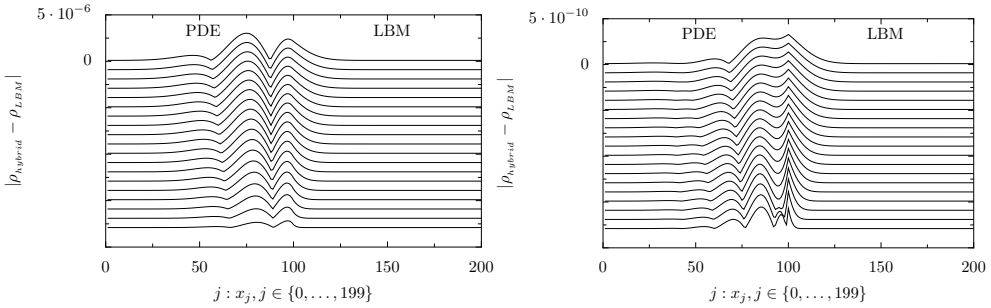

FIG. 5.3. $|\rho_{hybrid} - \rho_{LBM}|$ *after* 200 *time steps in the model problem of Example* 1. *The domain that is considered is shown in Figure* 2.1. *To deal with the initial error and the error in the ghost points of the LBM domain the numerical Chapman–Enskog expansion (order spatial expansion* 6*) is used. The considered PDE in the hybrid domain is the analytically known PDE* (2.4) *in the left figure and that obtained from the numerical Chapman–Enskog expansion in the right figure.*

at the ghost points of the LBM domain.

When the numerical Chapman–Enskog expansion (up to the sixth spatial derivative) is used in our one-dimensional hybrid model problem, Figure 5.3 is obtained. Here, we have two possibilities. First, act as if we know the PDE (2.4) obtained from the exact Chapman–Enskog expansion. $|\rho_{\text{hybrid}} - \rho_{\text{LBM}}|$ is given in the left of Figure 5.3 for which the hybrid domain is shown in Figure 2.1 and the PDE is the one given in (2.4). Second, use the PDE that is obtained from the proposed lifting

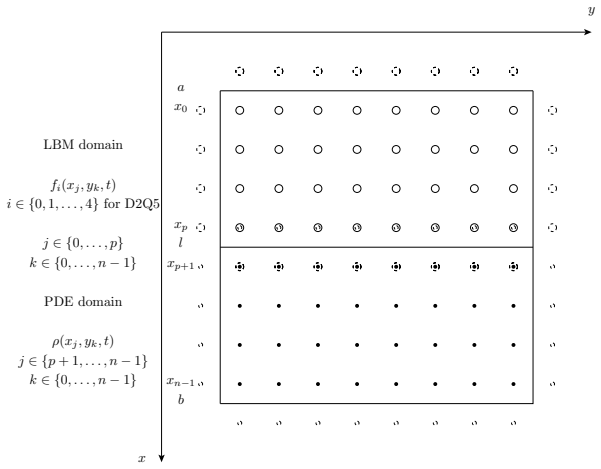

FIG. 5.4. *The two-dimensional spatial domain* $[a,b]\times[a,b] \subset \mathbb{R}^2$ *in the hybrid model is split into* $[a,l[\times[a,b]$*, on which we solve the LBM, and* $[l,b]\times[a,b]$*, on which we solve the PDE model. The solid points (*$\bullet$*) represent the grid for the density* ρ *of the discrete PDE; the circles (*$\circ$*) represent the LBM variables* $f_i(x,y,t)$*,* $i \in \{0,\ldots,4\}$*, for D2Q5. The periodic boundaries and the coupling are implemented with ghostcells which are drawn by dashed circles. The density in the ghostcells of the PDE domain, in* (x_p, y_k)*,* $k \in \{0,\ldots,n-1\}$*, and* (x_n, y_k)*, are found by taking* $\sum_i f_i$ *in* (x_p, y_k) *and* (x_0, y_k)*, respectively. However, the ghostcells for the LBM domain, in* (x_{-1}, y_k) *and* (x_{p+1}, y_k)*, require a lifting operator that lifts* ρ *to the distribution functions in these points.*

operator through summing the proposed lifting operator or considering the nullspace as explained in section 4.4. With this PDE, the result for $|\rho_{\text{hybrid}} - \rho_{\text{LBM}}|$ is shown in the right of Figure 5.3.

