SIAM J. SCI. COMPUT.
Vol. 42, No. 5, pp. B1250–B1270

VECTOR-TYPE BOUNDARY SCHEMES FOR THE LATTICE BOLTZMANN METHOD BASED ON VECTOR-BGK MODELS*

JIN ZHAO†, ZHIMIN ZHANG‡, AND WEN-AN YONG§

Abstract. In this paper, we propose a vector-type bounce-back boundary scheme for the lattice Boltzmann method based on vector-BGK models. The scheme is shown to have second-order accuracy if the boundary is located at the middle of two neighboring lattice nodes. We analyze the numerical stability of the new boundary scheme by using a subtle structural property of the models. Based on the new scheme, we devise a family of parameterized second-order boundary schemes with accuracy independent of the location of the boundary. The accuracy and stability of the boundary schemes above are validated via numerical experiments with both straight and curved boundaries.

Key words. vector-type bounce-back scheme, lattice Boltzmann method, incompressible Navier–Stokes equations, Maxwell iteration, weighted L^2-stability

AMS subject classifications. 65M12, 76M28, 76P05, 76M20

DOI. 10.1137/19M1308542

1. Introduction.

The goal of this paper is to present a novel bounce-back boundary scheme for the vectorial lattice Boltzmann method (LBM). The LBM is an effective mesoscopic numerical approach to simulate complex fluid flows [17, 27]. Owing to its kinetic nature and easy implementation, the LBM has been successfully used in a range of applications including microflows, particulate suspensions, multiphase flows, and fluid turbulence [1, 9, 28]. The traditional LBM is often based on scalar discrete-velocity kinetic models such as the Bhatnagar–Gross–Krook (BGK) [3] model, the multi-relation-time model [11], the two-relaxation-time model [15], and the entropic LBM [2, 23, 24].

Despite the great success, the scalar models were pointed out by Dellar in [10] to not be possible in the construction of kinetic formulations for the magnetic induction equation, in a way analogous to the use of the scalar-BGK equation to simulate the vector Navier–Stokes momentum equations. Moreover, following Bouchut's vector-BGK models proposed in [6], the author of [10] introduced an effective vectorial kinetic model for the induction equation.

In recent years, vectorial kinetic models and corresponding numerical methods have drawn more and more attention [4, 5, 7, 12, 19]. For example, it was shown in [12] that the vectorial LBM can be used to simulate strong nonlinear waves in nonstationary situations. In [19], a vectorial kinetic method was proposed to solve the incompressible Navier–Stokes equations at pore scale and excellent results were obtained. On the other hand, from the viewpoint of partial differential equations it

*Submitted to the journal's Computational Methods in Science and Engineering section December 23, 2019; accepted for publication (in revised form) July 20, 2020; published electronically October 13, 2020.

https://doi.org/10.1137/19M1308542

Funding: This work was supported by the NSFC through grant U1930402.

†School of Mathematical Sciences, Peking University, Beijing 100871, People's Republic of China, and Division of Applied and Computational Mathematics, Beijing Computational Science Research Center, Beijing 100193, People's Republic of China (zjin@csrc.ac.cn).

‡Division of Applied and Computational Mathematics, Beijing Computational Science Research Center, Beijing 100193, People's Republic of China (zmzhang@csrc.ac.cn).

§Department of Mathematical Sciences, Tsinghua University, Beijing 100084, People's Republic of China (wayong@tsinghua.edu.cn).

was proved in [4, 5, 31] that the vector-BGK models can be regarded as good approximations to the incompressible Navier–Stokes equations. Furthermore, the vectorial kinetic models require fewer discrete velocities. For example, a 5-velocity vectorial LBM was developed in [30] to solve the two-dimensional incompressible Navier–Stokes equations successfully. The 7-velocity LBM is also available for the three-dimensional problems. The fewer discrete velocities enable us to handle complex geometries easily.

Nevertheless, boundary schemes for the vectorial LBM are seldom studied and, to the best of our knowledge, [7] and [12] are the only known works in the literature. In [12], the boundary conditions were treated with an easy adaptation of the usual bounce-back and "anti-bounce-back" methods. And the standard ghost cell method was used in [7] to handle the Dirichlet boundary conditions.

In this work, we first propose a bounce-back boundary scheme for the vectorial LBM with the Dirichlet boundary conditions. This scheme relies on the vector nature of the distribution functions, which allows us to treat each component separately. Consequently, it is quite different from the existing boundary schemes for the scalar LBM (see, e.g., [8, 16, 32]). By using the Maxwell iteration [33], we show that the new boundary scheme has second-order accuracy when the boundary is located at the middle of two neighboring lattice nodes. We also prove the numerical stability of the vector-type boundary scheme and the proof crucially utilizes a subtle structural property (4.9) of the lattice Boltzmann model. Moreover, on the basis of this vector-type boundary scheme, we follow [32] and devise a family of parameterized second-order boundary schemes with accuracy independent of the boundary location. Finally, the accuracy and stability of the boundary schemes are validated through numerical experiments with both straight and curved boundaries.

This paper is organized as follows. In section 2, we introduce the LBM based on the vector-BGK collision model and discuss its consistency to the incompressible Navier–Stokes equations. A vector-type bounce-back boundary scheme with its accuracy analysis is proposed in section 3. We show the stability of the vectorial LBM together with the new boundary scheme in section 4. In section 5 we devise a class of parameterized second-order boundary schemes with accuracy independent of the location of the boundary. Numerical experiments are reported in section 6. Finally, some conclusions and remarks are given in section 7.

2. Preliminary. In this section, we introduce the LBM based on the vector-BGK models [30] and show its consistency to the incompressible Navier–Stokes equations. For the sake of simplicity, we only consider a two-dimensional with five discrete velocities (D2N5) model. The extension to other models is straightforward.

The LBM reads as

$$f_i(t+\Delta t, x+he_i) - f_i(t,x) = \frac{M_i - f_i}{\tau}(t,x), \quad i = 1,2,\ldots,5. \tag{2.1}$$

