

Research Article

High-Accuracy Approximation of High-Rank Derivatives: Isotropic Finite Differences Based on Lattice-Boltzmann Stencils

Keijo Kalervo Mattila,¹ Luiz Adolfo Hegele Júnior,² and Paulo Cesar Philippi¹

¹ *Laboratory of Porous Media and Thermophysical Properties, Mechanical Engineering Department, Federal University of Santa Catarina, 88040-900 Florianópolis, SC, Brazil*

² *Department of Petroleum Engineering, State University of Santa Catarina, 88330-668 Balneário Camboriú, SC, Brazil*

Correspondence should be addressed to Keijo Kalervo Mattila; keijo.mattila@lmpt.ufsc.br

Received 8 August 2013; Accepted 12 October 2013; Published 29 January 2014

Academic Editors: J. Banaś and C. Yiu

Copyright © 2014 Keijo Kalervo Mattila et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We propose isotropic finite differences for high-accuracy approximation of high-rank derivatives. These finite differences are based on direct application of lattice-Boltzmann stencils. The presented finite-difference expressions are valid in any dimension, particularly in two and three dimensions, and any lattice-Boltzmann stencil isotropic enough can be utilized. A theoretical basis for the proposed utilization of lattice-Boltzmann stencils in the approximation of high-rank derivatives is established. In particular, the isotropy and accuracy properties of the proposed approximations are derived directly from this basis. Furthermore, in this formal development, we extend the theory of Hermite polynomial tensors in the case of discrete spaces and present expressions for the discrete inner products between monomials and Hermite polynomial tensors. In addition, we prove an equivalency between two approaches for constructing lattice-Boltzmann stencils. For the numerical verification of the presented finite differences, we introduce 5th-, 6th-, and 8th-order two-dimensional lattice-Boltzmann stencils.

1. Introduction

The approximation of derivatives by finite differences is the cornerstone of numerical computing. Forward, backward, and central differences, the five-point stencil for approximating Laplacian in a two-dimensional domain, and the numerical analysis of the convergence rate of the related approximation errors, based on the application of Taylor series, require no introduction for anyone working in the field of scientific computing.

Construction of finite-difference stencils for the approximation of high-rank derivatives in two or three dimensions, say gradient of Laplacian or biLaplacian, will already be a more advanced topic—even if achieved by solving a modest linear system of equations. A further complication is introduced when requiring an isotropic approximation of derivatives. More specifically, when the leading-order error term of the finite difference approximation is required to be an isotropic expression or, in other words, to be free of directional bias. Such a property may be essential, for example, when solving certain partial differential equations.

Conventional finite differences are not isotropic in the above sense. Isotropic finite differences of second-order accuracy for the approximation of first and second derivatives, both in two and three dimensions, together with a systematic procedure for constructing the differences, are presented in [1]. Patra and Karttunen proceed further: they present up to fourth-order accurate isotropic stencils, in two and three dimensions, for the numerical computation of second, third, and fourth derivatives [2].

In the context of lattice Boltzmann methods, isotropic finite differences have been well known for some time, mainly because of their importance in the approximation of interparticle forces in multiphase and multicomponent models. For example, in the so-called Shan-Chen multiphase model [3], as was remarked by Yuan and Schaefer [4], the originally proposed approximation of interparticle forces includes an isotropic finite-difference approximation of the gradient of the interaction potential. In fact, when the standard D2Q9 lattice-Boltzmann stencil is used, the approximation is equivalent to the one proposed by Kumar [1].

Recently new efforts were undertaken to construct, or define, isotropic finite-difference stencils by utilizing the lattice-Boltzmann method framework. Namely, in the construction of lattice-Boltzmann stencils, a set of weights and discrete velocity vectors have to be found, together with a scaling factor related to the lattice speed of sound, in such a way that these weights and velocities will fulfill isotropy conditions up to a given order—a property ensuring a correspondence between continuous and discrete moments of the equilibrium distribution functions [5–7]. For example, in [8–10], this construction procedure is adopted in order to produce isotropic finite-difference stencils: there second-order accurate approximations are presented, but the isotropy goes beyond the leading order error term.

Lee et al. present a different strategy: they propose to approximate derivatives in 2D and 3D by taking moments of the conventional 1D finite differences along the characteristic lines—the moments are isotropic finite-differences [11, 12]. In the computation of these moments, the weights, discrete velocity vectors, and the scaling factor of a lattice Boltzmann stencil are utilized. Philippi et al. present up to fourth-order accurate isotropic finite difference stencils, constructed directly with the weights and velocity vectors of a given lattice Boltzmann stencil, for the approximation of gradient, Laplacian, and gradient of Laplacian [13]. Independently of Philippi et al., but by adopting the same approach, second-order accurate isotropic finite-difference approximation of Laplacian was proposed by Thampi et al. [14]. In [15], following the work of Thampi et al., approximations for the divergence and curl were presented together with fourth-order approximations for the gradient and Laplacian.

Naturally, in both approaches by Lee and Philippi, the validity of the presented expressions for the isotropic finite differences, and their isotropy properties, depends directly on the order of isotropy fulfilled by the lattice Boltzmann stencil utilized. On the other hand, these approaches provide beauty in abstraction: expressions for the finite differences are valid regardless of dimension, and any lattice Boltzmann stencil isotropic enough can be utilized—a family of finite differences is defined with a single expression.

Here our purpose is to go further by presenting isotropic finite differences, based on the direct utilization of lattice Boltzmann stencils, for the high-accuracy approximation of high-rank derivatives. In the construction of these isotropic differences we make use of the Hermite polynomial tensors in a very similar manner in which they are used in the construction of so-called kinetic projectors. Theory of Hermite polynomial tensors was first introduced by Grad in his innovative article [16]. This formal approach allows us to have very compact and abstract expressions for the coefficients of the stencils. It also allows the isotropy and accuracy properties of the proposed approximations to be derived theoretically. After these formal developments, we resort to simple calculus and construct, hierarchically, higher-order accurate stencils for the approximation of high-rank derivatives.

First, an introduction to Hermite polynomial tensors, monomials and their properties is presented in Section 2 using a specific notation. The fundamentals of lattice-Boltzmann schemes and stencils are then introduced

in Section 3. Also in this section, 5th-, 6th-, and 8th-order lattice-Boltzmann stencils for two dimensions are introduced. The theory of using Hermite polynomial tensors together with the weights, velocity vectors, and the scaling factor for the construction of isotropic differences is presented in Section 4. This is followed by the hierarchical calculus of higher-order accurate stencils in Section 5. Application of the proposed stencils is discussed in Section 6. The discussion involves explicit expressions for the coefficients of some finite-difference approximations; these coefficients are then compared with other corresponding approximations proposed in the literature. The new lattice-Boltzmann stencils, introduced in Section 3, are utilized in Section 7 for the numerical verification of the proposed finite differences. Finally, a short conclusion is presented.

2. Theoretical Background

The theory of lattice-Boltzmann method, as well as the theory of isotropic finite differences proposed in this work, relies on Hermite polynomial tensors, monomials, and their properties. In this section, these fundamental concepts are defined using a specific notation explained below. This section is strongly based on the work of Grad [16].

2.1. Notational Conventions. The presentation of explicit expressions for Hermite polynomial tensors, or the related mathematical derivations, in an unambiguous yet comprehensible way is not an easy task because the expressions involve combinations. Here we will adopt the notation used by Grad in his original article with a minor modification. Namely, let the nonstandard operation

$$A_{\alpha_1 \dots \alpha_k}^{(k)} \otimes B_{\beta_1 \dots \beta_{n-k}}^{(n-k)} \equiv \mathbf{A}^{(k)} \otimes \mathbf{B}^{(n-k)} \quad (1)$$

denote summation of tensor products over all possible combinations of indices: the number of combinations is $n!/(k!(n-k)!)$, where n is the rank of the tensor products. For example,

$$\begin{aligned} \mathbf{A}^{(2)} \otimes \mathbf{B}^{(2)} &\equiv A_{\alpha\beta}^{(2)} \otimes B_{\gamma\delta}^{(2)} = A_{\alpha\beta}^{(2)} B_{\gamma\delta}^{(2)} + A_{\alpha\gamma}^{(2)} B_{\beta\delta}^{(2)} + A_{\alpha\delta}^{(2)} B_{\beta\gamma}^{(2)} \\ &\quad + A_{\beta\gamma}^{(2)} B_{\alpha\delta}^{(2)} + A_{\beta\delta}^{(2)} B_{\alpha\gamma}^{(2)} + A_{\gamma\delta}^{(2)} B_{\alpha\beta}^{(2)}. \end{aligned} \quad (2)$$

The notation is a shorthand, and it always has precedence over all other mathematical operations and manipulations. A similar notation was recently adopted in [17] for the same purpose of facilitating mathematical treatments.

Furthermore, the subscripts with bold typesetting, \mathbf{p} and \mathbf{q} , denote separate nonempty index sets; that is, $\mathbf{p} = \{\alpha_1 \dots \alpha_k\}$ and $\mathbf{q} = \{\beta_1 \dots \beta_l\}$; \mathbf{pq} denotes their union and the number of indices in each set is implied by the context. The operator \otimes is used in conjunction with these index sets: the implied summations over index combinations are only over the shared index set \mathbf{q} —the index set \mathbf{p} is considered as fixed. For example, when $\mathbf{p} = \{\alpha\}$ and $\mathbf{q} = \{\beta\gamma\delta\}$,

$$\mathbf{A}_{\mathbf{pq}}^{(2)} \otimes \mathbf{B}_{\mathbf{q}}^{(2)} = A_{\alpha\beta}^{(2)} B_{\gamma\delta}^{(2)} + A_{\alpha\gamma}^{(2)} B_{\beta\delta}^{(2)} + A_{\alpha\delta}^{(2)} B_{\beta\gamma}^{(2)} \quad (3)$$

or when $\mathbf{p} = \{\alpha\}$ and $\mathbf{q} = \{\beta\gamma\delta\epsilon\}$,

$$\begin{aligned} \mathbf{A}_{\mathbf{pq}}^{(3)} \otimes \mathbf{B}_{\mathbf{q}}^{(2)} &= A_{\alpha\beta\gamma}^{(3)} B_{\delta\epsilon}^{(2)} + A_{\alpha\beta\delta}^{(3)} B_{\gamma\epsilon}^{(2)} + A_{\alpha\beta\epsilon}^{(3)} B_{\gamma\delta}^{(2)} \\ &+ A_{\alpha\gamma\delta}^{(3)} B_{\beta\epsilon}^{(2)} + A_{\alpha\gamma\epsilon}^{(3)} B_{\beta\delta}^{(2)} + A_{\alpha\delta\epsilon}^{(3)} B_{\beta\gamma}^{(2)}. \end{aligned} \quad (4)$$

The appendix of [6] provides an alternative notation for the same operations.

With $\Delta^{(n)} \equiv \Delta_{\alpha_1 \dots \alpha_{2n}}^{(n)}$ we denote the generalized Kronecker delta; these isotropic tensors are symmetric with respect to all of their $2n$ subscripts. The first generalized Kronecker deltas are

$$\begin{aligned} \Delta^{(0)} &= 1, \\ \Delta_{\alpha\beta}^{(1)} &= \delta_{\alpha\beta}, \\ \Delta_{\alpha\beta\gamma\delta}^{(2)} &= (\delta_{\alpha\beta}\delta_{\gamma\delta} + \delta_{\alpha\gamma}\delta_{\beta\delta} + \delta_{\alpha\delta}\delta_{\beta\gamma}), \\ \Delta_{\alpha\beta\gamma\delta\epsilon\zeta}^{(3)} &= (\delta_{\alpha\beta}\Delta_{\gamma\delta\epsilon\zeta}^{(2)} + \delta_{\alpha\gamma}\Delta_{\beta\delta\epsilon\zeta}^{(2)} \\ &+ \delta_{\alpha\delta}\Delta_{\beta\gamma\epsilon\zeta}^{(2)} + \delta_{\alpha\epsilon}\Delta_{\beta\gamma\delta\zeta}^{(2)} + \delta_{\alpha\zeta}\Delta_{\beta\gamma\delta\epsilon}^{(2)}), \\ \Delta_{\alpha\beta\gamma\delta\epsilon\zeta\eta\theta}^{(4)} &= (\delta_{\alpha\beta}\Delta_{\gamma\delta\epsilon\zeta\eta\theta}^{(3)} + \delta_{\alpha\gamma}\Delta_{\beta\delta\epsilon\zeta\eta\theta}^{(3)} + \delta_{\alpha\delta}\Delta_{\beta\gamma\epsilon\zeta\eta\theta}^{(3)} \\ &+ \delta_{\alpha\epsilon}\Delta_{\beta\gamma\delta\zeta\eta\theta}^{(3)} + \delta_{\alpha\zeta}\Delta_{\beta\gamma\delta\epsilon\eta\theta}^{(3)} + \delta_{\alpha\eta}\Delta_{\beta\gamma\delta\epsilon\zeta\theta}^{(3)} \\ &+ \delta_{\alpha\theta}\Delta_{\beta\gamma\delta\epsilon\zeta\eta}^{(3)}). \end{aligned} \quad (5)$$

Their expressions are given by the recurrence relation

$$\Delta_{\mathbf{pq}}^{(n)} = \Delta_{\mathbf{pq}}^{(1)} \otimes \Delta_{\mathbf{q}}^{(n-1)}. \quad (6)$$

The number of separate terms in the expressions for the generalized Kronecker delta, if written only by using the standard Kronecker delta symbol, is given by the so-called double factorial $(2n)!/(2^n n!)$. For example, $\Delta^{(4)}$ and $\Delta^{(8)}$ involve 105 and 2 027 025 separate terms, respectively.