As can be seen in Figure 5.3, a change in the PDE—by considering the PDE obtained through the numerical Chapman–Enskog expansion—results in an even smaller modeling error compared to that obtained via the classical Chapman–Enskog expansion.

Changing the parameters of the model such that advection is included can also be considered and shows similar results. However, these are not shown in the paper.

5.3. Two-dimensional test problem. This section generalizes the previous one. Two spatial dimensions are considered. Two-dimensional problems can take different discrete sets of velocities into account. We show results for D2Q5, but similar results can be obtained for D2Q9.

The hybrid test domain for this section is represented in Figure 5.4 for D2Q5. Again, the domain is split into subdomains. One part of the domain is described by the LBM, while another part is described by a macroscopic PDE. Example 2 describes the parameters for the model problem in this two-dimensional setting.

EXAMPLE 2. *The considered model problem has the following parameters for a two-dimensional domain—described by five possible velocity directions (D2Q5)—of length* $L \times L$ *(with* n^2 *the number of grid points):*

$$L = 10, \quad n = 200, \quad \Delta x = \Delta y = \frac{L}{n}, \quad \Delta t = 0.0001, \quad \omega = 1.6129.$$

For these parameters the classical Chapman–Enskog expansion predicts a diffusion coefficient $D = 1$ *for the two-dimensional diffusion equation.*

The comparison of $|\rho_{\text{hybrid}} - \rho_{\text{LBM}}|$ is represented in Figure 5.5 for Example 2.