Here $f_i = f_i(t,x)$ is a vector-valued distribution function taking values in $\mathbb{R}^3$; Δt, h are the time step and mesh size, respectively; the discrete velocities are $e_1 = -e_3 = (1,0)$, $e_2 = -e_4 = (0,1)$ and $e_5 = (0,0)$; and τ is a relaxation time. The equilibrium distribution function (Maxwellian) $M_i = M_i(w) \in \mathbb{R}^3$ is defined as

$$M_i = M_i(w) = \begin{cases} aw \pm \dfrac{1}{2\lambda} A_1, & i = 1,3, \\ aw \pm \dfrac{1}{2\lambda} A_2, & i = 2,4, \\ (1-4a)w, & i = 5, \end{cases} \tag{2.2}$$

where

(2.3)
$$w=\sum_{i=1}^{5} f_i := \begin{pmatrix} w_0 \\ w_1 \\ w_2 \end{pmatrix}, \quad A_1 = A_1(w) = \begin{pmatrix} w_1 \\ \frac{w_1^2+w_0^2}{w_0} \\ \frac{w_1 w_2}{w_0} \end{pmatrix}, \quad A_2 = A_2(w) = \begin{pmatrix} w_2 \\ \frac{w_1 w_2}{w_0} \\ \frac{w_2^2+w_0^2}{w_0} \end{pmatrix}$$

with a and λ two positive numbers [30]. For this D2N5 model, it is easy to verify that

$$\sum_{i=1}^{5} M_i(w) = w, \quad \sum_{i=1}^{5} e_{ij} M_i(w) = \frac{A_j(w)}{\lambda}, \quad \sum_{i=1}^{5} (e_i \cdot \nabla)^2 M_i(w) = 2a \sum_{j=1}^{2} \partial_{x_j}^2 w, \tag{2.4}$$

$$\sum_{i=1}^{5} (e_i \cdot \nabla)^3 M_i(w) = \frac{1}{\lambda} \sum_{j=1}^{2} \partial_{x_j}^3 A_j(w).$$

Here e_{ij} is the jth component of the ith discrete velocity e_i and ∇ is the usual gradient operator.

Next we show how to recover the incompressible Navier–Stokes equations from the LBM (2.1) together with (2.2) under the diffusive scaling

$$\Delta t = \alpha h^2 \tag{2.5}$$

with α a positive parameter. Recall that h is the spatial mesh size. In [31], the fluid density ρ and velocity $u = (u_1, u_2)$ are related to $w = \sum_{i=1}^{5} f_i$ as

$$\rho = w_0 \quad \text{and} \quad \rho u_l = w_l/h, \qquad l = 1, 2.$$

Thus, we may rewrite

(2.6)
$$w = \begin{pmatrix} \rho \\ h\rho u_1 \\ h\rho u_2 \end{pmatrix}, \quad A_1(w) = \begin{pmatrix} h\rho u_1 \\ h^2 \rho u_1^2 + \rho \\ h^2 \rho u_1 u_2 \end{pmatrix} \quad \text{and} \quad A_2(w) = \begin{pmatrix} h\rho u_2 \\ h^2 \rho u_1 u_2 \\ h^2 \rho u_2^2 + \rho \end{pmatrix}.$$

With the diffusive scaling (2.5), the left-hand side of (2.1) can be expanded at (t, x) as

$$f_i(t + \alpha h^2, x + h e_i) - f_i(t, x) \sim \sum_{k=1}^{\infty} h^k D_{i,k} f_i(t, x). \tag{2.7}$$

Here the differential operator $D_{i,k}$ is

$$D_{i,k} = \sum_{m+2n=k} \frac{(e_i \cdot \nabla)^m (\alpha \partial_t)^n}{m! n!}$$

with m and n two nonnegative integers. Specifically, we have $D_{i,0} = id$, $D_{i,1} = e_i \cdot \nabla$, $D_{i,2} = \alpha \partial_t + \frac{1}{2}(e_i \cdot \nabla)^2$, $D_{i,3} = \alpha \partial_t (e_i \cdot \nabla) + \frac{1}{6}(e_i \cdot \nabla)^3, \ldots$. With the expansion (2.7), the LBM (2.1) becomes

$$f_i(t, x) = M_i(t, x) - \tau \sum_{k=1}^{\infty} h^k D_{i,k} f_i(t, x).$$

Substituting the last equation into its right-hand side repeatedly (this is the Maxwell iteration [33]), we get the formal expansion $f_i(t,x)$ in terms of $M_i(t,x)$:

$$f_i(t,x) = \sum_{k\geq 0} h^k a_i^{(k)} M_i(t,x). \tag{2.8}$$

Here the expansion coefficients $a_i^{(k)}$ are differential operators that can be directly computed as

$$\begin{aligned}
a_i^{(0)} &= id,\\
a_i^{(1)} &= -\tau\left(e_i\cdot\nabla\right),\\
a_i^{(2)} &= -\tau\alpha\partial_t + \tau\left(\tau-\frac{1}{2}\right)\left(e_i\cdot\nabla\right)^2,\\
a_i^{(3)} &= \alpha\tau(2\tau-1)\partial_t\left(e_i\cdot\nabla\right) - \tau\left(\tau^2-\tau+\frac{1}{6}\right)\left(e_i\cdot\nabla\right)^3,\\
&\cdots\cdots
\end{aligned}$$

Under the incompressibility assumption

$$\rho = \overline{\rho} + O(h^2) \tag{2.9}$$

with $\overline{\rho}$ the mean density, it follows from (2.4) and (2.6) that

$$\sum_{i=1}^{5}\left(e_i\cdot\nabla\right)M_i = \sum_{i=1}^{5}\sum_{j=1}^{2} e_{ij}\partial_{x_j}M_i = \frac{1}{\lambda}\sum_{j=1}^{2}\partial_{x_j}A_j = O\begin{pmatrix}h\\h^2\\h^2\end{pmatrix},$$

$$\sum_{i=1}^{5}\left(e_i\cdot\nabla\right)^3 M_i = \sum_{i=1}^{5}\sum_{j=1}^{2}\left(e_{ij}\partial_{x_j}\right)^3 M_i = \frac{1}{\lambda}\sum_{j=1}^{2}\partial_{x_j}^3 A_j = O\begin{pmatrix}h\\h^2\\h^2\end{pmatrix},$$

and thereby,

$$\sum_{i=1}^{5} a_i^{(3)}M_i = \alpha\tau(2\tau-1)\partial_t\sum_{i=1}^{5}\left(e_i\cdot\nabla\right)M_i - \tau\left(\tau^2-\tau+\frac{1}{6}\right)\sum_{i=1}^{5}\left(e_i\cdot\nabla\right)^3 M_i = O\begin{pmatrix}h\\h^2\\h^2\end{pmatrix}.$$

Furthermore, it follows from (2.6) and (2.9) that

$$\left(e_i\cdot\nabla\right)M_i = O(h) \quad \text{and} \quad \partial_t M_i = O(h). \tag{2.10}$$

We notice that the coefficient $a_i^{(j)}$, $j>0$, is a combination of the operator ∂_t and $(e_i\cdot\nabla)$ and therefore

$$a_i^{(j)}M_i = O(h), \quad j>1.$$

Thus, we sum up (2.8) over i to obtain

$$\sum_{i=1}^{5} f_i = \sum_{i=1}^{5} M_i - h\tau\sum_{i=1}^{5}(e_i\cdot\nabla)M_i - h^2\tau\alpha\partial_t\sum_{i=1}^{5}M_i + h^2\tau\left(\tau-\frac{1}{2}\right)\sum_{i=1}^{5}(e_i\cdot\nabla)^2 M_i + O\begin{pmatrix}h^4\\h^5\\h^5\end{pmatrix}.$$