In addition, by $\Lambda^{(n)} \equiv \Lambda_{\alpha_1 \dots \alpha_n \beta_1 \dots \beta_n}^{(n)}$ we denote rank $2n$ orthogonality tensor. The first orthogonality tensors are

$$\begin{aligned} \Lambda^{(0)} &= 1, \\ \Lambda_{\alpha_1\beta_1}^{(1)} &= \delta_{\alpha_1\beta_1}, \\ \Lambda_{\alpha_1\alpha_2\beta_1\beta_2}^{(2)} &= (\delta_{\alpha_1\beta_1}\delta_{\alpha_2\beta_2} + \delta_{\alpha_1\beta_2}\delta_{\alpha_2\beta_1}), \\ \Lambda_{\alpha_1\alpha_2\alpha_3\beta_1\beta_2\beta_3}^{(3)} &= (\delta_{\alpha_1\beta_1}\delta_{\alpha_2\beta_2}\delta_{\alpha_3\beta_3} + \delta_{\alpha_1\beta_1}\delta_{\alpha_2\beta_3}\delta_{\alpha_3\beta_2} \\ &+ \delta_{\alpha_1\beta_2}\delta_{\alpha_2\beta_1}\delta_{\alpha_3\beta_3} + \delta_{\alpha_1\beta_2}\delta_{\alpha_2\beta_3}\delta_{\alpha_3\beta_1} \\ &+ \delta_{\alpha_1\beta_3}\delta_{\alpha_2\beta_1}\delta_{\alpha_3\beta_2} + \delta_{\alpha_1\beta_3}\delta_{\alpha_2\beta_2}\delta_{\alpha_3\beta_1}). \end{aligned} \quad (7)$$

More generally, $\Lambda^{(n)}$ is a sum of $n!$ terms where each term is a product of n Kronecker deltas; the sum is over permutation of the indices so that each Kronecker delta has always one

index from the set α_k and the other from β_l . The orthogonality tensors can be defined by the recurrence relation

$$\Lambda_{\alpha_1 \dots \alpha_n \beta_1 \dots \beta_n}^{(n)} = \Delta_{\alpha_1 \beta_1}^{(1)} \otimes \Lambda_{\alpha_2 \dots \alpha_n \beta_2 \dots \beta_n}^{(n-1)}. \quad (8)$$

The tensor $\Lambda^{(n)}$ can also be regarded as a symmetry operator. That is, a tensor product between an arbitrary rank n tensor and $\Lambda^{(n)}$ will extract the symmetric part of that tensor. When the arbitrary tensor $\mathbf{A}^{(n)}$ is already symmetric, the product is (Einstein summation convention is applied)

$$\Lambda_{\alpha_1 \dots \alpha_n \beta_1 \dots \beta_n}^{(n)} A_{\beta_1 \dots \beta_n}^{(n)} = n! A_{\alpha_1 \dots \alpha_n}^{(n)}. \quad (9)$$

Finally, a summation over index combinations can be split into two separate summations. Let $\mathbf{A}^{(k)}$ and $\mathbf{B}^{(m)}$ again denote arbitrary tensors, and \mathbf{p} is assumed to be a single index set. Then,

$$(\mathbf{A}^{(k)} \otimes \mathbf{B}^{(m)})_{\mathbf{pq}} = \mathbf{A}_{\mathbf{pq}}^{(k)} \otimes \mathbf{B}_{\mathbf{q}}^{(m)} + \mathbf{B}_{\mathbf{pq}}^{(m)} \otimes \mathbf{A}_{\mathbf{q}}^{(k)}, \quad (10)$$

where the left-hand side involves all possible index combinations from the set \mathbf{pq} .

2.2. Hermite Polynomial Tensors, Monomials, and Their Properties. Let us define a monomial tensor $\mathbf{M}^{(n)}(\mathbf{v}) \equiv \mathcal{M}_{\alpha_1 \dots \alpha_n}^{(n)} := v_{\alpha_1} \dots v_{\alpha_n}$; notation $\mathbf{M}^{(n)}$ will always imply an argument \mathbf{v} . Hermite polynomial tensors $\mathbf{H}^{(n)} \equiv \mathcal{H}_{\alpha_1 \dots \alpha_n}^{(n)}$ are defined by the formula

$$\begin{aligned} \mathbf{H}^{(n)}(\mathbf{v}) &= \frac{(-1)^n}{\omega(\mathbf{v})} \nabla_{\mathbf{v}}^{(n)} \omega(\mathbf{v}), \\ \omega(\mathbf{v}) &= \frac{1}{(2\pi)^{D/2}} e^{-(\mathbf{v}^2/2)}, \quad \nabla_{\mathbf{v}}^{(n)} \equiv \partial_{\alpha_1} \dots \partial_{\alpha_n}, \end{aligned} \quad (11)$$

where ω is a weighting function, D refers to the spatial dimension, and $\mathbf{v}^2 = v_{\alpha} v_{\alpha}$. Like with monomials, notation $\mathbf{H}^{(n)}$ will always imply an argument \mathbf{v} . Furthermore, a rank n Hermite polynomial tensor is also a polynomial of degree n . The explicit expressions for the first few Hermite polynomial tensors are given in Appendix A.

An important property related to the weighting function is

$$\int \omega(\mathbf{v}) \mathbf{M}^{(2n)} d\mathbf{v} = \Delta^{(n)}. \quad (12)$$

Equation (12) defines the moments of the weighting function; all odd order moments vanish. Clearly, (12) can also be regarded as an expression for the weighted tensorial inner product of two monomials. An analogous expression for the inner product of Hermite polynomial tensors is

$$\begin{aligned} \int \omega(\mathbf{v}) \mathbf{H}^{(n)} \mathbf{H}^{(m)} d\mathbf{v} &\equiv \int \omega(\mathbf{v}) H_{\alpha_1 \dots \alpha_n}^{(n)} H_{\beta_1 \dots \beta_m}^{(m)} d\mathbf{v} \\ &= \begin{cases} \mathbf{0}, & n \neq m, \\ \Delta^{(n)}, & n = m. \end{cases} \end{aligned} \quad (13)$$

This means that not only Hermite polynomial tensors of different rank are orthogonal, but also distinct polynomials of the same degree.

In fact, an explicit formula for Hermite polynomial tensors is given by a simple summation:

$$\mathbf{H}^{(2n+s)} = \sum_{k=0}^n (-1)^k \mathbf{M}^{(2(n-k)+s)} \otimes \Delta^{(k)}, \quad (14)$$

where $s = 0$ and $s = 1$ define even and odd tensors, respectively. A useful recurrence relation for Hermite polynomial tensors is

$$\mathbf{v}_p \mathbf{H}_q^{(n)} = \mathbf{H}_{pq}^{(n+1)} + \Delta_{pq}^{(1)} \otimes \mathbf{H}_q^{(n-1)}. \quad (15)$$

The recurrence relation is very convenient for a computer implementation of high-rank Hermite polynomial tensors. Useful mathematical tools are provided also by the two relations

$$\nabla_p^{(1)} \mathbf{H}_q^{(n)} = \Delta_{pq}^{(1)} \otimes \mathbf{H}_q^{(n-1)}, \quad \nabla_p^{(1)} \mathbf{M}_q^{(n)} = \Delta_{pq}^{(1)} \otimes \mathbf{M}_q^{(n-1)}. \quad (16)$$

2.3. Inner Products in Discrete Spaces. Above we presented weighted tensorial inner products between two monomials and between two Hermite polynomial tensors in a continuous space. However, numerical methods inherently involve discrete spaces. Hence, it is of computational interest to construct discrete representations for the continuous space admitting the above inner products. The construction of such a discrete representation is basically a quadrature problem: find the discrete weights w_i and abscissas \mathbf{v}_i so that

$$\sum_i w_i \mathbf{M}_i^{(2n)} = \Delta^{(n)} = \int \omega(\mathbf{v}) \mathbf{M}^{(2n)} d\mathbf{v} \quad (17)$$

or

$$\sum_i w_i \mathbf{H}_i^{(n)} \mathbf{H}_i^{(n)} = \Lambda^{(n)} = \int \omega(\mathbf{v}) \mathbf{H}^{(n)} \mathbf{H}^{(n)} d\mathbf{v} \quad (18)$$

is satisfied up to a given order N ; in the general solution $w_i \neq \omega(\mathbf{v}_i)$. In fact, in Appendix B we prove that a solution to either of the two quadrature problems will guarantee both (17) and (18).

Furthermore, matching the inner products of two Hermite polynomial tensors of equal rank, up to a given order N , will guarantee orthogonality of the Hermite polynomial tensors in the discrete space:

$$\sum_i w_i \mathbf{H}_i^{(n)} \mathbf{H}_i^{(m)} = \begin{cases} \mathbf{0}, & n \neq m, \\ \Lambda^{(n)}, & n = m, \end{cases} \quad n + m \leq 2N + 1. \quad (19)$$

In other words, the inner products of two Hermite polynomial tensors, not necessarily of equal rank, will match in the discrete and continuous spaces. This has been shown in the appendix of [5].

Now we will proceed to the discrete inner product between a monomial and a Hermite polynomial tensor. In

Appendix A, explicit expressions are given for the first few monomials as linear combinations of Hermite polynomial tensors. In fact, these expressions are given by a simple summation formula:

$$\mathbf{M}^{(2n+s)} = \sum_{k=0}^n \mathbf{H}^{(2(n-k)+s)} \otimes \Delta^{(k)}; \quad (20)$$

even and odd tensors are defined by $s = 0$ and $s = 1$, respectively. A proof for this summation formula is given in Appendix C. Since the Hermite polynomial tensor expansion for the monomials is now available, we find that

$$\sum_i w_i \mathbf{H}_i^{(n)} \mathbf{M}_i^{(m)} = \sum_i w_i \mathbf{H}_i^{(n)} \mathbf{H}_i^{(m)}, \quad m \leq n \leq N, \quad (21)$$

$$\sum_i w_i \mathbf{H}_{i,p}^{(n)} \mathbf{M}_{i,q}^{(n+2k)} = \Delta_{pq}^{(n)} \otimes \Delta_q^{(k)}, \quad k \geq 1. \quad (22)$$

Inner products between $\mathbf{H}_i^{(n)}$ and $\mathbf{M}_i^{(n+2k+1)}$, $k \geq 0$, vanish. Note that (22) is valid only when $(n+k) \leq N$.

3. Lattice-Boltzmann Method

The principal variable in Boltzmann model equations is the mass distribution function $f(\mathbf{r}, t, \mathbf{c})$: the arguments \mathbf{r} , t , and \mathbf{c} refer to the spatial, temporal, and microscopic velocity space, respectively. Lattice-Boltzmann methods can be directly derived from Boltzmann model equations. The first step in the derivation is to discretize the microscopic velocity space $\mathbf{c} = c_T \mathbf{v}$, where \mathbf{v} is a dimensionless velocity, $c_T = \sqrt{k_b T_0 / m}$ the thermal reference velocity, k_b the Boltzmann constant, m the molecular mass of the fluid, and T_0 a reference temperature. The relevance of Hermite polynomial tensors and monomials to lattice-Boltzmann method is related to the discretization of the velocity space.