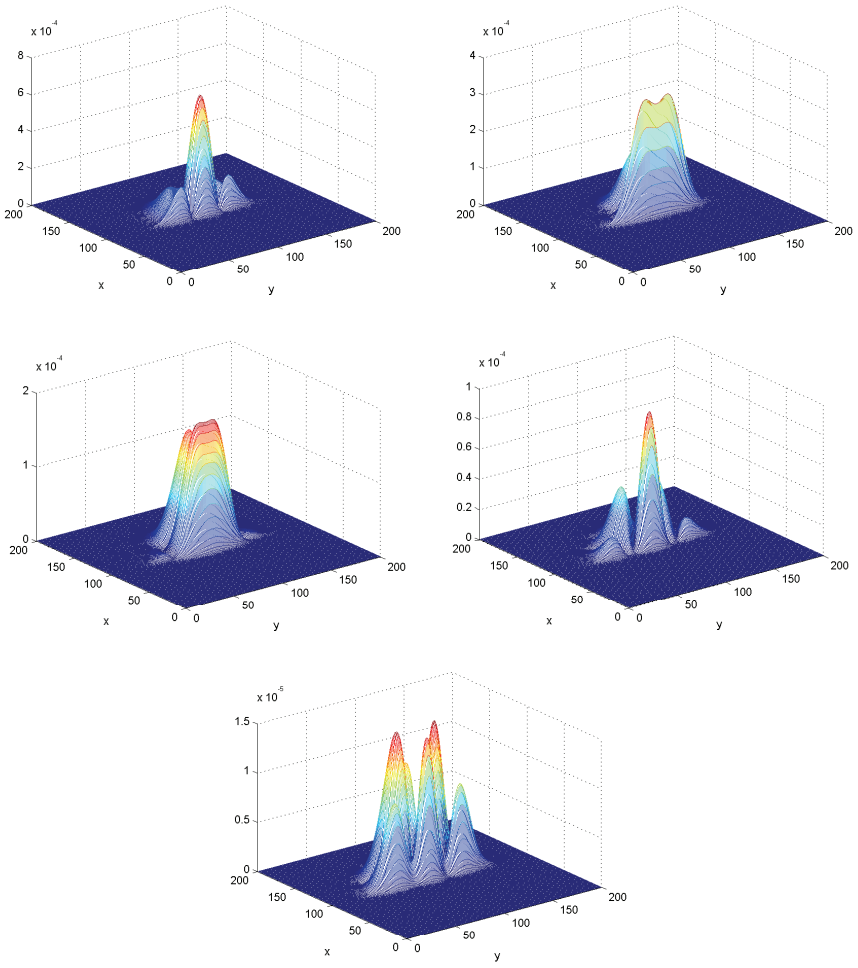

FIG. 5.5. *Comparison between different lifting operators for the model problem presented in Example* 2. *The absolute difference* $|\rho_{hybrid} - \rho_{LBM}|$ *is plotted at time step* 200. *To deal with the initial error and the error in the ghost points of the LBM we use the equilibrium distribution function (top left), the first order Chapman–Enskog expansion (top right), the second order Chapman–Enskog expansion (middle left), the numerical Chapman–Enskog expansion (order expansion* 4*) where the PDE in the hybrid domain is the analytically known PDE (middle right), and the numerical Chapman–Enskog expansion where the considered PDE in the hybrid domain is that obtained from the numerical Chapman–Enskog expansion (bottom).*

The different lifting operators are used to obtain distribution functions from a given density. The used lifting operators are the equilibrium distribution function in the top left figure, the first order Chapman–Enskog expansion in the top right, the second order Chapman–Enskog expansion in the middle left, and the numerical Chapman–Enskog expansion of spatial order expansion 4 in the middle right and the bottom—depending on the used PDE in the hybrid model.

5.4. Analysis of the computational cost of lifting. This section compares the computational cost of the lifting operators in the one-dimensional test problem (section 5.2).

TABLE 5.4

Analysis of the computational cost of the lifting operators in terms of the additional LBM steps. Both CR and the numerical Chapman–Enskog require additional LBM steps to lift ρ to the distribution function f. The table shows these additional steps for constructing the Jacobian operator for the Newton iteration. The listed values for the CR algorithm are even per ghost point and per time step, while those of the proposed lifting operator are the total number for the entire domain and for all time steps together since the vectors of constants needs to be determined only once and can be reused throughout the rest of the simulation.

Lifting operator	Number of iterations	LBM steps to perform one iteration	Total number of LBM steps
Exact Chapman–Enskog	/	/	0
CR algorithm			*per ghost point*
type of extrapolation in time			*per time step*
Constant	3	19×1	57
Linear	3	31×2	186
Quadratic	3	43×3	387
Cubic	3	55×4	660
Numerical Chapman–Enskog			*for entire domain*
with 18 *unknowns*			*and all time steps*
$\{\alpha, \beta, \delta, \epsilon, \theta, \iota\}$			
Constant	3/(4)	19	$57 + 2 = 59$
Linear	3/(4)	19×2	$114 + 2 = 116$
Quadratic	3/(4)	19×3	$171 + 2 = 173$
Cubic	3/(4)	19×4	$228 + 2 = 230$

The motivation for this paper is to bring down this cost. Especially, the CR algorithm requires many additional LBM steps to lift the density in the ghost points of the LBM domain, while numerical Chapman–Enskog requires only a single calculation with a fixed cost that can be done off-line before the simulation. This significantly reduces the cost of the lifting.

A detailed analysis of the lifting cost in terms of additional LBM steps is listed in Table 5.4. The table is an extension of that in [33] with results for the classical Chapman–Enskog expansion, the CR algorithm combined with Newton's method, and the numerical Chapman–Enskog expansion.

It can be seen that the total number of LBM steps for the CR algorithm is listed per ghost point and per time step. The number for the numerical Chapman–Enskog expansion is the total for the entire domain and at all time steps since the calculations for the coefficients are done off-line.