By using (2.4), the last equation can be rewritten as

$$h\alpha\partial_t w + \frac{1}{\lambda}\sum_{j=1}^{2}\partial_{x_j}A_j = h(\tau-\frac{1}{2})2a\sum_{j=1}^{2}\partial_{x_j}^2 w + O\begin{pmatrix} h^3 \\ h^4 \\ h^4 \end{pmatrix}.$$

For this system of three equations, we divide its first component by αh and the rest by αh^2 to obtain

$$\partial_t\begin{pmatrix}\rho\\ \rho u_1\\ \rho u_2\end{pmatrix} + \frac{1}{\alpha\lambda}\left[\partial_{x_1}\begin{pmatrix}\rho u_1\\ \rho u_1^2\\ \rho u_1u_2\end{pmatrix} + \partial_{x_2}\begin{pmatrix}\rho u_2\\ \rho u_2u_1\\ \rho u_2^2\end{pmatrix} + \begin{pmatrix}0\\ \partial_{x_1}\left(\frac{\rho-\overline{\rho}}{h^2}\right)\\ \partial_{x_2}\left(\frac{\rho-\overline{\rho}}{h^2}\right)\end{pmatrix}\right]$$
$$= \frac{(2\tau-1)\,a}{\alpha}\sum_{j=1}^{2}\partial_{x_j}^2\begin{pmatrix}\rho\\ \rho u_1\\ \rho u_2\end{pmatrix} + O\left(h^2\right).$$

To see the Navier–Stokes equations, we take $\alpha\lambda = 1$ in the last equation. Moreover, we use the incompressibility assumption (2.9) to get

$$\begin{cases} \partial_{x_1}u_1 + \partial_{x_2}u_2 = O(h^2), \\ \bar{\rho}[\partial_t u_1 + \partial_{x_1}u_1^2 + \partial_{x_2}(u_1u_2)] + \partial_{x_1}\left(\frac{\rho-\bar{\rho}}{h^2}\right) = \nu\bar{\rho}\sum\limits_{j=1}^{2}\partial_{x_j}^2 u_1 + O(h^2), \\ \bar{\rho}[\partial_t u_2 + \partial_{x_1}(u_1u_2) + \partial_{x_2}u_2^2] + \partial_{x_2}\left(\frac{\rho-\bar{\rho}}{h^2}\right) = \nu\bar{\rho}\sum\limits_{j=1}^{2}\partial_{x_j}^2 u_2 + O(h^2), \end{cases} \tag{2.11}$$

where

$$\nu = \frac{(2\tau-1)\,a}{\alpha} \quad \text{or} \quad \nu = (2\tau-1)\,a\lambda \tag{2.12}$$

is the kinematic viscosity and

$$P \equiv \frac{\rho-\overline{\rho}}{h^2}$$

acts as the pressure. In conclusion, we have shown formally that the LBM (2.1) is consistent to the incompressible Navier–Stokes equations up to $O(h^2)$. In what follows, we set $\overline{\rho} = 1$.

3. Vector-type bounce-back scheme. In this section, we propose a vector-type bounce-back boundary scheme accompanying the vectorial LBM (2.1) for the incompressible Navier–Stokes equations with the Dirichlet boundary condition

$$u(t,x) = \phi(t,x), \quad x \in \partial\Omega, \tag{3.1}$$

where $\phi(t,x)$ is a given function of t and x on the boundary $\partial\Omega$ of the fluid domain Ω. For this purpose, we decompose (2.1) into the following two steps:

$$\text{Collision:} \quad f_i'\left(t_n,x\right) = f_i\left(t_n,x\right) + Q_i\left(t_n,x\right), \tag{3.2a}$$

$$\text{Transport:} \quad f_i\left(t_{n+1},x\right) = f_i'\left(t_n,x-he_i\right) \tag{3.2b}$$

for lattice node x in the computational domain Ω. Here $t_n = n\Delta t$ and $Q_i := \frac{1}{\tau}\left(M_i(w) - f_i\right)$.

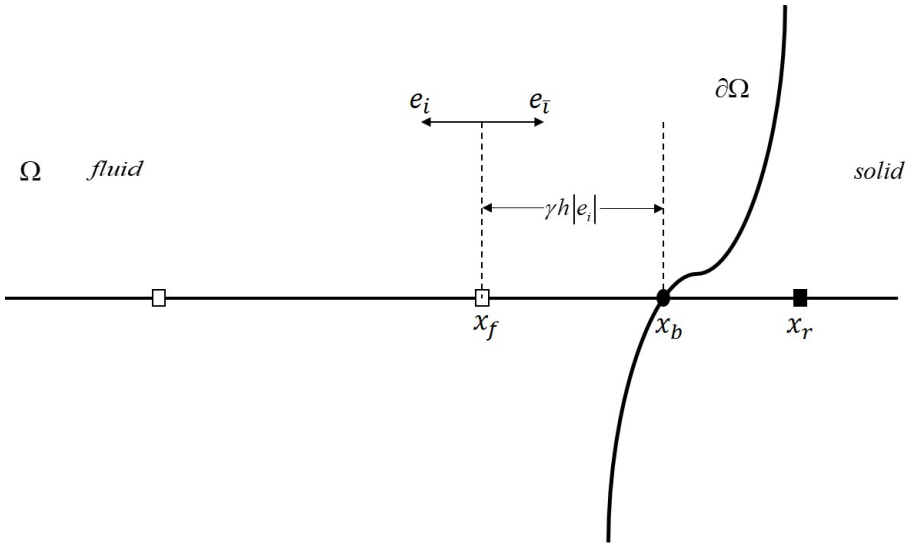

FIG. 1. *Illustration of the boundary. The square boxes (□) stand for lattice nodes in the fluid, the black circle (●) is an intersection of the boundary and the lattice, and the black square (■) stands for a lattice node out of the fluid.*

Notice that the transport step (3.2b) is not defined when the node $(x-he_i)$ is out of the domain Ω. See Figure 1, where $x_f \in \Omega$ but $x_r = x_f - he_i \notin \Omega$. In this situation, proper rules should be specified for the node x_f near the boundary $\partial\Omega$. The first idea to determine $f_i(t_{n+1}, x_f)$ is to use the classical bounce-back scheme [17, 25, 26, 27]

$$f_i(t_{n+1}, x_f) = f'_{\bar{i}}(t_n, x_f) + (M_i - M_{\bar{i}})(t_n, x_b). \tag{3.3}$$

Here $\bar{i}$ is the index such that $e_{\bar{i}} = -e_i$, x_b is the intersection of the lattice and the boundary (see Figure 1), and the equilibrium distributions $M_i(t_n, x_b)$ and $M_{\bar{i}}(t_n, x_b)$ are determined by the macroscopic fluid velocity u and density ρ, as shown in (2.2).