Particularly, in conventional lattice-Boltzmann schemes, the discrete velocity vectors always connect two sites of a uniform lattice; that is, $(\mathbf{r} + \delta_t \mathbf{c}_i)$ is a lattice site whenever \mathbf{r} is. By h_r and δ_t we denote the spatial spacing of a uniform lattice and the discrete time step of the temporal evolution, respectively; the lattice reference velocity $c_r = h_r / \delta_t$. It is hence implied that $\mathbf{c}_i = c_r \mathbf{c}_i^* = c_T \mathbf{v}_i$, where \mathbf{c}_i^* denote appropriate dimensionless discrete velocities. The relation $c_r / c_T = a_s$ is called the scaling factor. The triplet \mathbf{c}_i , w_i , and a_s is called a lattice-Boltzmann stencil.

A way to construct lattice-Boltzmann stencils is to first prescribe \mathbf{c}_i^* and to assign $\mathbf{v}_i = a_s \mathbf{c}_i^*$. Moreover, the discrete velocity set is prescribed so that a vector from the set always has an opposite counterpart; that is, $\mathbf{c}_{-i} = -\mathbf{c}_i$ where index $-i$ refers to the opposite vector—the zero or rest vectors are an obvious exception. The unknowns a_s and w_i are then defined by solving either of the two quadrature problems, (17) or (18), up to a given order N . This is the method of prescribed abscissas [5]. The underlying motivation is to ensure correspondence between continuous and discrete moments of the equilibrium distribution functions. For example, the well-known stencils D2Q9 and D2V37 are of second and fourth order ($N = 2$ and $N = 4$) [5, 18]. Specification of three lattice-Boltzmann stencils are given in Tables 1 and 2.

TABLE 1: Specification of a fifth-order lattice-Boltzmann stencil: $a_s = 1.148732248838539$. The second column indicates the number of velocity vectors, obtained by permutating the vector components, sharing the given weight.

\mathbf{c}_i^*	p	w_i (D2V49)
(0, 0)	1	$2.1486215695621588 \times 10^{-1}$
($\pm 1, 0$)	4	$1.0466324937773665 \times 10^{-1}$
($\pm 1, \pm 1$)	4	$5.9279902836032117 \times 10^{-2}$
($\pm 2, 0$)	4	$1.6329411241508002 \times 10^{-2}$
($\pm 2, \pm 1$)	8	$6.6560728515645176 \times 10^{-3}$
($\pm 2, \pm 2$)	4	$1.4236189699711052 \times 10^{-3}$
($\pm 3, 0$)	4	$3.8641376938724455 \times 10^{-4}$
($\pm 3, \pm 1$)	8	$4.4284732290059492 \times 10^{-4}$
($\pm 3, \pm 3$)	4	$2.9169039140666429 \times 10^{-6}$
($\pm 5, 0$)	4	$8.8685055032561952 \times 10^{-7}$
($\pm 4, \pm 4$)	4	$2.2046291628739507 \times 10^{-7}$

The two-dimensional stencils are here presented for the first time: D2V49, D2V81, and D2V141 are fifth-, sixth-, and eighth-order stencils, respectively. In Section 7 we will apply D2V141 to numerically confirm accuracy of our isotropic finite differences. Furthermore, D2V49 and D2V81 will be used in a numerical comparison of some specific finite-difference approximations.

The simplest evolution equation, that is, a lattice-Boltzmann equation (LBE), for the single-particle distribution function $f_i(\mathbf{r}, t) \equiv f(\mathbf{r}, t, \mathbf{c}_i)$ is

$$f_i(\mathbf{r} + \delta_t \mathbf{c}_i, t + \delta_t) = f_i(\mathbf{r}, t) + \delta_t \Omega_i(\mathbf{r}, t), \quad i = 0, \dots, q-1, \quad (23)$$

where Ω_i is, in general, a nonlinear collision operator. For example, $\Omega_i = -f_i^{\text{neq}}/\tau$ will specify the famous lattice-BGK equation involving only a single relaxation time τ . The nonequilibrium function $f_i^{\text{neq}} := f_i - f_i^{\text{eq}}$ is defined with respect to the equilibrium function f_i^{eq} discussed below. The equilibrium function, in turn, is defined by the conserved hydrodynamic variables, that is, by the first few moments of the distribution function in the velocity space:

$$\rho = \sum_i f_i, \quad \rho \mathbf{u} = \sum_i f_i \mathbf{c}_i, \quad \rho \epsilon = \sum_i f_i (\mathbf{c}_i - \mathbf{u})^2; \quad (24)$$

the local fluid density ρ , velocity \mathbf{u} , and internal energy ϵ are the zeroth, first, and second moment of the distribution function, respectively. The above LBE results from a first-order discretization along the characteristics: higher-order discretizations are considered, for example, in [19].

A discrete equilibrium function can be specified by expanding the famous Maxwell-Boltzmann equilibrium distribution function in Hermite polynomial tensors; the expansion is truncated to a given order. For isothermal models, the resulting discrete equilibrium function is

$$f_i^{\text{eq}}(\rho, \mathbf{u}) = \rho \left(\mathcal{H}_i^{(0)} + \mathcal{H}_{i,\alpha}^{(1)} u_\alpha + \mathcal{H}_{i,\alpha\beta}^{(2)} u_\alpha u_\beta + \mathcal{H}_{i,\alpha\beta\gamma}^{(3)} u_\alpha u_\beta u_\gamma + \dots \right), \quad (25)$$

where

$$\mathcal{H}_{i,\alpha_1 \dots \alpha_n}^{(n)} \equiv \mathbf{K}_i^{(n)} := \frac{w_i}{n! c_T^{2n}} \mathbf{H}^{(n)}(\mathbf{c}_i) \quad (26)$$

is the so-called kinetic projector mapping hydrodynamic variables into a space of kinetic description; note the relation $\mathbf{H}^{(n)}(\mathbf{c}_i) = c_T^n \mathbf{H}^{(n)}(\mathbf{v}_i)$. In the next section we will base our isotropic finite-difference stencils on *difference projectors*: close analogs to kinetic projectors. From programming point of view, there remain three properties which deserve a note.

- (1) By definition, Hermite polynomial tensors $\mathbf{H}_i^{(n)}$ are symmetric in their n subscripts.
- (2) The expressions for even order Hermite polynomial tensors involve only even order monomials. Hence, $\mathbf{H}_{-i}^{(n)} = \mathbf{H}_i^{(n)}$.
- (3) The expressions for odd order Hermite polynomial tensors involve only odd order monomials. Hence, $\mathbf{H}_{-i}^{(n)} = -\mathbf{H}_i^{(n)}$.

By acknowledging these properties, whether implementing equilibrium functions based on kinetic projectors or isotropic finite differences based on difference projectors, a significant improvement in computational performance can be achieved. The number of independent components in a symmetric tensor can be obtained directly from *Pascal's triangle*. For example, a general rank-eight tensor, in three dimensions, has $3^8 = 6561$ components whereas a symmetric tensor has only 45 independent components. Furthermore, the sum of components in general tensors up to rank eight, again in 3D, is 9841; the corresponding number for symmetric tensors is 165—a number also directly obtained from Pascal's triangle. The second and third property together reduce the number of independent tensor components by a factor of two (but only approximately, because the discrete velocity set usually includes a zero vector, associated with the so-called rest particles, and it does not have an opposite counterpart).

4. Isotropic Finite Differences of Second-Order Accuracy

We propose to approximate partial derivatives of a function g with a simple finite difference:

$$\nabla^{(n)} g(\mathbf{r}) \approx \sum_i \mathbf{D}_i^{(n,m)} g(\mathbf{r} + \delta_t \mathbf{c}_i), \quad (27)$$

where the rank n tensor $\mathbf{D}_i^{(n,m)}(\mathbf{c}_i) \equiv \mathbf{D}_i^{(n,m)}$ is the difference projector storing the coefficients of the finite-difference stencil; the superscript m denotes the order of accuracy of the finite difference. The second-order accurate difference projectors are the Hermite polynomial tensors multiplied by appropriate constants:

$$\mathbf{D}_i^{(n,2)}(\mathbf{c}_i) := \frac{w_i}{\delta_t^n c_T^{2n}} \mathbf{H}^{(n)}(\mathbf{c}_i), \quad (28)$$

where $n \leq N$ and N is the order of the lattice-Boltzmann stencil.

TABLE 2: Specification of a sixth-order (D2V81) and eighth-order (D2V141) lattice-Boltzmann stencil: $a_s = 0.970008498739395$ and $a_s = 0.8369204054303525$, respectively. The second column indicates the number of velocity vectors, obtained by permutating the vector components, sharing the given weight.

\mathbf{c}_i^*	p	w_i (D2V81)	w_i (D2V141)
(0, 0)	1	$1.5381368561993594 \times 10^{-1}$	$1.1508949125706189 \times 10^{-1}$
(± 1 , 0)	4	$9.0384577224028734 \times 10^{-2}$	$7.5595334899625166 \times 10^{-2}$
(± 1 , ± 1)	4	$6.0915016936039915 \times 10^{-2}$	$5.7734363121950370 \times 10^{-2}$
(± 2 , 0)	4	$2.4220944974511140 \times 10^{-2}$	$2.9033095241565582 \times 10^{-2}$
(± 2 , ± 1)	8	$1.3146486542343966 \times 10^{-2}$	$1.8074672747613353 \times 10^{-2}$
(± 2 , ± 2)	4	$3.9625069477690709 \times 10^{-3}$	$7.4435479269746455 \times 10^{-3}$
(± 3 , 0)	4	$1.8542634450574055 \times 10^{-3}$	$4.2549928887228462 \times 10^{-3}$
(± 3 , ± 1)	8	$1.6035095499280431 \times 10^{-3}$	$3.7759145451527918 \times 10^{-3}$
(± 3 , ± 2)	8	$2.2249142091156890 \times 10^{-4}$	$9.5510661516062627 \times 10^{-4}$
(± 4 , 0)	4	$1.0005397521769796 \times 10^{-4}$	$4.9926973701248283 \times 10^{-4}$
(± 4 , ± 1)	8	$3.4398327658312326 \times 10^{-5}$	$2.1766901272228135 \times 10^{-4}$
(± 3 , ± 3)	4	$5.4494804327695976 \times 10^{-5}$	$2.7427360033189315 \times 10^{-4}$
(± 4 , ± 2)	8	$1.8837067344235765 \times 10^{-5}$	$1.3852638332696371 \times 10^{-4}$
(± 5 , 0)	4	$2.4044498210968849 \times 10^{-6}$	$1.2905073342509215 \times 10^{-5}$
(± 4 , ± 3)	8	—	$5.9323648903820310 \times 10^{-6}$
(± 5 , ± 1)	8	—	$1.6161185887309810 \times 10^{-5}$
(± 5 , ± 2)	8	$4.3501093550344725 \times 10^{-7}$	$2.4159493337948245 \times 10^{-6}$
(± 4 , ± 4)	4	—	$3.3101853527875648 \times 10^{-6}$
(± 5 , ± 3)	8	—	$1.3086410701892049 \times 10^{-6}$
(± 6 , 0)	4	—	$1.1118416570950374 \times 10^{-7}$
(± 6 , ± 1)	8	—	$4.6121305137932601 \times 10^{-7}$
(± 6 , ± 3)	8	—	$3.6530518727364592 \times 10^{-8}$
(± 7 , 0)	4	—	$8.3273853753395782 \times 10^{-9}$
(± 7 , ± 2)	8	—	$2.3109247814347261 \times 10^{-9}$

In order to prove the second-order accuracy of $\mathbf{D}_i^{(n,2)}$, we first Taylor-expand function g at the right-hand side of (27), use the definition in (28), and change the order of summation:

$$\begin{aligned} & \sum_i \mathbf{D}_{i,\mathbf{p}}^{(n,2)} g(\mathbf{r} + \delta_i \mathbf{c}_i) \\ &= \frac{1}{\delta_t^n c_T^{2n}} \left(\sum_{m=0}^{\infty} \frac{\delta_t^m}{m!} \nabla_{\mathbf{q}}^{(m)} \sum_i w_i \mathbf{H}_{\mathbf{p}}^{(n)}(\mathbf{c}_i) \mathbf{M}_{\mathbf{q}}^{(m)}(\mathbf{c}_i) \right) g(\mathbf{r}). \end{aligned} \quad (29)$$

Note that here the repeated index set \mathbf{q} emphasizes a full tensor contraction between $\mathbf{M}_{i,\mathbf{q}}^{(m)}$ and $\nabla_{\mathbf{q}}^{(m)}$. By using the inner products presented in Section 2.3 and by acknowledging the simple relations $\mathbf{H}^{(n)}(\mathbf{c}_i) = c_T^n \mathbf{H}^{(n)}(\mathbf{v}_i)$ and $\mathbf{M}^{(m)}(\mathbf{c}_i) = c_T^m \mathbf{M}^{(m)}(\mathbf{v}_i)$, we can transform the right-hand side into

$$\begin{aligned} & \left(\frac{1}{n!} \Lambda_{\mathbf{p}\mathbf{q}}^{(n)} \nabla_{\mathbf{q}}^{(n)} + \sum_{k=1}^{N-n} \frac{\delta_t^{2k} c_T^{2k}}{(n+2k)!} \nabla_{\mathbf{q}}^{(n+2k)} \right. \\ & \quad \left. \times (\Lambda_{\mathbf{p}\mathbf{q}}^{(n)} \otimes \Delta_{\mathbf{q}}^{(k)}) + \dots \right) g(\mathbf{r}). \end{aligned} \quad (30)$$

Remember that the operator \otimes has precedence over all other mathematical operations and manipulations. Hence, the

correct interpretation above is to first expand the summation over combinations and then enforce the tensor contractions for each term separately.