In two-dimensional problems the numerical Chapman–Enskog expansion still has a limited computational cost. Only a few additional coefficients need to be determined associated with the extra spatial derivatives.

Note that, once the coefficients of the numerical Chapman–Enskog expansion are determined, the computational cost of applying the numerical Chapman–Enskog lifting operator is the same as applying the analytical Chapman–Enskog operator. For each grid point we need the derivatives of ρ, which can be calculated by finite differences using the densities at neighboring grid points.

6. Conclusions. This article proposes a numerical lifting operator for LBMs that maps a given density to the corresponding distribution functions. This new lifting operator is based on the Chapman–Enskog expansion that writes the missing distribution functions as analytical series of the density and its derivatives. The

coefficients of this expansion are now determined through a numerical method, in contrast to the original expansion, where they are found analytically. The numerical method is based on the CR algorithm that relies on the attraction of the dynamics toward the slow manifold.

A systematic numerical comparison of the accuracy and the computational cost among the analytical Chapman–Enskog expansion, the CR algorithm, and the new lifting operator is performed in this article. The cheapest way to lift is with the Chapman–Enskog expansion. However, the analytical expressions are not always available for the system of interest. An alternative numerical lifting operator is CR, but its computational cost grows significantly with the order of accuracy. It needs many additional LBM steps to find the missing distribution functions.

The new result and the main focus of this paper is a numerical lifting method that combines the ideas of CR and the Chapman–Enskog expansion. Instead of using CR to find for each grid point the missing moments, we use CR to find the unknown coefficients of the Chapman–Enskog expansion. This numerical lifting method has several advantages. First, it significantly reduces the number of unknowns in the lifting step: only the coefficients are needed rather than the full state $f(x, v, t)$. Second, it can be done off-line before the calculations. Indeed, once the coefficients are found they can be reused every time step and every grid point to realize the lifting at no significant additional cost. A third advantage is that the expansion gives, as a spin-off, the transport coefficients of the macroscopic PDE.

The new lifting operator, the numerical Chapman–Enskog expansion, is then used in a hybrid domain that spatially couples a macroscopic PDE with an LBM. This creates a missing data problem at the interfaces since the PDE model has too few variables to provide the LBM with the correct boundary conditions.

The numerical Chapman–Enskog expansion deals with this mismatch in variables. It maps the variables of the PDE model to those of the LBM. We evaluate and compare various lifting operators. In particular, we focus on a simple LBM and PDE model discretized with equal grid and time steps such that the error created by the coupling can be highlighted. The paper presents numerical results both for one- and two-dimensional hybrid domains where part of the LBM domain is replaced by the macroscopic PDE. In both cases the error associated with the coupling can be made smaller than the modeling error, related to the PDE approximation of the LBM.

This paper reports on our initial efforts, where we have focused on a simple model problem with several limiting assumptions. In the model we have assumed an equilibrium distribution function that depends only on the local density, while in general it also depends on the local momentum and temperature. We believe that this limitation can be alleviated by considering a Chapman–Enskog expansion with a more general equilibrium function.

A further assumption is that we used the same time and space grid for the PDE and the LBM. This choice was made to highlight the error made by the coupling mechanism, the ease of implementation, and to eliminate the error due to the different discretizations. However, there is no reason to prohibit different grid and time spacings. Extra care is then needed to interpolate between time and grid spacings. In practice, the grid of the PDE can be further coarsened, depending on local discretization errors. Ideally, the hybrid model is embedded in an adaptive mesh refinement simulation, where at the finest level an LBM is used.

We have also kept the boundary between the LBM and the PDE domain fixed during the simulation at an arbitrary position. In the future, this boundary should be moved adaptively using an accuracy requirement based on the properties of the

lifting operator.

For the model problem with periodic boundary conditions studied in this paper, the Chapman–Enskog expansion exists everywhere and we could in principle put the boundary between the PDE and the Boltzmann model at any location, provided that we lift accurately. For general Boltzmann models, with complicated collision integral operators, such a Chapman–Enskog expansion might not exist everywhere in the domain. Then a hybrid model can be constructed where a PDE can replace the Boltzmann model only in the regions where the Chapman–Enskog expansion is known to exist.