To see the accuracy of (3.3), we deduce from (2.8) together with (2.2), (2.6), and (2.10) that

$$f_i(t_n, x) = M_i(t_n, x) - h\tau\,(e_i \cdot \nabla)\, M_i(t_n, x) + O(h^3). \tag{3.4}$$

In addition, we refer to Figure 1 and define the scaled distance between x_f and x_b as

$$\gamma = \frac{|x_b - x_f|}{|x_r - x_f|} = \frac{|x_b - x_f|}{|he_i|}. \tag{3.5}$$

Remark that γ depends on i. Then we have $x_f = x_b + h\gamma e_i$.

With γ defined above and by using (2.10) and (3.4), we expand $f_i(t_{n+1}, x_f)$ and $f'_{\bar{i}}(t_n, x_f)$ (in the scheme (3.3)) at (t_n, x_b):

$$\begin{aligned} f_i(t_{n+1}, x_f) &= f_i(t_n + \alpha h^2, x_b + \gamma h e_i) \\ &= f_i(t_n, x_b) + h^2\alpha\partial_t f_i(t_n, x_b) + h\gamma\,(e_i \cdot \nabla)\, f_i(t_n, x_b) \\ &\quad + \frac{1}{2}h^2\gamma^2\,(e_i \cdot \nabla)^2\, f_i(t_n, x_b) + O(h^3) \\ &= M_i\,(t_n, x_b) + h(\gamma - \tau)\,(e_i \cdot \nabla)\, M_i\,(t_n, x_b) + O(h^3) \end{aligned} \tag{3.6}$$

and

$$
\begin{aligned}
f'_{\bar{i}}(t_n,x_f) &= f_{\bar{i}}(t_n,x_f)+\frac{1}{\tau}\left(M_{\bar{i}}-f_{\bar{i}}\right)(t_n,x_f)\\
&= M_{\bar{i}}(t_n,x_f)-h\tau(e_{\bar{i}}\cdot\nabla)M_{\bar{i}}(t_n,x_f)+h(e_{\bar{i}}\cdot\nabla)M_{\bar{i}}(t_n,x_f)+O(h^3)\\
&= M_{\bar{i}}(t_n,x_b-h\gamma e_{\bar{i}})-h\tau(e_{\bar{i}}\cdot\nabla)M_{\bar{i}}(t_n,x_b-h\gamma e_{\bar{i}})\\
&\quad +h(e_{\bar{i}}\cdot\nabla)M_{\bar{i}}(t_n,x_b-h\gamma e_{\bar{i}})+O(h^3)\\
&= M_{\bar{i}}(t_n,x_b)-h\gamma(e_{\bar{i}}\cdot\nabla)M_{\bar{i}}(t_n,x_b)-h\tau(e_{\bar{i}}\cdot\nabla)M_{\bar{i}}(t_n,x_b)\\
&\quad +h\left(e_{\bar{i}}\cdot\nabla\right)M_{\bar{i}}(t_n,x_b)+O(h^3)\\
&= M_{\bar{i}}(t_n,x_b)+h(\gamma+\tau-1)(e_i\cdot\nabla)M_{\bar{i}}(t_n,x_b)+O(h^3).
\end{aligned}
\tag{3.7}
$$

Thus, it follows from (2.2) and (2.6) that the error of the boundary scheme (3.3) is

$$
\begin{aligned}
E_{i-} :&= f_i(t_{n+1},x_f)-f'_{\bar{i}}(t_n,x_f)-\left(M_i-M_{\bar{i}}\right)(t_n,x_b)\\
&= h(\gamma-\tau)\left(e_i\cdot\nabla\right)M_i\left(t_n,x_b\right)-h(\gamma+\tau-1)\left(e_i\cdot\nabla\right)M_{\bar{i}}\left(t_n,x_b\right)+O(h^3)\\
&= h(\gamma-\tau)\left(e_i\cdot\nabla\right)\left(aw+\frac{1}{2\lambda}A^{(i)}\right)(t_n,x_b)\\
&\quad -h(\gamma+\tau-1)\left(e_i\cdot\nabla\right)\left(aw+\frac{1}{2\lambda}A^{(\bar{i})}\right)(t_n,x_b)+O(h^3)\\
&= h^2\left[\frac{(2\gamma-1)}{2\lambda}\left(e_i\cdot\nabla\right)\begin{pmatrix}\rho u^{(i)}\\0\\0\end{pmatrix}-(2\tau-1)a\left(e_i\cdot\nabla\right)\begin{pmatrix}0\\\rho u_1\\\rho u_2\end{pmatrix}\right](t_n,x_b)+O(h^3),
\end{aligned}
\tag{3.8}
$$

where $A^{(i)}$ and $u^{(i)}$ are defined as

$$
u^{(i)}=\frac{\bar{i}-i}{2}\begin{cases}u_1, & i=1,3,\\ u_2, & i=2,4,\end{cases}\quad\text{and}\quad A^{(i)}=\frac{\bar{i}-i}{2}\begin{cases}A_1(w), & i=1,3,\\ A_2(w), & i=2,4.\end{cases}
\tag{3.9}
$$

In the last step of (3.8) we have used the incompressibility assumption (2.9).

The above accuracy analysis in (3.8) indicates that the classical bounce-back boundary scheme (3.3) for the vectorial LBM (2.1) cannot reach second-order accuracy, since $\tau\neq 1/2$ due to the consistency constraint (2.12). Likewise, we can show that the anti-bounce-back boundary scheme [13, 14, 29]

$$
f_i(t_{n+1},x_f)=-f'_{\bar{i}}(t_n,x_f)+\left(M_i+M_{\bar{i}}\right)(t_n,x_b)
\tag{3.10}
$$

cannot reach second-order accuracy either, since

$$
\begin{aligned}
E_{i+} :&= f_i(t_{n+1},x_f)+f'_{\bar{i}}(t_n,x_f)-\left(M_i+M_{\bar{i}}\right)(t_n,x_b)\\
&= h^2\left[(2\gamma-1)a\left(e_i\cdot\nabla\right)\begin{pmatrix}0\\\rho u_1\\\rho u_2\end{pmatrix}-\frac{(2\tau-1)}{2\lambda}\left(e_i\cdot\nabla\right)\begin{pmatrix}\rho u^{(i)}\\0\\0\end{pmatrix}\right](t_n,x_b)+O\left(h^3\right).
\end{aligned}
\tag{3.11}
$$