Since $\nabla_{\mathbf{q}}^{(n+2k)}$ is clearly a symmetric tensor, we can make use of (9) and obtain $\Lambda_{\mathbf{p}\mathbf{q}}^{(n)} \nabla_{\mathbf{q}}^{(n+2k)} = n! \nabla_{\mathbf{p}}^{(n)} \nabla_{\mathbf{q}}^{(2k)}$, $k \geq 0$; that is, the orthogonality tensor changes n indices from set \mathbf{q} to set \mathbf{p} . Furthermore, the contraction $\nabla_{\mathbf{q}}^{(2k)} \Delta_{\mathbf{q}}^{(k)}$ results in $(\partial^2)^k$, where $\partial^2 = \partial_{\alpha} \partial_{\alpha}$ denotes Laplacian, $(\partial^2)^2$ denotes Laplacian of Laplacian, and so on. In fact, because $\Delta_{\mathbf{q}}^{(k)}$ involves $(2k)!/2^k k!$ terms, the Laplacians will be repeated an equal number of times. Finally, the summation over combinations includes $(n+2k)!/n!(2k)!$ terms. Hence

$$\begin{aligned} & \sum_i \mathbf{D}_{i,\mathbf{p}}^{(n,2)} g(\mathbf{r} + \delta_i \mathbf{c}_i) \\ &= \nabla_{\mathbf{p}}^{(n)} \left(1 + \sum_{k=1}^{N-n} \frac{h_r^{2k}}{a_s^{2k} 2^k k!} (\partial^2)^k + \dots \right) g(\mathbf{r}). \end{aligned} \quad (31)$$

This is our master equation. It confirms that the difference projectors $\mathbf{D}_i^{(n,2)}$ defined in (28), together with the finite difference specified in (27), result in second-order approximations for the partial derivatives $\nabla^{(n)}$. Also, (31) exposes the isotropy: with $\mathbf{D}_i^{(n,2)}$, the first $(N-n)$ leading order error terms are isotropic (Laplacian is an isotropic operator).

5. High-Order Accurate Isotropic Finite Differences

After establishing the second-order accurate finite-difference stencils, we proceed to define higher-order accurate stencils. In order to do so, we explicitly write the first few leading-order error terms of the second-order approximation:

$$\begin{aligned} \sum_i \mathbf{D}_i^{(n,2)} g(\mathbf{r} + \delta_i \mathbf{c}_i) \\ = \nabla^{(n)} \left(1 + \frac{h_r^2}{2a_s^2} \partial^2 + \frac{h_r^4}{8a_s^4} (\partial^2)^2 + \frac{h_r^6}{48a_s^6} (\partial^2)^3 + \dots \right) g(\mathbf{r}). \end{aligned} \quad (32)$$

With the above equation, it is straightforward to write higher-order difference projectors. First, the fourth-order accurate finite-difference stencils are defined with

$$\begin{aligned} D_{i,\alpha}^{(1,4)} &= D_{i,\alpha}^{(1,2)} - \frac{h_r^2}{2a_s^2} D_{i,\alpha\beta\beta}^{(3,2)}, \\ D_{i,\alpha\beta}^{(2,4)} &= D_{i,\alpha\beta}^{(2,2)} - \frac{h_r^2}{2a_s^2} D_{i,\alpha\beta\gamma\gamma}^{(4,2)}, \\ D_{i,\alpha\beta\gamma}^{(3,4)} &= D_{i,\alpha\beta\gamma}^{(3,2)} - \frac{h_r^2}{2a_s^2} D_{i,\alpha\beta\gamma\delta\delta}^{(5,2)}, \\ D_{i,\alpha\beta\gamma\delta}^{(4,4)} &= D_{i,\alpha\beta\gamma\delta}^{(4,2)} - \frac{h_r^2}{2a_s^2} D_{i,\alpha\beta\gamma\delta\epsilon\epsilon}^{(6,2)}, \\ D_{i,\alpha\beta\gamma\delta\epsilon}^{(5,4)} &= D_{i,\alpha\beta\gamma\delta\epsilon}^{(5,2)} - \frac{h_r^2}{2a_s^2} D_{i,\alpha\beta\gamma\delta\epsilon\zeta\zeta}^{(7,2)}, \\ D_{i,\alpha\beta\gamma\delta\epsilon\zeta}^{(6,4)} &= D_{i,\alpha\beta\gamma\delta\epsilon\zeta}^{(6,2)} - \frac{h_r^2}{2a_s^2} D_{i,\alpha\beta\gamma\delta\epsilon\zeta\eta\eta}^{(8,2)}. \end{aligned} \quad (33)$$

Now by using the above fourth-order stencils, we can write the sixth-order stencils:

$$\begin{aligned} D_{i,\alpha}^{(1,6)} &= D_{i,\alpha}^{(1,2)} - \frac{h_r^2}{2a_s^2} D_{i,\alpha\beta\beta}^{(3,4)} - \frac{h_r^4}{8a_s^4} D_{i,\alpha\beta\beta\gamma\gamma}^{(5,2)}, \\ D_{i,\alpha\beta}^{(2,6)} &= D_{i,\alpha\beta}^{(2,2)} - \frac{h_r^2}{2a_s^2} D_{i,\alpha\beta\gamma\gamma}^{(4,4)} - \frac{h_r^4}{8a_s^4} D_{i,\alpha\beta\gamma\gamma\delta\delta}^{(6,2)}, \\ D_{i,\alpha\beta\gamma}^{(3,6)} &= D_{i,\alpha\beta\gamma}^{(3,2)} - \frac{h_r^2}{2a_s^2} D_{i,\alpha\beta\gamma\delta\delta}^{(5,4)} - \frac{h_r^4}{8a_s^4} D_{i,\alpha\beta\gamma\delta\delta\epsilon\epsilon}^{(7,2)}, \\ D_{i,\alpha\beta\gamma\delta}^{(4,6)} &= D_{i,\alpha\beta\gamma\delta}^{(4,2)} - \frac{h_r^2}{2a_s^2} D_{i,\alpha\beta\gamma\delta\epsilon\epsilon}^{(6,4)} - \frac{h_r^4}{8a_s^4} D_{i,\alpha\beta\gamma\delta\epsilon\epsilon\zeta\zeta}^{(8,2)}. \end{aligned} \quad (34)$$

In a similar way, we use the sixth-order stencils in the definitions of eighth-order stencils:

$$\begin{aligned} D_{i,\alpha}^{(1,8)} &= D_{i,\alpha}^{(1,2)} - \frac{h_r^2}{2a_s^2} D_{i,\alpha\beta\beta}^{(3,6)} - \frac{h_r^4}{8a_s^4} D_{i,\alpha\beta\beta\gamma\gamma}^{(5,4)} \\ &\quad - \frac{h_r^6}{48a_s^6} D_{i,\alpha\beta\beta\gamma\gamma\delta\delta}^{(7,2)}, \\ D_{i,\alpha\beta}^{(2,8)} &= D_{i,\alpha\beta}^{(2,2)} - \frac{h_r^2}{2a_s^2} D_{i,\alpha\beta\gamma\gamma}^{(4,6)} - \frac{h_r^4}{8a_s^4} D_{i,\alpha\beta\gamma\gamma\delta\delta}^{(6,4)} \\ &\quad - \frac{h_r^6}{48a_s^6} D_{i,\alpha\beta\gamma\gamma\delta\delta\epsilon\epsilon}^{(8,2)}. \end{aligned} \quad (35)$$

Above we have defined the high-accuracy difference projectors in an hierarchical way. In practice, however, more straightforward definition might be more convenient for a computer implementation. That is, we can simply write out the expressions and get, for example,

$$\begin{aligned} D_{i,\alpha}^{(1,6)} &= D_{i,\alpha}^{(1,2)} - \frac{h_r^2}{2a_s^2} D_{i,\alpha\beta\beta}^{(3,2)} + \frac{h_r^4}{8a_s^4} D_{i,\alpha\beta\beta\gamma\gamma}^{(5,2)}, \\ D_{i,\alpha}^{(1,8)} &= D_{i,\alpha}^{(1,2)} - \frac{h_r^2}{2a_s^2} D_{i,\alpha\beta\beta}^{(3,2)} + \frac{h_r^4}{8a_s^4} D_{i,\alpha\beta\beta\gamma\gamma}^{(5,2)} - \frac{h_r^6}{48a_s^6} D_{i,\alpha\beta\beta\gamma\gamma\delta\delta}^{(7,2)}. \end{aligned} \quad (36)$$

In this way, the high-accuracy projectors are defined using only second-order accurate projectors.

The degree of isotropy in the case of high-accuracy approximation of high-rank derivatives is discussed next. First we remind the reader that n denotes the rank of the derivative, N is the order of the lattice-Boltzmann stencil utilized, and m refers to the order of accuracy of the approximation. Then, the number of leading-order error terms which are isotropic, or the degree of isotropy, is

$$N_I = (N - n) - (m - 2). \quad (37)$$

6. Application of the Isotropic Finite Differences

In order to elucidate the usage of the above proposed finite-differences, we will provide the coefficients of some simple stencils in their most explicit form. Let us consider a particular application where the gradient and the gradient of Laplacian of an arbitrary function g must be approximated with fourth- and second-order accuracy, respectively. Furthermore, the approximations must be isotropic meaning that at least the leading-order error term is not directionally biased. In the case of the gradient, $n = 1$ and $m = 4$. Hence, from (37), $N \geq 4$ in order to have $N_I \geq 1$. The gradient of Laplacian case, $n = 3$ and $m = 2$, leads to the same conclusion. That is, at least a fourth-order lattice-Boltzmann stencil must be utilized: the well-known D2V37 stencil is a candidate [5].