This situation appears in the modeling of laser ablation, where a laser heats a surface that consequently melts and evaporates. The escaping plasma plume can be described by a Boltzmann equation. Close to the melting surface a complicated nonequilibrium situation appears where escaping particles evaporate but particles that impinge on the melted surface condensate. There is no Chapman–Enskog expansion that can describe this situation close to the surface. Only away from the surface the plasma reaches an equilibrium situation. A hybrid model will then use a full Boltzmann model near melt, while a reduced PDE model can be used away from the surface.

Acknowledgments. We would like to acknowledge the fruitful comments and suggestions made by the editor and referees.

REFERENCES

[1] C. K. Aidun and J. R. Clausen, *Lattice-Boltzmann method for complex flows*, in Annual Review of Fluid Mechanics, Annu. Rev. Fluid Mech. 42, Annual Reviews, Palo Alto, CA, 2010, pp. 439–472.

[2] M. Banda, A. Klar, L. Pareschi, and M. Seaïd, *Lattice-Boltzmann type relaxation systems and high order relaxation schemes for the incompressible Navier-Stokes equations*, Math. Comp., 77 (2008), pp. 943–965.

[3] R. Benzi, S. Succi, and M. Vergassola, *The lattice Boltzmann equation: Theory and applications*, Phys. Rep., 222 (1992), pp. 145–197.

[4] J. F. Bourgat, P. Le Tallec, B. Perthame, and Y. Qiu, *Coupling Boltzmann and Euler equations without overlapping*, in Domain Decomposition Methods in Science and Engineering, Contemp. Math. 157, AMS, Providence, RI, 1994, pp. 377–398.

[5] C. Cercignani, *The Boltzmann Equation and Its Applications*, Springer, Berlin, 1988.

[6] S. Chapman and T. G. Cowling, *The Mathematical Theory of Non-uniform Gases*, Cambridge University Press, Cambridge, UK, 1953.

[7] B. Chopard and M. Droz, *Cellular Automata Modeling of Physical Systems*, Cambridge University Press, Cambridge, UK, 1998.

[8] P. Degond, G. Dimarco, and L. Mieussens, *A moving interface method for dynamic kinetic-fluid coupling*, J. Comput. Phys., 227 (2007), pp. 1176–1208.

[9] P. Degond, G. Dimarco, and L. Mieussens, *A multiscale kinetic-fluid solver with dynamic localization of kinetic effects*, J. Comput. Phys., 229 (2010), pp. 4907–4933.

[10] G. Dimarco and L. Pareschi, *Hybrid multiscale methods* II. *Kinetic equations*, Multiscale Model. Simul., 6 (2008), pp. 1169–1197.

[11] A. Dupuis, E. M. Kotsalis, and P. Koumoutsakos, *Coupling lattice Boltzmann and molecular dynamics models for dense fluids*, Phys. Rev. E (3), 75 (2007), 046704.

[12] E. G. Flekkøy, G. Wagner, and J. Feder, *Hybrid model for combined particle and continuum dynamics*, Europhys. Lett., 52 (2000), pp. 271–276.

[13] A. L. Garcia, J. B. Bell, W. Y. Crutchfield, and B. J. Alder, *Adaptive mesh and algorithm refinement using direct simulation Monte Carlo*, J. Comput. Phys., 154 (1999), pp. 134–155.

[14] C. W. Gear, T. J. Kaper, I. G. Kevrekidis, and A. Zagaris, *Projecting to a slow manifold: Singularly perturbed systems and legacy codes*, SIAM J. Appl. Dyn. Syst., 4 (2005), pp. 711–732.

[15] M. Junk, A. Klar, and L. Luo, *Asymptotic analysis of the lattice Boltzmann equation*, J.