However, the expansions (3.8) and (3.11) suggest to us a new boundary scheme for the vectorial LBM (3.2). The bounce-back scheme (3.3) is applied to the first component of the vector-valued distribution f_i, while the anti-bounce-back is adopted for the last two components. Thus, our new boundary scheme reads as

$$
\begin{pmatrix}f_{i0}\\f_{i1}\\f_{i2}\end{pmatrix}(t_{n+1},x_f)=\begin{pmatrix}f'_{\bar{i}0}\\-f'_{\bar{i}1}\\-f'_{\bar{i}2}\end{pmatrix}(t_n,x_f)+\begin{pmatrix}M_{i0}-M_{\bar{i}0}\\M_{i1}+M_{\bar{i}1}\\M_{i2}+M_{\bar{i}2}\end{pmatrix}(t_n,x_b).
\tag{3.12}
$$

For this new scheme, by combining (3.8) and (3.11) we have

$$E_{i\mp} := \begin{pmatrix} f_{i0} \\ f_{i1} \\ f_{i2} \end{pmatrix}(t_{n+1}, x_f) - \begin{pmatrix} f'_{\bar{i}0} \\ -f'_{\bar{i}1} \\ -f'_{\bar{i}2} \end{pmatrix}(t_n, x_f) - \begin{pmatrix} M_{i0} - M_{\bar{i}0} \\ M_{i1} + M_{\bar{i}1} \\ M_{i2} + M_{\bar{i}2} \end{pmatrix}(t_n, x_b)$$
$$= h^2 \left[(2\gamma - 1)(e_i \cdot \nabla) \begin{pmatrix} \rho u^{(i)}/2\lambda \\ a\rho u_1 \\ a\rho u_2 \end{pmatrix} \right](t_n, x_b) + O(h^3).$$

This indicates that the new vector-type boundary scheme has second-order accuracy when $\gamma = 1/2$ and is first-order accurate otherwise.

4. Stability analysis. This section is devoted to the numerical stability, with respect to disturbances of initial data, of the vectorial LBM (2.1) together with its vector-type bounce-back boundary scheme (3.12). In view of the convergence analyses in [20, 21], it is sufficient to consider the linearization of the vectorial LBM at the quiescent state $(\rho, u) = (1, 0)$.

To do this, we set $f = (f_1, \ldots, f_5)$, $M = (M_1, \ldots, M_5)$, and $Q = (M - f)/\tau$, where M_i is the ith equilibrium distribution (Maxwellian) defined in (2.2) and can be regarded as a function of f (see (2.3)). Denote by $f_* = M|_{(\rho,u)=(1,0)}$ the corresponding distribution at the quiescent state $(\rho, u) = (1, 0)$. Note that $Q(f_*) = 0$. Then the linearized scheme can be written as

$$f(t_{n+1}, x_j + he_i) = f(t_n, x_j) + \left.\frac{\partial Q}{\partial f}\right|_{f=f_*} f(t_n, x_j) \tag{4.1}$$

with the homogeneous vector-type bounce-back boundary scheme

$$\begin{pmatrix} f_{i0} \\ f_{i1} \\ f_{i2} \end{pmatrix}(t_{n+1}, x_f) = \begin{pmatrix} f'_{\bar{i}0} \\ -f'_{\bar{i}1} \\ -f'_{\bar{i}2} \end{pmatrix}(t_n, x_f), \tag{4.2}$$

where x_j's are the lattice nodes in the fluid domain Ω and x_f is a node near the boundary (see Figure 1). In what follows, we often use the usual two-step decomposition of the linearized LBM (4.1):

$$\text{Collision:} \quad f'(t_n, x_j) = f(t_n, x_j) + \left.\frac{\partial Q}{\partial f}\right|_{f=f_*} f(t_n, x_j), \tag{4.3a}$$

$$\text{Transport:} \quad f(t_{n+1}, x_j) = f'(t_n, x_j - he_i). \tag{4.3b}$$

Next we follow [22] and establish the following result for the linearized collision step (4.3a).

LEMMA 1. *Assume $\frac{1}{2\lambda} < a < \frac{1}{4}$ and $\tau > 1/2$. Then there is an invertible matrix $P \in \mathbb{R}^{15\times 15}$ such that $P^T P$ is a block-diagonal matrix and*

$$P \left.\frac{\partial Q}{\partial f}\right|_{f=f_*} = \Lambda P$$

with Λ a diagonal matrix. Moreover, $f'(t_n, x_j)$ satisfies

$$|Pf'(t_n, x_j)| \le |Pf(t_n, x_j)|$$

for each $n \ge 0$ and $x_j \in \Omega$. Here $|\cdot|$ denotes the usual 2-norm of vectors or matrices.

Proof. Note that $\frac{\partial Q}{\partial f} = (\frac{\partial M}{\partial f} - I_{15})/\tau$. We compute

$$\frac{\partial M}{\partial f} = \frac{\partial M}{\partial w}\frac{\partial w}{\partial f} = \begin{pmatrix} aI_3 + \frac{1}{2\lambda}\partial_w A_1(w) \\ aI_3 + \frac{1}{2\lambda}\partial_w A_2(w) \\ aI_3 - \frac{1}{2\lambda}\partial_w A_1(w) \\ aI_3 - \frac{1}{2\lambda}\partial_w A_2(w) \\ (1-4a)I_3 \end{pmatrix} \begin{pmatrix} I_3 & I_3 & I_3 & I_3 & I_3 \end{pmatrix}.$$

It is obvious that $(\frac{\partial M}{\partial f})^2 = \frac{\partial M}{\partial f}$. Namely, $\frac{\partial M}{\partial f}$ is a projection. At $f_* = M|_{(\rho,u)=(1,0)}$ we have

$$\left.\frac{\partial M}{\partial f}\right|_{f=f_*} = \begin{pmatrix} K_1 \\ K_2 \\ \vdots \\ K_5 \end{pmatrix} \begin{pmatrix} I_3 & I_3 & \cdots & I_3 \end{pmatrix}.$$

Here

(4.4)

$$K_{1,3} = aI_3 \pm \frac{1}{2\lambda}\begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \quad K_{2,4} = aI_3 \pm \frac{1}{2\lambda}\begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}, \quad K_5 = (1-4a)I_3.$$

It is easy to verify that the K_i's are all symmetric positive definite if and only if

$$\frac{1}{2\lambda} < a < \frac{1}{4}. \tag{4.5}$$

In this situation, the projection matrix $\frac{\partial M}{\partial f}|_{f=f_*}$ has rank 3 and is similar to $\mathrm{diag}(I_3, 0)$.

On the other hand, we set

$$K_0 = \mathrm{diag}\left(K_1^{-1}, K_2^{-1}, \ldots, K_5^{-1}\right).$$

Since $K_0\frac{\partial M}{\partial f}|_{f=f_*}$ is symmetric and K_0 is symmetric positive definite, it is well-known that there is an invertible matrix P such that

$$K_0 = P^T P \qquad \text{and} \qquad K_0 \left.\frac{\partial M}{\partial f}\right|_{f=f_*} = P^T \mathrm{diag}(I_3, 0) P.$$

Thus, we have

$$P \left.\frac{\partial Q}{\partial f}\right|_{f=f_*} = \Lambda P$$

with $\Lambda = -\mathrm{diag}(0, I_{12})/\tau$.