Directly from (27) and (28), the second-order approximations of gradient and gradient of Laplacian, in their most explicit form, are

$$\partial_\alpha g(\mathbf{r}) = \frac{a_s^2}{h_r} \sum_i w_i c_{i,\alpha}^* g(\mathbf{r} + h_r \mathbf{c}_i^*) + \mathcal{O}(h_r^2), \quad (38)$$

$$\partial_\alpha \partial^2 g(\mathbf{r}) = \frac{a_s^6}{h_r^3} \sum_i w_i c_{i,\alpha}^* \left[c_i^2 - \frac{B_1}{a_s^2} \right] g(\mathbf{r} + h_r \mathbf{c}_i^*) + \mathcal{O}(h_r^2), \quad (39)$$

where $c_i^2 \equiv c_{i,\beta}^* c_{i,\beta}^*$, the auxiliary coefficient $B_1 := (D + 2)$, and $D \equiv \delta_{\beta\beta}$. The approximation of the gradient, (38), is well known in the lattice-Boltzmann context and the approximation of the gradient of Laplacian, (39), is exactly the same as proposed in Appendix B of [13]. Furthermore, using the appropriate expression from Section 5, the explicit form of the fourth-order approximation of the gradient is

$$\partial_\alpha g(\mathbf{r}) = \frac{a_s^2}{2h_r} \sum_i w_i c_{i,\alpha}^* [B_2 - a_s^2 c_i^2] g(\mathbf{r} + h_r \mathbf{c}_i^*) + \mathcal{O}(h_r^4), \quad (40)$$

where $B_2 := (D + 4)$. Again, the corresponding expression given by Philippi et al. agrees with this approximation (Equation (B1) in [13] involves a mistake: (40) is the correct expression).

Our approximations are derived from the theoretical basis developed in the previous sections. In [13], Philippi et al. use a three-step procedure for constructing their stencils: (1) the general function $g(\mathbf{r} + \delta_i \mathbf{c}_i)$ is Taylor-expanded, (2) discrete moments (in the velocity space) of the Taylor expansion are computed, where the moments involve the discrete weights, and (3) linear combinations of these moments, and $g(\mathbf{r})$, are computed so as to deliver appropriate approximations. That is, the three-step procedure allows tailored approximations which, in some cases, can be, for example, more compact than the approximations derived here. Thampi et al. proposed the same procedure [14], independently of Philippi et al., and applied it for a second-order accurate isotropic approximation of Laplacian.

As an example, let us consider second- and fourth-order accurate approximations of Laplacian. From (37), $N \geq 3$ and $N \geq 5$ in order to have $N_f \geq 1$ for the two approximations, respectively. Hence, for example, the D2V17 and D2V49 stencils could be utilized. Using only (27) and (28), the explicit expressions for the second-order approximations of Laplacian and BiLaplacian ($\partial^2 \partial^2$) are

$$\partial^2 g(\mathbf{r}) = \frac{a_s^4}{h_r^2} \sum_i w_i \left[c_i^2 - \frac{D}{a_s^2} \right] g(\mathbf{r} + h_r \mathbf{c}_i^*) + \mathcal{O}(h_r^2), \quad (41)$$

$$\begin{aligned} \partial^2 \partial^2 g(\mathbf{r}) = & \frac{a_s^8}{h_r^4} \sum_i w_i \left[c_i^2 \left(c_i^2 - \frac{2}{a_s^2} B_1 \right) + \frac{D}{a_s^4} B_1 \right] \\ & \times g(\mathbf{r} + h_r \mathbf{c}_i^*) + \mathcal{O}(h_r^2). \end{aligned} \quad (42)$$

Then, with the appropriate expression from Section 5, the explicit form for the fourth-order approximation of the Laplacian is

$$\begin{aligned} \partial^2 g(\mathbf{r}) = & \frac{a_s^4}{2h_r^2} \sum_i w_i \left[c_i^2 (2B_3 - a_s^2 c_i^2) - \frac{D}{a_s^2} B_2 \right] \\ & \times g(\mathbf{r} + h_r \mathbf{c}_i^*) + \mathcal{O}(h_r^4), \end{aligned} \quad (43)$$

where $B_3 := D + 3$.

The corresponding second- and fourth-order approximations of Laplacian obtained with the three-step procedure, as proposed by Philippi et al., are

$$\partial^2 g(\mathbf{r}) = \frac{2a_s^2}{h_r^2} \left[\sum_i w_i g(\mathbf{r} + h_r \mathbf{c}_i^*) - g(\mathbf{r}) \right] + \mathcal{O}(h_r^2), \quad (44)$$

$$\begin{aligned} \partial^2 g(\mathbf{r}) = & \frac{a_s^2}{h_r^2} \left[\sum_i w_i (B_2 - a_s^2 c_i^2) g(\mathbf{r} + h_r \mathbf{c}_i^*) - 4g(\mathbf{r}) \right] \\ & + \mathcal{O}(h_r^4); \end{aligned} \quad (45)$$

the same second-order approximation of Laplacian was also proposed by Thampi et al. [14]. In fact, (44) is equivalent to the approximation proposed earlier by Lee and Lin [11]. In addition, when (44) is utilized together with D2Q9, it coincides with the *Mehrstellen* approximation (see, e.g., [1, 2]).

Clearly, in the case of Laplacians, our approximations are not equivalent to those obtained using the three-step procedure. Moreover, the approximations given in (44) and (45) are isotropic whenever $N \geq 2$ and $N \geq 4$, respectively. Hence, for example, (45) allows more compact approximations than (43): the span (or the spatial extent in an axis direction) of the fifth-order stencil D2V49 presented in Table 1 is 5 lattice spacings, whereas the span of D2V37 is only 3.

Above the explicit forms of our approximations, still valid in any dimension, are given only as examples. In the implementations, however, it is not necessary to use the explicit forms in order to reach high computational efficiency; we consider it more convenient to use the abstract definitions given in the previous sections. First, even the high-rank Hermite polynomial tensors can be easily implemented, in an hierarchical manner, using the recurrence relation of (15). For example, preprocessor directives (macros), if available in the programming language, can be used for implementing the tensors. The finite-difference coefficients, that is, components of an appropriate difference projector, can then be stored by utilizing the appropriate Hermite polynomial tensors. Due to their definitions, the difference projectors share important properties with the Hermite polynomial tensors. Namely, the difference projectors are fully symmetric in their subscripts. Also,

$$\mathbf{D}_{-i}^{(n,m)} = \mathbf{D}_i^{(n,m)}, \quad \mathbf{D}_{-i}^{(n,m)} = -\mathbf{D}_i^{(n,m)} \quad (46)$$

for even and odd n , respectively. Hence, a relatively small number of finite-difference coefficients need to be stored even when approximating high-rank derivatives.

7. Numerical Experiments and Discussion

In order to verify the accuracy of the presented finite differences, we numerically compute the derivative approximations and compare them against analytical solutions. Our analytical reference function is

$$\begin{aligned} s(x, y, \omega) &= \sin(\omega \pi (2x + y)), \\ (x, y) &\in [0, 1] \times [0, 2], \quad \omega \in [1, 2[, \end{aligned} \quad (47)$$

for which derivatives, even high rank, are easily available. The additional constant parameter ω is introduced as an extra degree of freedom for the numerical experiments. In order to construct a square lattice (equal lattice spacing in the x - and y -directions), we always define the number of lattice nodes in the y -direction to be two times the number of nodes in the x -direction. Numerical error of the approximations is measured with a relative L_2 -error norm:

$$L_2 = \sqrt{\frac{\sum (n(x, y) - a(x, y))^2}{\sum a(x, y)^2}}, \quad (48)$$

where the summation is over all lattice nodes, $n(x, y)$ denotes a numerically computed value, and $a(x, y)$ is a corresponding analytical value.

First we verify the theoretical convergence rates of the approximation errors. In such a verification, the truncation errors must dominate over the round-off errors arising from the floating-point arithmetic. Hence, we use relatively small lattices: in the reported results for the convergence rates, the number of nodes in the x -direction is between 15 and 60. Furthermore, we set $\omega = 1$ and apply the eighth-order lattice-Boltzmann stencil D2V141, presented in Section 3, for the specification of the difference projectors. The difference projectors are defined in Sections 4 and 5. Figure 1 reports the relative L_2 -error norms for some numerical approximations: for each rank, an approximation with a difference projector of highest attainable order of accuracy is chosen. Otherwise, the approximations for the figure are randomly chosen. Figure 1 confirms the theoretical results for the chosen approximations; we have checked all the approximations and the theoretical results are indeed verified. In the approximations of high-rank derivatives, round-off errors start to dominate with larger lattice spacings than in the approximations of lower-rank derivatives—as expected.

Next we measure the computational efficiency of the approximations proposed here. For reference, we also measure the efficiency of some previously published approximations. In the approach of Lee et al. [11, 12], derivatives in 2D and 3D are approximated by taking moments of the conventional 1D finite differences along the characteristic lines. As an example, in order to have a minimal isotropic fourth-order approximation of the first derivative—only the leading-order error term is isotropic—a standard fourth-order central difference together with a third-order lattice-Boltzmann stencil can be used. It is also possible to use the third-order isotropic stencil $E^{(6)}$ presented in [8, 10]: $E^{(6)}$ has 12 nodes (without the center node) and a span of 2 lattice

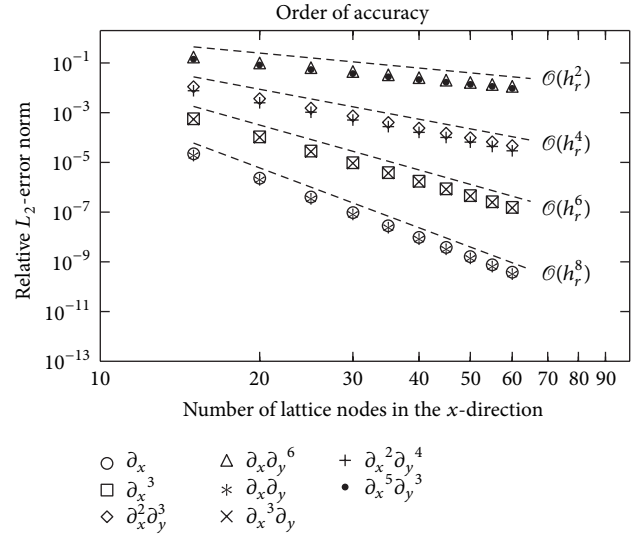


FIGURE 1: Order of convergence of the numerical error for some derivative approximations. The lattice-Boltzmann stencil D2V141 has been utilized. The dashed lines indicate 2nd-, 4th-, 6th-, and 8th-order convergence.

spacings. The fourth-order central difference together with the D2V17 and $E^{(6)}$ stencils will result in effective stencils of 32 and 24 nodes, respectively; the effective spans are 6 and 4 lattice spacings. In comparison, utilization of (40) requires at least a fourth-order lattice-Boltzmann stencil, for example, D2V37. Note also that it remains to be investigated whether the Lee et al. approach can be extended for the approximation of cross derivatives.

The approximations proposed by Lee et al. [11, 12], Thampai et al. [14], and by Philippi et al. [13] are equal to the approximations presented here in one important sense: expressions for the finite differences are valid regardless of dimension, and any lattice Boltzmann stencil isotropic enough can be utilized; a family of finite differences are defined with single expressions. In contrast, Patra and Karttunen presented more specified isotropic stencils for the numerical computation of second, third, and fourth derivatives [2]. In their approach, the labor of explicitly specifying the coefficients for each case is rewarded with highly compact stencils (a narrow span and a small number of nodes) (sections G and H of [2] appear to involve a mistake: we utilized the coefficients after inverting the signs. Probably the same applies also for section I).

In order to compare the computational efficiency of the aforementioned approaches, we approximate some low- and moderate-rank derivatives with second- and fourth-order accuracy. Our analytical reference function is again given by (47) and a lattice of 600×1200 nodes is used. The reference function values are precomputed for the lattice nodes and stored in the computer memory. For all finite-difference stencils, the computational times are measured after repeating the approximations 50 times; that is, the lattice is iterated 50 times and during each iteration the derivatives are approximated once at every node. With this repetition we aim for robust computational times. The additional

TABLE 3: Comparison of various approximations for low- and moderate-rank derivatives with second- and fourth-order accuracy. The computational times are given in milliseconds and the *Empty load* refers to the Mehrstellen case where only the actual approximation is skipped: the relative computational times are defined with respect to the empty load. References for the below compared approximations are as follows: Lee and Lin [11], Lee and Fischer [12], Patra and Karttunen [2], and Philippi et al. [13].