Comput. Phys., 210 (2005), pp. 676–704.
[16] B. D. Kandhai, *Large Scale Lattice-Boltzmann Simulations: Computational Methods and Applications*, Ph.D. thesis, Universiteit van Amsterdam, Amsterdam, The Netherlands, 1999.
[17] I. G. Kevrekidis, C. W. Gear, J. M. Hyman, P. G. Kevrekidis, O. Runborg, and C. Theodoropoulos, *Equation-free, coarse-grained multiscale computation: Enabling microscopic simulators to perform system-level analysis*, Commun. Math. Sci., 1 (2003), pp. 715–762.
[18] J. Latt, B. Chopard, and P. Albuquerque, *Spatial coupling of a lattice Boltzmann fluid model with a finite difference Navier-Stokes solver*, arXiv:physics/0511243v1 [physics. comp-ph], 2008.
[19] P. Le Tallec and F. Mallinger, *Coupling Boltzmann and Navier-Stokes equations by half fluxes*, J. Comput. Phys., 136 (1997), pp. 51–67.
[20] M. Mehl, T. Neckel, and P. Neumann, *Navier-Stokes and Lattice-Boltzmann on octree-like grids in the Peano framework*, Internat. J. Numer. Methods Fluids, 65 (2011), pp. 67–86.
[21] R. Mei, L. Luo, P. Lallemand, and D. d'Humières, *Consistent initial conditions for lattice Boltzmann simulations*, Comput. & Fluids, 35 (2006), pp. 855–862.
[22] M. L. Parks and R. B. Lehoucq, *Atomistic-to-Continuum Coupling*, SIAM News, 39 (2006).
[23] M. Rheinländer, *On the stability structure for lattice Boltzmann schemes*, Comput. Math. Appl., 59 (2010), pp. 2150–2167.
[24] J. D. Sterling and S. Chen, *Stability analysis of lattice Boltzmann methods*, J. Comput. Phys., 123 (1996), pp. 196–206.
[25] S. Succi, *The Lattice Boltzmann Equation for Fluid Dynamics and Beyond*, Oxford University Press, Oxford, UK, 2001.
[26] S. Succi, O. Filippova, G. Smith, and E. Kaxiras, *Applying the lattice Boltzmann equation to multiscale fluid problems*, Comput. Sci. Eng., 3 (2001), pp. 26–37.
[27] R. G. M. Van der Sman and M. H. Ernst, *Convection-diffusion lattice Boltzmann scheme for irregular lattices*, J. Comput. Phys., 160 (2000), pp. 766–782.
[28] P. Van Leemput, *Multiscale and Equation-Free Computing for Lattice Boltzmann Models*, Ph.D. thesis, K.U. Leuven, Leuven, Belgium, 2007.
[29] P. Van Leemput, C. Vandekerckhove, W. Vanroose, and D. Roose, *Accuracy of hybrid lattice Boltzmann/finite difference schemes for reaction-diffusion systems*, Multiscale Model. Simul., 6 (2007), pp. 838–857.
[30] P. Van Leemput, W. Vanroose, and D. Roose, *Mesoscale analysis of the equation-free constrained runs initialization scheme*, Multiscale Model. Simul., 6 (2008), pp. 1234–1255.
[31] C. Vandekerckhove, *Macroscopic Simulation of Multiscale Systems within the Equation-Free Framework*, Ph.D. thesis, K.U. Leuven, Leuven, Belgium, 2008.
[32] C. Vandekerckhove, I. Kevrekidis, and D. Roose, *An efficient Newton-Krylov implementation of the constrained runs scheme for initializing on a slow manifold*, J. Sci. Comput., 39 (2009), pp. 167–188.
[33] Y. Vanderhoydonc and W. Vanroose, *Lifting in hybrid lattice Boltzmann and PDE models*, Comput. Vis. Sci., 14 (2011), pp. 67–78.
[34] D. A. Wolf-Gladrow, *Lattice-Gas Cellular Automata and Lattice Boltzmann Models*, Springer, Berlin, 2000.