Now we multiply (4.3a) with P from the left and get

$$Pf'(t_n, x_j) = (I_{15} + \Lambda)\, Pf(t_n, x_j).$$

For $\tau > 1/2$ it follows that

$$\begin{aligned} |Pf'(t_n, x_j)| &= |(I_{15} - \mathrm{diag}(0, I_{12})/\tau)\, Pf(t_n, x_j)| \\ &\le \max|(I_{15} - \mathrm{diag}(0, I_{12})/\tau)|\, |Pf(t_n, x_j)| \le |Pf(t_n, x_j)|. \end{aligned}$$

This completes the proof. ∎

Now we turn to the transport step (4.3b). For this, we introduce the following weighted $\mathbb{L}^2$-norm:

$$\|f(t_n)\|_P = \Big(\sum_{x_j \in \Omega} |Pf(t_n, x_j)|^2 \Big)^{\frac{1}{2}}.$$

With this norm, Lemma 1 implies that

$$\|f'(t_n)\|_P \le \|f(t_n)\|_P . \tag{4.6}$$

And we are in a position to state the following result for the transport step (4.3b).

LEMMA 2. *Assume* $\frac{1}{2\lambda} < a < \frac{1}{4}$. *Then, for the transport step* (4.3b) *with the vector-type bounce-back boundary scheme* (4.2) *in a bounded domain* Ω, *we have*

$$\|f(t_{n+1})\|_P = \|f'(t_n)\|_P .$$

Proof. First, we refer to the transport step (4.3b) and define

$$\partial_i\Omega := \{x_j \in \Omega : x_j - he_i \notin \Omega\}$$

for each i. Namely, $\partial_i\Omega$ consists exactly of the boundary nodes where the transport step for f_i does not apply. For $\partial_i\Omega$ thus defined, we have

$$\{\Omega\backslash\partial_i\Omega - he_i\} = \Omega\backslash\partial_{\bar{i}}\Omega. \tag{4.7}$$

Indeed, for $x_j \in \{\Omega\backslash\partial_i\Omega - he_i\}$ we can write $x_j = x_j' - he_i$ with $x_j' \in \Omega\backslash\partial_i\Omega$, by which we have $x_j = x_j' - he_i \in \Omega$ and $x_j - he_{\bar{i}} = x_j' \in \Omega$. That is, $x_j \in \Omega\backslash\partial_{\bar{i}}\Omega$. Conversely, for $x_j \in \Omega\backslash\partial_{\bar{i}}\Omega$ we have $x_j, x_j - he_{\bar{i}} \in \Omega$ and $x_j = (x_j - he_{\bar{i}}) - he_i \in \{\Omega\backslash\partial_i\Omega - he_i\}$. This proves (4.7).

Next we recall the proof of Lemma 1. It is known that

$$P^TP = \mathrm{diag}(K_1^{-1}, K_2^{-1}, \dots, K_5^{-1})$$

and each K_i^{-1} is a symmetric positive definite matrix under the condition (4.5). Thus we set $|f_i|_{k_i}^2 = f_i^T K_i^{-1} f_i$ and have

$$\begin{aligned}
\|f(t_{n+1})\|_P^2 :&= \sum_{x_j\in\Omega} |Pf(t_{n+1}, x_j)|^2 = \sum_{x_j\in\Omega} f(t_{n+1}, x_j)^T P^T P f(t_{n+1}, x_j) \\
&= \sum_{x_j\in\Omega} f(t_{n+1}, x_j)^T \mathrm{diag}(K_1^{-1}, K_2^{-1}, \dots, K_5^{-1}) f(t_{n+1}, x_j) \\
&= \sum_{x_j\in\Omega} \sum_{i=1}^{5} |f_i(t_{n+1}, x_j)|_{k_i}^2 \\
&= \sum_{i=1}^{5} \Big(\sum_{x_j\in\Omega\backslash\partial_i\Omega} + \sum_{x_j\in\partial_i\Omega} \Big) |f_i(t_{n+1}, x_j)|_{k_i}^2 \\
&= \sum_{i=1}^{5} \frac{1}{2} \Bigg[\Big(\sum_{x_j\in\Omega\backslash\partial_i\Omega} + \sum_{x_j\in\partial_i\Omega} \Big) |f_i(t_{n+1}, x_j)|_{k_i}^2 \\
&\qquad + \Big(\sum_{x_j\in\Omega\backslash\partial_{\bar{i}}\Omega} + \sum_{x_j\in\partial_{\bar{i}}\Omega} \Big) |f_{\bar{i}}(t_{n+1}, x_j)|_{k_{\bar{i}}}^2 \Bigg].
\end{aligned} \tag{4.8}$$

In the last step we have used the symmetry of the set of the discrete velocities.

For the terms in the last line of (4.8), we observe from (4.4) that

$$K_i = \mathrm{diag}(1,-1,-1)K_{\bar{i}}\mathrm{diag}(1,-1,-1). \tag{4.9}$$

Then it follows from the vector-type bounce-back scheme (4.2) that

$$\begin{aligned} &\sum_{x_j\in\partial_i\Omega} |f_i(t_{n+1},x_j)|^2_{k_i} = \sum_{x_j\in\partial_i\Omega} \left|f'_{\bar{i}}(t_n,x_j)\right|^2_{k_{\bar{i}}} \\ &\text{and} \quad \sum_{x_j\in\partial_{\bar{i}}\Omega} |f_{\bar{i}}(t_{n+1},x_j)|^2_{k_{\bar{i}}} = \sum_{x_j\in\partial_{\bar{i}}\Omega} |f'_i(t_n,x_j)|^2_{k_i}. \end{aligned} \tag{4.10}$$

On the other hand, for $x_j \in \Omega\backslash\partial_i\Omega$ we deduce from the transport step (4.3b) that

$$\sum_{x_j\in\Omega\backslash\partial_i\Omega} |f_i(t_{n+1},x_j)|^2_{k_i} = \sum_{x_j\in\Omega\backslash\partial_i\Omega} |f'_i(t_n,x_j-he_i)|^2_{k_i} = \sum_{x_j\in\Omega\backslash\partial_i\Omega-he_i} |f'_i(t_n,x_j)|^2_{k_i}.$$