	Nodes	Span	C. time	Rel. time	Rel. L_2 -error
Empty load	—	—	7984	0.0	—
∂^2 (2nd-order approximation)					
Mehrstellen	9	1	8435	0.0565	4.124×10^{-5}
Equation (41), D2V17	17	3	8813	0.1038	9.160×10^{-5}
∂_x (4th-order approximation)					
Lee, E ⁽⁶⁾	24	4	8978	0.1245	1.632×10^{-8}
Lee, D2V17	32	6	9341	0.1699	1.678×10^{-8}
Equation (40), D2V37	36	3	9547	0.1958	1.491×10^{-8}
$\partial_x \partial^2$ (2nd-order approximation)					
Patra, $c_1 = 1/24$	12	2	8641	0.0823	1.237×10^{-4}
Patra, $c_1 = 0$	16	2	8844	0.1077	1.237×10^{-4}
Equation (39), D2V37	36	3	9688	0.2134	1.727×10^{-4}
∂^2 (4th-order approximation)					
Patra, $c_1 = -1/120$	17	2	8803	0.1026	2.721×10^{-9}
Patra, $c_1 = 0$	21	2	8900	0.1147	2.721×10^{-9}
Philippi, D2V37	37	3	9503	0.1903	4.969×10^{-9}
Equation (43), D2V49	49	5	10109	0.2662	1.758×10^{-8}
$\partial^2 \partial^2$ (2nd-order approximation)					
Patra, $c_1 = 1/12$	17	2	8781	0.0998	8.247×10^{-5}
Patra, $c_1 = 0$	21	2	8891	0.1136	8.248×10^{-5}
Equation (42), D2V49	49	5	10294	0.2893	1.875×10^{-4}
$\partial_x \partial^2$ (4th-order approximation)					
Patra, $c_1 = -17/2160$	26	3	9006	0.1281	1.428×10^{-8}
Patra, $c_1 = 0$	30	3	9435	0.1817	1.428×10^{-8}
Section 5, D2V81	80	5	11253	0.4095	3.456×10^{-8}

parameter ω now evolves with the iterations: the initial value 1 is incremented by 1/50 after each iteration. The purpose of evolving ω is to prevent aggressive compiler optimizations which may compromise the comparisons between different stencils. The reference function values are updated at a node after the approximation of the local derivative.

The computational times are measured in a computer equipped with *AMD Athlon 64 X2 Dual Core Processor 5000+* (2.6 GHz and 2 GB RAM); the *GNU compiler g++* (version 4.4.1) is used with the option -O3. The computational times reported in Table 3 are average values of 5 executions and given in milliseconds. The *Empty load* refers to the Mehrstellen case where only the actual approximation of the derivative is skipped: the relative computational times are defined with respect to the empty load. The relative times are presented also in Figure 2 for visual inspection. With the Patra and Karttunen stencils, we use $c_2 = c_5 = 0$ in the fourth-order accurate approximations of $\partial_x \partial^2$. The leading-order error terms related to the isotropic stencils of Patra and Karttunen are (theoretically) independent of the degrees of freedom present. Note that, in the case of odd-rank derivatives, the number of nodes in Table 3 does not agree with the lattice-Boltzmann stencil utilized. This is due

to our choice of omitting the zero-velocity vector from the finite-difference stencils on those particular approximations in an attempt to have a fair efficiency comparison between the schemes.

Our first observation is that even the heaviest approximation, utilizing the D2V81 stencil, requires only about 41% more computing time than that of the empty load. In many applications the approximations of derivatives are only a part of the computation: the relative computation times measured suggest that the isotropic derivative approximations do not necessarily introduce a large overhead into the total computing time. Be that as it may, the relative computing times measured appear to be in-line with those reported in [2].

On the other hand, the comparison of relative times between various approximations reveals large differences. In fact, the relative computing times seem to depend linearly on the number of nodes in the finite-difference stencil. The right part of Figure 2 presents a linear fit to the computing times. Furthermore, the stencil with the smallest span among the candidates consistently produces the most accurate approximation. In summary, our observations indicate that compact finite-difference stencils with a small number of nodes are to be preferred. Hence, from strictly a computational efficiency

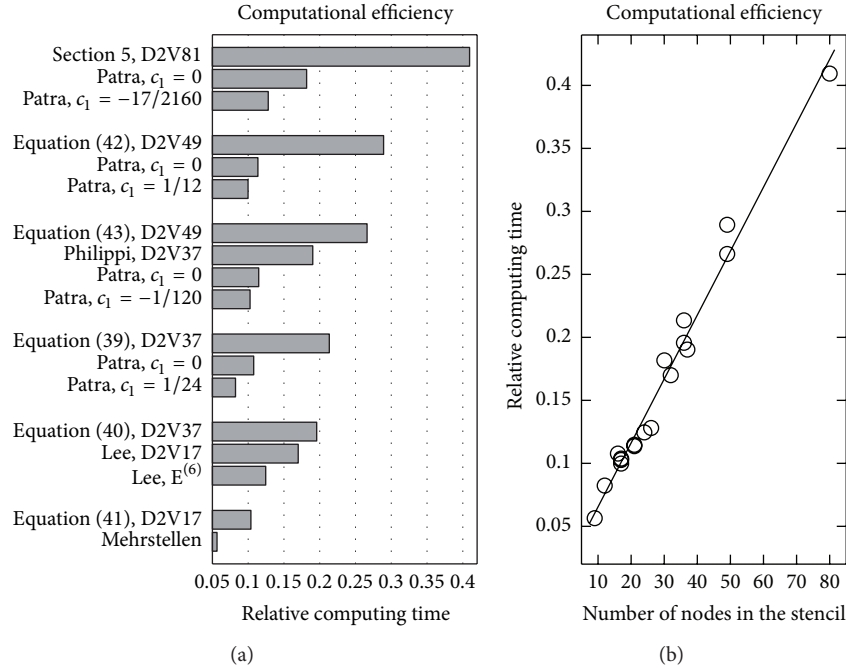


FIGURE 2: The relative computational times of Table 3 in a graphical format. On the right, the relative times are plotted as functions of the number of nodes in the finite-difference stencils; the line indicates a linear fit. References for the above compared approximations are as follows: Lee et al. [11, 12], Patra and Karttunen [2], and Philippi et al. [13].

point of view, the approximations proposed in this work are suboptimal and this can be attributed to their generality. At the same time, the generality is also the major asset: it allows a straightforward construction of high-accuracy, isotropic approximations of high-rank derivatives. An interesting compromise between efficiency and generality might be feasible by combining the Philippi et al. approach with the isotropic (nonlattice-Boltzmann) stencils presented in [8–10]—this possibility warrants further investigation.

In our last numerical experiments, we validate the theoretically derived isotropy properties of the finite-difference stencils. This is done by approximating solutions of the diffusion equation $\partial_t C = \mathcal{D} \partial^2 C$ in a two-dimensional case: the local concentration C is the dynamic variable and \mathcal{D} the constant diffusion coefficient. Here we will use $\mathcal{D} = 0.05$. For an unbounded domain and when the initial condition is given by a point impulse P_c , the analytical solution for the concentration is

$$C(r, t) = \frac{P_c}{4\pi\mathcal{D}t} e^{-r^2/4\mathcal{D}t}, \quad (49)$$

where r is the distance from the point impulse. To compute numerical solutions, we approximate the Laplacian with a finite-difference stencil and then treat the resulting ordinary differential equation using the standard second-order *Runge-Kutta* scheme with the *midpoint rule*. We execute the computations on a 141×141 grid and the grid spacing is set to $h_r = 0.01$; the point impulse is located at the center of the domain. The numerical computations are initialized with the analytical field evaluated at time $t_0 = 0.01$.

The numerical solutions are advanced until $t = 0.1$ with a constant time step $\delta_t = 10^{-5}$ after which the local relative errors are measured with respect to the analytical solution. Note that the purpose is to investigate the isotropy properties of the finite-difference stencils. Hence, due to the relatively low-order numerical time-integration scheme, the time step has to be small enough in order to allow spatial discretization errors to dominate over the temporal discretization errors. This is especially true for the high-order accurate finite-difference approximations. Figure 3 presents the local relative errors for four different second-order accurate finite-difference approximations of the Laplacian. For visualization purposes, the local errors are presented only for the domain $r \leq 0.3$. The errors for two of them, (a) the standard five-point stencil and (b) (41) with D2Q9, are anisotropic while for (c) the Mehrstellen approximation and (d) (41) with D2V17 the errors are isotropic. The local relative errors hence confirm the theoretically predicted isotropy properties.

Similarly, Figure 4 presents the local errors for fourth-order accurate finite-difference approximations. For (a) the standard nine-point stencil, (c) (45) with D2V37, and (d) Patra and Karttunen ($c_1 = 0$) the errors conform with the theoretical predictions. The error for (b) (43) with D2V37 is isotropic: this is not in accordance with the prediction from (37). In general, according to (37), the proposed fourth-order approximation of Laplacian with a fourth-order LB stencil is anisotropic. However, occasionally LB stencils fulfill particular isotropy conditions beyond their order of construction. This can perhaps explain the observed, positive anomaly but, nevertheless, further investigation is needed to explain the observation. Figure 5 presents the errors for sixth-

Second-order approximation of the Laplacian

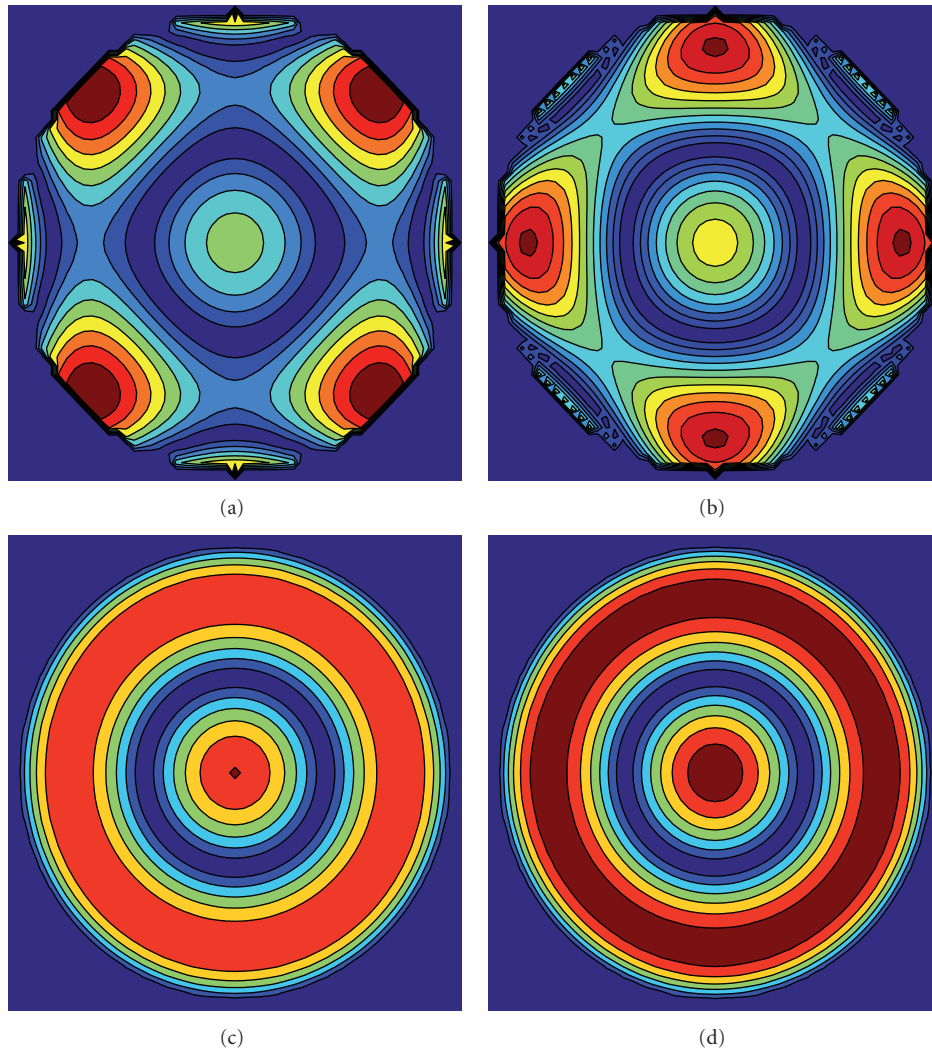


FIGURE 3: Isocontours of the local relative errors when numerical solutions of the diffusion equation are computed using second-order accurate finite-difference approximations for the Laplacian. The errors are anisotropic for two of the approximations, (a) the standard five-point stencil and (b) (41) with D2Q9, while for the remaining two, (c) the Mehrstellen approximation and (d) (41) with D2V17, the errors are isotropic. The numerical results conform with the theoretical predictions. The colors are not scaled between the approximations.

and eighth-order approximations and, again, the numerical results conform with the theory. In summary, the numerical results verify the theoretically derived isotropy properties, particularly (37), with the above-discussed exception.