Similarly, we have

$$\sum_{x_j\in\Omega\backslash\partial_{\bar{i}}\Omega} |f_{\bar{i}}(t_{n+1},x_j)|^2_{k_{\bar{i}}} = \sum_{x_j\in\Omega\backslash\partial_{\bar{i}}\Omega} \left|f'_{\bar{i}}(t_n,x_j-he_{\bar{i}})\right|^2_{k_{\bar{i}}} = \sum_{x_j\in\Omega\backslash\partial_{\bar{i}}\Omega-he_{\bar{i}}} \left|f'_{\bar{i}}(t_n,x_j)\right|^2_{k_{\bar{i}}}.$$

These equations, together with (4.8) and (4.7), yield

$$\begin{aligned} \|f(t_{n+1})\|^2_P &= \sum_{i=1}^{5} \frac{1}{2}\Bigg[\left(\sum_{x_j\in\Omega\backslash\partial_i\Omega-he_i} + \sum_{x_j\in\partial_{\bar{i}}\Omega}\right)|f'_i(t_n,x_j)|^2_{k_i} \\ &\qquad + \left(\sum_{x_j\in\Omega\backslash\partial_{\bar{i}}\Omega-he_{\bar{i}}} + \sum_{x_j\in\partial_i\Omega}\right)\left|f'_{\bar{i}}(t_n,x_j)\right|^2_{k_{\bar{i}}}\Bigg] \\ &= \sum_{i=1}^{5}\sum_{x_j\in\Omega} |f'_i(t_n,x_j)|^2_{k_i} = \|f'(t_n)\|^2_P. \end{aligned}$$

This completes the proof. ∎

Combining Lemmas 2 and 1 (or (4.6)), we arrive at the main result of this section.

THEOREM 1. *Under the condition of Lemma* 1*, the linearized vectorial LBM* (4.1) *with the vector-type bounce-back boundary scheme* (4.2) *is stable in the following sense:*

$$\|f(t_{n+1})\|_P \le \|f(t_n)\|_P$$

for every $n \ge 0$.

5. Second-order boundary schemes. In this section, we follow [32] and construct a parameterized second-order boundary scheme, with accuracy independent of the location of boundary, by using interpolations together with the vector-type bounce-back scheme (3.12)

$$\begin{pmatrix} f_{i0} \\ f_{i1} \\ f_{i2} \end{pmatrix}(t_{n+1},x_f) = \begin{pmatrix} f_{\bar{i}0} \\ -f_{\bar{i}1} \\ -f_{\bar{i}2} \end{pmatrix}(t_{n+1},x_r) + \begin{pmatrix} M_{i0}-M_{\bar{i}0} \\ M_{i1}+M_{\bar{i}1} \\ M_{i2}+M_{\bar{i}2} \end{pmatrix}(t_n,x_b). \tag{5.1}$$

This is a rewriting of the scheme (3.12) by using the transport step (3.2b). Here

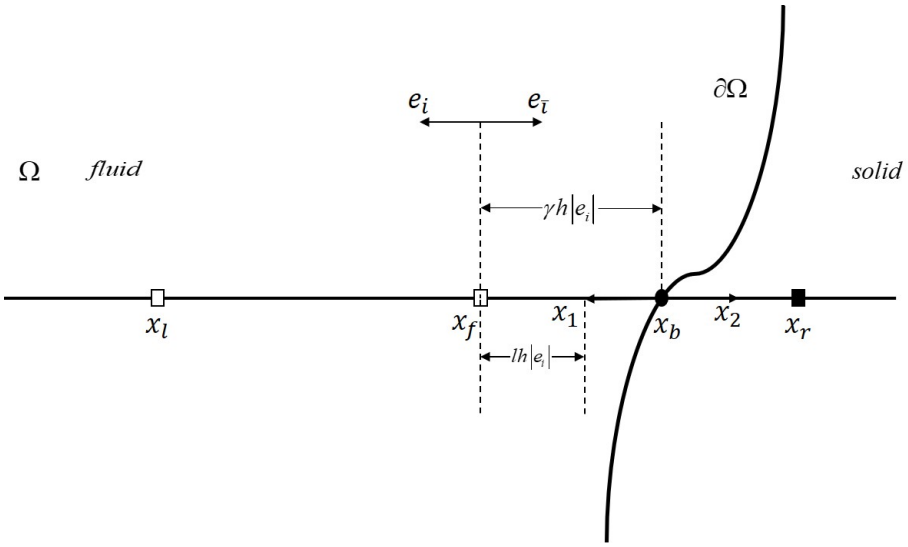

FIG. 2. *Configuration for boundary scheme* (5.7). x_l *and* x_f *are lattice nodes in the fluid domain* Ω, x_b *is an intersection of the boundary* ($\partial\Omega$) *and the lattice, and* x_r *is a node out of the fluid domain.*

x_f, x_r, and x_b are illustrated in Figure 2,

$$x_r = x_f - he_i, \quad x_b = x_f - \gamma he_i, \quad \gamma \in (0, 1],$$

and the scaled distance γ is defined in (3.5).

Recall that the vector-type bounce-back scheme has second-order accuracy only when $\gamma = \frac{1}{2}$. Bearing this fact in mind, we introduce two auxiliary points x_1 and x_2 equidistantly locating at the two sides of x_b (see Figure 2):

$$x_1 = x_f - lhe_i, \quad x_2 = 2x_b - x_1$$

with l a nonnegative number. Then the boundary scheme

$$\begin{pmatrix} f_{i0} \\ f_{i1} \\ f_{i2} \end{pmatrix}(t_{n+1}, x_1) = \begin{pmatrix} f_{\bar{i}0} \\ -f_{\bar{i}1} \\ -f_{\bar{i}2} \end{pmatrix}(t_{n+1}, x_2) + \begin{pmatrix} M_{i0} - M_{\bar{i}0} \\ M_{i1} + M_{\bar{i}1} \\ M_{i2} + M_{\bar{i}2} \end{pmatrix}(t_n, x_b) \tag{5.2}$$

has second-order accuracy for a fictitious lattice with size $2|\gamma - l|h$.