8. Conclusions

We have presented, and numerically verified, finite differences based on application of lattice-Boltzmann stencils. In particular, high-order accurate isotropic approximations for high-rank derivatives are presented. The expressions for finite differences are valid in arbitrary dimensions, particularly in two and three dimensions, and any lattice Boltzmann stencil isotropic enough can be utilized.

The isotropy and accuracy properties of the proposed approximations are derived directly from the theoretical basis developed in this work. For the numerical verification of the presented theory, we introduced 5th-, 6th-, and 8th-order two-dimensional lattice-Boltzmann stencils. Moreover, in the construction of the finite differences, we extended the theory of Hermite polynomial tensors in discrete spaces. First, we proved the equivalency between two approaches for constructing lattice-Boltzmann stencils. Secondly, we presented the expressions for discrete inner products between monomials and Hermite polynomial tensors. These inner products can be useful tools, for example, in the Chapman-Enskog analysis of lattice-Boltzmann schemes and, more generally, in any numerical analysis involving functions expanded in Hermite polynomial tensors.

Fourth-order approximation of the Laplacian

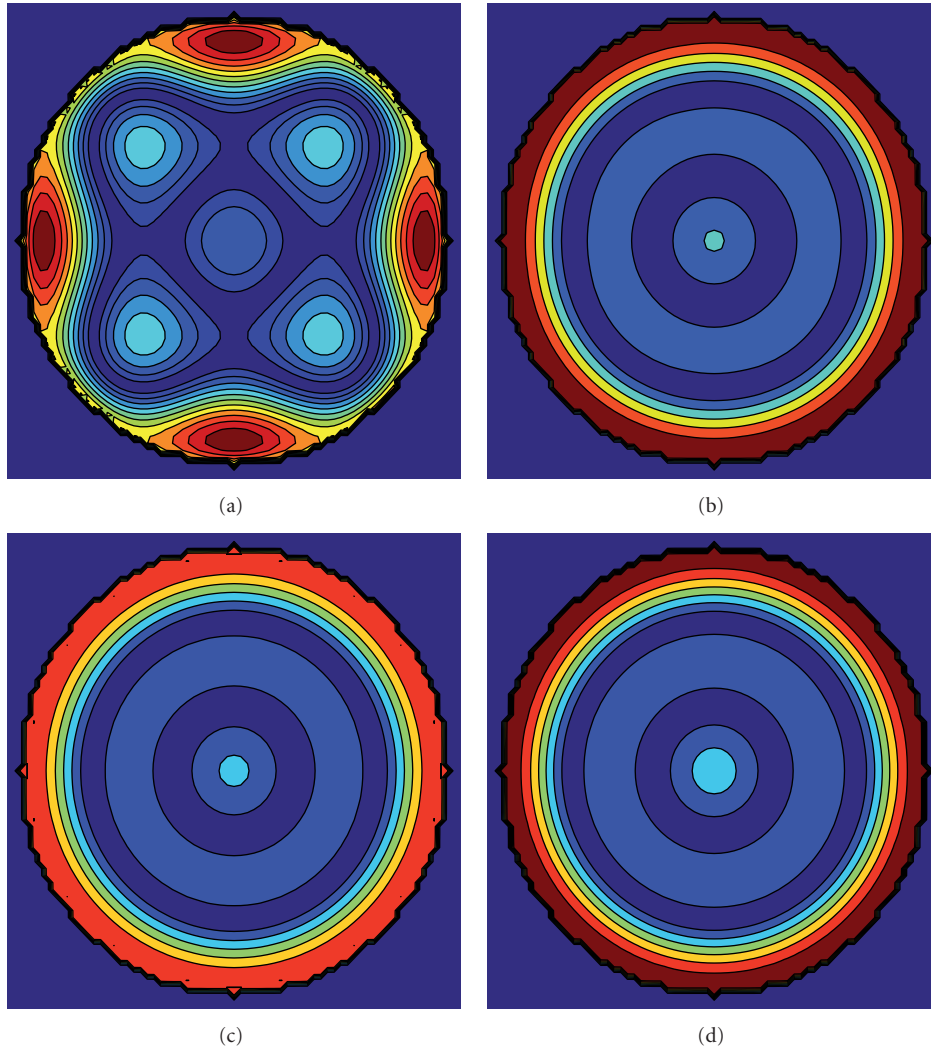


FIGURE 4: See the caption of Figure 3. The anisotropic error for (a) the standard nine-point stencil and the isotropic errors for (c) (45) with D2V37 and (d) Patra and Karttunen ($c_1 = 0$) conform with the theoretical predictions. The error for (b) (43) with D2V37 is isotropic: this is not in accordance with the prediction from (37). The colors are not scaled between the approximations.

Finally, the isotropic finite-difference approximations appear to be more stable than their anisotropic counterparts. Hence, an analytical and numerical investigation on stability properties of isotropic finite differences is a relevant research topic for the future.

Appendices

A. Hermite Polynomial Tensors and Monomials

Using the notation presented in Section 2.1, the explicit expressions for the first few Hermite polynomial tensors are $\mathbf{H}^{(0)} = 1 = \mathbf{M}^{(0)}$, $\mathbf{H}^{(1)} = \mathbf{v} = \mathbf{M}^{(1)}$, and

$$\mathbf{H}^{(2)} = \mathbf{M}^{(2)} - \Delta^{(1)},$$

$$\begin{aligned} \mathbf{H}^{(3)} &= \mathbf{M}^{(3)} - \mathbf{M}^{(1)} \otimes \Delta^{(1)}, \\ \mathbf{H}^{(4)} &= \mathbf{M}^{(4)} - \mathbf{M}^{(2)} \otimes \Delta^{(1)} + \Delta^{(2)}, \\ \mathbf{H}^{(5)} &= \mathbf{M}^{(5)} - \mathbf{M}^{(3)} \otimes \Delta^{(1)} + \mathbf{M}^{(1)} \otimes \Delta^{(2)}, \\ \mathbf{H}^{(6)} &= \mathbf{M}^{(6)} - \mathbf{M}^{(4)} \otimes \Delta^{(1)} + \mathbf{M}^{(2)} \otimes \Delta^{(2)} - \Delta^{(3)}, \\ \mathbf{H}^{(7)} &= \mathbf{M}^{(7)} - \mathbf{M}^{(5)} \otimes \Delta^{(1)} + \mathbf{M}^{(3)} \otimes \Delta^{(2)} - \mathbf{M}^{(1)} \otimes \Delta^{(3)}, \\ \mathbf{H}^{(8)} &= \mathbf{M}^{(8)} - \mathbf{M}^{(6)} \otimes \Delta^{(1)} + \mathbf{M}^{(4)} \otimes \Delta^{(2)} - \mathbf{M}^{(2)} \otimes \Delta^{(3)} + \Delta^{(4)}. \end{aligned} \quad (\text{A.1})$$

Note that, for example, in the last expression above the second, third, and fourth terms at the right-hand side involve summation of 28, 70, and 28 tensor products, respectively. Furthermore, the expressions for even and odd rank Hermite

Sixth- and eighth-order approximation of the Laplacian

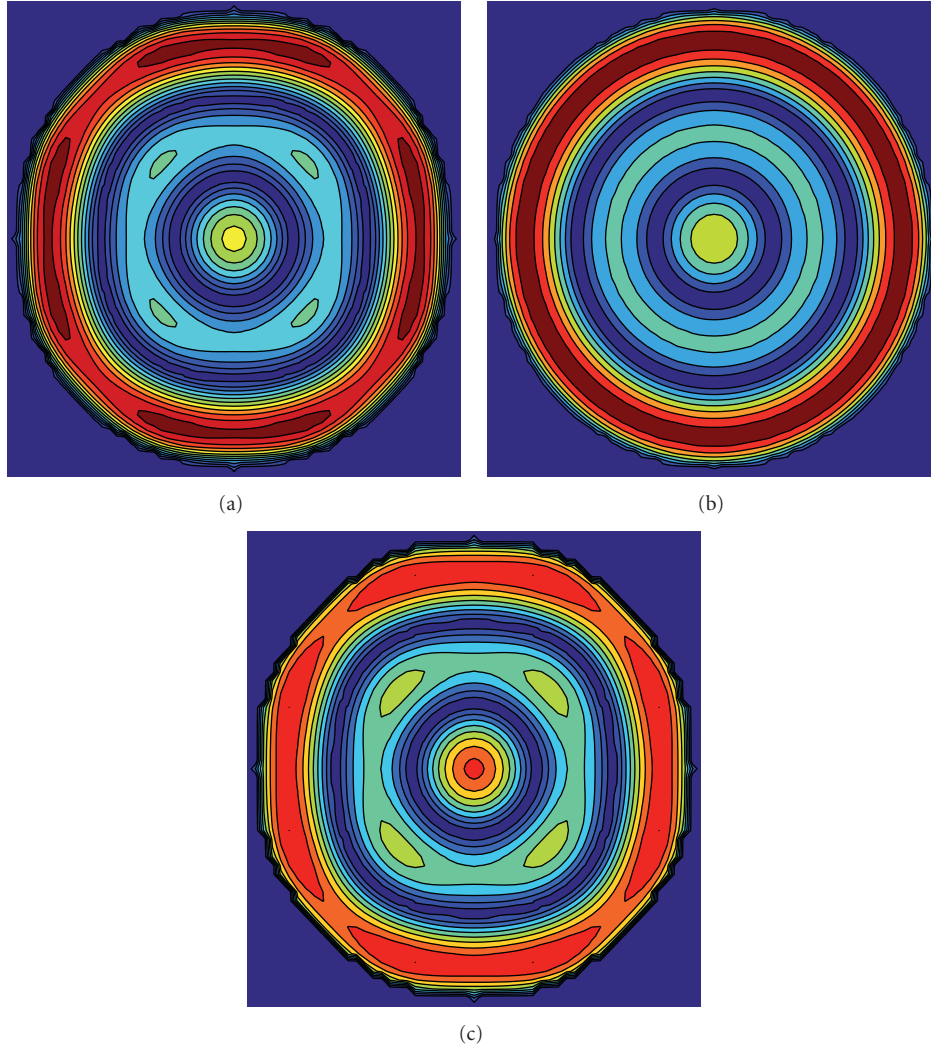


FIGURE 5: See the caption of Figure 3. The error for the sixth-order approximation for (a) Section 5 with D2V81 is slightly anisotropic while for (b) Section 5 with D2V141 the error is isotropic. The error for the eighth-order approximation for (c) Section 5 with D2V141 also exhibits small anisotropic patterns. The theoretically predicted isotropy properties are hence numerically confirmed. The colors are not scaled between the approximations.

polynomial tensors involve only even and odd rank monomials, respectively.

In a similar way, monomials can be expressed as linear combinations of Hermite polynomial tensors. By using the explicit expressions for the Hermite polynomial tensors presented above, the first few monomials can be written as

$$\begin{aligned}
 \mathbf{M}^{(2)} &= \mathbf{H}^{(2)} + \Delta^{(1)}, \\
 \mathbf{M}^{(3)} &= \mathbf{H}^{(3)} + \mathbf{H}^{(1)} \otimes \Delta^{(1)}, \\
 \mathbf{M}^{(4)} &= \mathbf{H}^{(4)} + \mathbf{H}^{(2)} \otimes \Delta^{(1)} + \Delta^{(2)}, \\
 \mathbf{M}^{(5)} &= \mathbf{H}^{(5)} + \mathbf{H}^{(3)} \otimes \Delta^{(1)} + \mathbf{H}^{(1)} \otimes \Delta^{(2)}, \\
 \mathbf{M}^{(6)} &= \mathbf{H}^{(6)} + \mathbf{H}^{(4)} \otimes \Delta^{(1)} + \mathbf{H}^{(2)} \otimes \Delta^{(2)} + \Delta^{(3)},
 \end{aligned}$$

$$\mathbf{M}^{(7)} = \mathbf{H}^{(7)} + \mathbf{H}^{(5)} \otimes \Delta^{(1)} + \mathbf{H}^{(3)} \otimes \Delta^{(2)} + \mathbf{H}^{(1)} \otimes \Delta^{(3)},$$

$$\mathbf{M}^{(8)} = \mathbf{H}^{(8)} + \mathbf{H}^{(6)} \otimes \Delta^{(1)} + \mathbf{H}^{(4)} \otimes \Delta^{(2)} + \mathbf{H}^{(2)} \otimes \Delta^{(3)} + \Delta^{(4)}. \quad (\text{A.2})$$

Above we have used the identities $\Delta^{(1)} \otimes \Delta^{(1)} = 2\Delta^{(2)}$, $\Delta^{(1)} \otimes \Delta^{(2)} = 3\Delta^{(3)}$, and $\Delta^{(1)} \otimes \Delta^{(3)} = 4\Delta^{(4)}$ repeatedly in the derivation. The expressions for even and odd rank monomials include only even and odd rank Hermite polynomial tensors, respectively.

B. Proof of Equivalency between Quadrature Problems

We prove that a solution to either of the two quadrature problems, (17) or (18), will also be solution to the other one

(up to the same order). First we assume a solution w_i and \mathbf{v}_i for $i = 0, \dots, q-1$, obtained by solving the system of equations arising from (18). That is, (18) holds for $n \leq N$, where N is the given order. Next we write the tensor product of monomials in terms of Hermite polynomial tensors in a specific way:

$$\mathbf{M}^{(n)} \mathbf{M}^{(n)} = \mathbf{b}_n^{(0)} \mathbf{H}^{(n)} \mathbf{H}^{(n)} + \mathbf{b}_{n-1}^{(2)} \mathbf{H}^{(n-1)} \mathbf{H}^{(n-1)} + \dots + \mathbf{b}_0^{(2n)} \mathbf{H}^{(0)} \mathbf{H}^{(0)}, \quad (\text{B.1})$$

where the polynomial tensor coefficients $\mathbf{b}_l^{(k)}$ are left undetermined. Then we substitute this expression into the right-hand side of (17) and obtain

$$\int \omega(\mathbf{v}) \mathbf{M}^{(n)} \mathbf{M}^{(n)} d\mathbf{v} = \int \omega(\mathbf{v}) \left(\mathbf{b}_n^{(0)} \mathbf{H}^{(n)} \mathbf{H}^{(n)} + \dots + \mathbf{b}_0^{(2n)} \mathbf{H}^{(0)} \mathbf{H}^{(0)} \right) d\mathbf{v}. \quad (\text{B.2})$$

With the solution for (18), we can equate the right-hand side of the above equation to

$$\begin{aligned} & \mathbf{b}_n^{(0)} \sum_i w_i \mathbf{H}_i^{(n)} \mathbf{H}_i^{(n)} + \dots + \mathbf{b}_0^{(2n)} \sum_i w_i \mathbf{H}_i^{(0)} \mathbf{H}_i^{(0)} \\ &= \sum_i w_i \left(\mathbf{b}_n^{(0)} \mathbf{H}_i^{(n)} \mathbf{H}_i^{(n)} + \dots + \mathbf{b}_0^{(2n)} \mathbf{H}_i^{(0)} \mathbf{H}_i^{(0)} \right). \end{aligned} \quad (\text{B.3})$$

Because expression (B.1) is true for every \mathbf{v} , it is especially true for \mathbf{v}_i . Hence, we can use expression (B.1) in (B.3) to obtain the left-hand side of (17) which concludes the first half of our proof. In other words, the solution to (18) will also satisfy (17) for $n \leq N$; that is, the inner products of the monomials in discrete and continuous space will be also matched. In the second half of the proof it is shown that the solution to (17) will also satisfy (18). The proof proceeds in exactly the same way as the first half above and, hence, is not shown here.

C. Proof of the Summation Formula for the Monomials

We use induction to prove (20). The expansion is valid for $\mathbf{M}^{(0)}$ and $\mathbf{M}^{(1)}$. Then, by first using the induction assumption and (15),

$$\begin{aligned} \mathbf{M}_{\mathbf{pq}}^{(n+1)} &= \mathbf{v}_{\mathbf{p}} \mathbf{M}_{\mathbf{q}}^{(n)} \\ &= \mathbf{v}_{\mathbf{p}} \left(\mathbf{H}_{\mathbf{q}}^{(n)} + \mathbf{H}_{\mathbf{q}}^{(n-2)} \otimes \Delta_{\mathbf{q}}^{(1)} + \mathbf{H}_{\mathbf{q}}^{(n-4)} \otimes \Delta_{\mathbf{q}}^{(2)} + \dots \right) \\ &= \left(\mathbf{H}_{\mathbf{pq}}^{(n+1)} + \Delta_{\mathbf{pq}}^{(1)} \otimes \mathbf{H}_{\mathbf{q}}^{(n-1)} \right) \\ &\quad + \left(\mathbf{H}_{\mathbf{pq}}^{(n-1)} + \Delta_{\mathbf{pq}}^{(1)} \otimes \mathbf{H}_{\mathbf{q}}^{(n-3)} \right) \otimes \Delta_{\mathbf{q}}^{(1)} \\ &\quad + \left(\mathbf{H}_{\mathbf{pq}}^{(n-3)} + \Delta_{\mathbf{pq}}^{(1)} \otimes \mathbf{H}_{\mathbf{q}}^{(n-5)} \right) \otimes \Delta_{\mathbf{q}}^{(2)} + \dots \end{aligned} \quad (\text{C.1})$$

By rearranging the terms on the right-hand side we get

$$\begin{aligned} & \mathbf{H}_{\mathbf{pq}}^{(n+1)} + \left(\Delta_{\mathbf{pq}}^{(1)} \otimes \mathbf{H}_{\mathbf{q}}^{(n-1)} + \mathbf{H}_{\mathbf{pq}}^{(n-1)} \otimes \Delta_{\mathbf{q}}^{(1)} \right) \\ &+ \left(\Delta_{\mathbf{pq}}^{(1)} \otimes \Delta_{\mathbf{q}}^{(1)} \otimes \mathbf{H}_{\mathbf{q}}^{(n-3)} + \mathbf{H}_{\mathbf{pq}}^{(n-3)} \otimes \Delta_{\mathbf{q}}^{(2)} \right) \\ &+ \Delta_{\mathbf{pq}}^{(1)} \otimes \Delta_{\mathbf{q}}^{(2)} \otimes \mathbf{H}_{\mathbf{q}}^{(n-5)} + \dots \end{aligned} \quad (\text{C.2})$$

which, after applying (6), becomes

$$\begin{aligned} & \mathbf{H}_{\mathbf{pq}}^{(n+1)} + \left(\Delta_{\mathbf{pq}}^{(1)} \otimes \mathbf{H}_{\mathbf{q}}^{(n-1)} + \mathbf{H}_{\mathbf{pq}}^{(n-1)} \otimes \Delta_{\mathbf{q}}^{(1)} \right) \\ &+ \left(\Delta_{\mathbf{pq}}^{(2)} \otimes \mathbf{H}_{\mathbf{q}}^{(n-3)} + \mathbf{H}_{\mathbf{pq}}^{(n-3)} \otimes \Delta_{\mathbf{q}}^{(2)} \right) \\ &+ \Delta_{\mathbf{pq}}^{(3)} \otimes \mathbf{H}_{\mathbf{q}}^{(n-5)} + \dots \end{aligned} \quad (\text{C.3})$$

Finally, we use (10) and arrive at

$$\begin{aligned} \mathbf{M}_{\mathbf{pq}}^{(n+1)} &= \mathbf{H}_{\mathbf{pq}}^{(n+1)} + \left(\mathbf{H}^{(n-1)} \otimes \Delta^{(1)} \right)_{\mathbf{pq}} \\ &+ \left(\mathbf{H}^{(n-3)} \otimes \Delta^{(2)} \right)_{\mathbf{pq}} + \dots \end{aligned} \quad (\text{C.4})$$

The expansion terminates properly, and hence we have completed the induction.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

Petrobras funding is gratefully acknowledged.

References

- [1] A. Kumar, "Isotropic finite-differences," *Journal of Computational Physics*, vol. 201, no. 1, pp. 109–118, 2004.
- [2] M. Patra and M. Karttunen, "Stencils with isotropic discretization error for differential operators," *Numerical Methods for Partial Differential Equations*, vol. 22, no. 4, pp. 936–953, 2006.
- [3] X. Shan and H. Chen, "Lattice Boltzmann model for simulating flows with multiple phases and components," *Physical Review E*, vol. 47, no. 3, pp. 1815–1819, 1993.
- [4] P. Yuan and L. Schaefer, "Equations of state in a lattice Boltzmann model," *Physics of Fluids*, vol. 18, no. 4, Article ID 042101, 2006.
- [5] P. C. Philippi, L. A. Hegele, L. O. E. Dos Santos, and R. Surmas, "From the continuous to the lattice Boltzmann equation: the discretization problem and thermal models," *Physical Review E*, vol. 73, no. 5, Article ID 056702, 2006.
- [6] X. Shan, X.-F. Yuan, and H. Chen, "Kinetic theory representation of hydrodynamics: a way beyond the Navier-Stokes equation," *Journal of Fluid Mechanics*, vol. 550, pp. 413–441, 2006.
- [7] H. Chen, I. Goldhirsch, and S. A. Orszag, "Discrete rotational symmetry, moment isotropy, and higher order lattice Boltzmann models," *Journal of Scientific Computing*, vol. 34, no. 1, pp. 87–112, 2008.

- [8] X. Shan, "Analysis and reduction of the spurious current in a class of multiphase lattice Boltzmann models," *Physical Review Letters E*, vol. 73, no. 4, Article ID 047701, 2006.
- [9] G. Falcucci, G. Bella, G. Chiatti, S. Chibbaro, M. Sbragaglia, and S. Succi, "Lattice Boltzmann models with mid-range interactions," *Communications in Computational Physics*, vol. 2, no. 6, pp. 1071–1084, 2007.
- [10] M. Sbragaglia, R. Benzi, L. Biferale, S. Succi, K. Sugiyama, and F. Toschi, "Generalized lattice Boltzmann method with multi-range pseudopotential," *Physical Review E*, vol. 75, no. 2, Article ID 026702, 2007.
- [11] T. Lee and C.-L. Lin, "A stable discretization of the lattice Boltzmann equation for simulation of incompressible two-phase flows at high density ratio," *Journal of Computational Physics*, vol. 206, no. 1, pp. 16–47, 2005.
- [12] T. Lee and P. F. Fischer, "Eliminating parasitic currents in the lattice Boltzmann equation method for nonideal gases," *Physical Review E*, vol. 74, no. 4, Article ID 046709, 2006.
- [13] P. Philippi, K. Mattila, D. Siebert, L. dos Santos, L. Hegele Jr, and R. Surmas, "Lattice-Boltzmann equations for describing segregation in non-ideal mixtures," *Journal of Fluid Mechanics*, vol. 713, pp. 564–587, 2012.
- [14] S. Thampi, S. Ansumali, R. Adhikari, and S. Succi, "Isotropic discrete Laplacian operators from lattice hydrodynamics," *Journal of Computational Physics*, vol. 234, no. 1, pp. 1–7, 2013.
- [15] R. Ramadugu, S. Thampi, R. Adhikari, S. Succi, and S. Ansumali, "Lattice differential operators for computational physics," *Europhysics Letters*, vol. 101, no. 5, article 50006, 2013.
- [16] H. Grad, "Note on N-dimensional hermite polynomials," *Communications on Pure and Applied Mathematics*, vol. 2, no. 4, pp. 325–330, 1949.
- [17] J. Shim and R. Gatignol, "How to obtain higher-order multivariate Hermite expansion of Maxwell-Boltzmann distribution by using Taylor expansion?" *Zeitschrift für Angewandte Mathematik und Physik*, vol. 64, no. 3, pp. 473–482, 2013.
- [18] Y. Qian, D. d'Humières, and P. Lallemand, "Lattice BGK models for navier-stokes equation," *Europhysics Letters*, vol. 17, no. 6, article 479, 1992.
- [19] K. Mattila, D. Siebert, L. Hegele Jr., and P. Philippi, "High-order lattice-Boltzmann equations and stencils for multiphase models," *International Journal of Modern Physics C*, vol. 24, no. 12, Article ID 1340006, 2013.