On the basis of (5.2), we can derive a formula to compute the distribution function $f_i(x_f, t_{n+1})$. First, $f_i(x_f, t_{n+1})$ can be interpolated as

$$f_i(t_{n+1}, x_f) = \frac{l}{1+l} f_i(t_{n+1}, x_l) + \frac{1}{1+l} f_i(t_{n+1}, x_1) \tag{5.3}$$

with $x_l = x_f + he_i$. Note that $f_i(t_{n+1}, x_l) = f_i'(t_n, x_f)$ due to the transport term (3.2b), while $f_i(t_{n+1}, x_1)$ is given in (5.2) in terms of $f_{\bar{i}}(t_{n+1}, x_2)$. The latter can be interpolated with those at x_f and x_r as

$$f_{\bar{i}}(t_{n+1}, x_2) = (1 + l - 2\gamma) f_{\bar{i}}(t_{n+1}, x_f) + (2\gamma - l) f_{\bar{i}}(t_{n+1}, x_r). \tag{5.4}$$

Note that $f_{\bar{i}}(t_{n+1}, x_r) = f'_{\bar{i}}(t_n, x_f)$ due to the transport term (3.2b). Consequently, we have

$$
\begin{aligned}
f_i(t_{n+1}, x_f) = {} & \frac{l}{1+l} f'_i(t_n, x_f) \\
& + \frac{1+l-2\gamma}{1+l} \begin{pmatrix} f_{\bar{i}0} \\ -f_{\bar{i}1} \\ -f_{\bar{i}2} \end{pmatrix} (t_{n+1}, x_f) + \frac{2\gamma - l}{1+l} \begin{pmatrix} f'_{\bar{i}0} \\ -f'_{\bar{i}1} \\ -f'_{\bar{i}2} \end{pmatrix} (t_n, x_f) \\
& + \frac{1}{1+l} \begin{pmatrix} M_{i0} - M_{\bar{i}0} \\ M_{i1} + M_{\bar{i}1} \\ M_{i2} + M_{\bar{i}2} \end{pmatrix} (t_n, x_b) \,.
\end{aligned} \tag{5.5}
$$

This has second-order accuracy, since so do (5.2) and the interpolations. Furthermore, the second term in the right-hand side of (5.5), depending on t_{n+1}, can be approximated as

$$
\begin{pmatrix} f_{\bar{i}0} \\ -f_{\bar{i}1} \\ -f_{\bar{i}2} \end{pmatrix} (t_{n+1}, x_f) \sim \begin{pmatrix} f_{\bar{i}0} \\ -f_{\bar{i}1} \\ -f_{\bar{i}2} \end{pmatrix} (t_n, x_f). \tag{5.6}
$$

This approximation is also of second-order accuracy thanks to (2.8), (2.10), and the diffusive scaling $\Delta t = O(h^2)$. Substituting (5.6) into (5.5), we arrive at the following single-node second-order boundary scheme:

$$
\begin{aligned}
f_i(t_{n+1}, x_f) = {} & \frac{l}{1+l} f'_i(t_n, x_f) \\
& + \frac{1+l-2\gamma}{1+l} \begin{pmatrix} f_{\bar{i}0} \\ -f_{\bar{i}1} \\ -f_{\bar{i}2} \end{pmatrix} (t_n, x_f) \\
& + \frac{2\gamma - l}{1+l} \begin{pmatrix} f'_{\bar{i}0} \\ -f'_{\bar{i}1} \\ -f'_{\bar{i}2} \end{pmatrix} (t_n, x_f) + \frac{1}{1+l} \begin{pmatrix} M_{i0} - M_{\bar{i}0} \\ M_{i1} + M_{\bar{i}1} \\ M_{i2} + M_{\bar{i}2} \end{pmatrix} (t_n, x_b) \,.
\end{aligned} \tag{5.7}
$$

This depends on the parameter l.

The stability of the boundary scheme (5.7) can be discussed simply as follows. It relies on that of (5.2) and the interpolations. The former is given in Theorem 1 and the latter is stable if all the interpolation coefficients belong to $[0, 1]$:

$$
l \geq 0, \quad 1 - 2\gamma + l \geq 0 \quad \text{and} \quad 2\gamma - l \geq 0.
$$

Namely,

$$
\max\{0, 2\gamma - 1\} \leq l \leq 2\gamma. \tag{5.8}
$$

We end this section with the following two remarks.

Remark 1. In deriving (5.7), the approximation (5.6) can be replaced with

$$
\begin{pmatrix} f_{\bar{i}0} \\ -f_{\bar{i}1} \\ -f_{\bar{i}2} \end{pmatrix} (t_{n+1}, x_f) = \begin{pmatrix} f'_{\bar{i}0} \\ -f'_{\bar{i}1} \\ -f'_{\bar{i}2} \end{pmatrix} (t_n, x_l)
$$

if the node x_l belongs to the computational domain. In this way, we get a two-node boundary scheme.

Remark 2. The scheme (5.7) does not contain $f_i(t_n, x_f)$. A convex combination of $f_i(t_n, x_f)$ and the right-hand side of (5.7) yields the following two-parameter second-order boundary scheme:

$$
\begin{aligned}
&f_i(t_{n+1}, x_f) = (1-b) f_i(t_n, x_f) \\
&+ b\left[\frac{l}{1+l} f_i'(t_n, x_f) + \frac{1+l-2\gamma}{1+l}\begin{pmatrix} f_{\bar{i}0} \\ -f_{\bar{i}1} \\ -f_{\bar{i}2} \end{pmatrix}(t_n, x_f) + \frac{2\gamma - l}{1+l}\begin{pmatrix} f'_{\bar{i}0} \\ -f'_{\bar{i}1} \\ -f'_{\bar{i}2} \end{pmatrix}(t_n, x_f)\right. \\
&\left. + \frac{1}{1+l}\begin{pmatrix} M_{i0} - M_{\bar{i}0} \\ M_{i1} + M_{\bar{i}1} \\ M_{i2} + M_{\bar{i}2} \end{pmatrix}(t_n, x_b)\right]
\end{aligned}
$$

with the parameter $b \in (0, 1]$.

6. Numerical experiments. In this section, we report several numerical experiments with the D2N5 model to validate the vector-type bounce-back boundary scheme (3.12) and the boundary scheme (5.7) for the incompressible Navier–Stokes equations with the Dirichlet boundary condition (3.1). As to the parameter l in (5.7), we will take the following four choices [32], $l = \gamma,\ 1.5\gamma,\ 2\gamma$, and γ^2, which all satisfy the stability constraint (5.8).

On the other hand, we consider the following three problems: the Poiseuille flow, the Couette flow with straight boundaries, and the Taylor–Green vortex flow with curved boundaries. All the flows are governed by the two-dimensional incompressible Navier–Stokes equation with the viscosity (2.12), where we take $a = 1/5,\ \lambda = 3$ due to the stability condition in Theorem 1. These problems all have analytical solutions so that the convergence order can be conveniently computed.

6.1. Poiseuille flow. This means the flow between two parallel no-slip walls driven by a constant body force $F = G(1, 0)$ (see Figure 3). This problem has the analytical solution

$$
u_1 = u(y) = 4U\left(1 - \frac{y}{H}\right)\frac{y}{H}, \quad u_2 = 0 \tag{6.1}
$$

for $y \in [0, H]$. Here H is the channel width and $U = GH^2/8\nu$ is the maximal velocity along the center line of the channel. In what follows, we take

$$
G = 0.8\nu, \qquad H = 1.
$$

Assume that the flow is periodic in the horizontal direction. The boundary schemes are applied at the upper and lower straight boundaries (see Figure 3). The lattice size is

$$
h = \frac{H}{Ny - 1 + 2\gamma} \tag{6.2}
$$

with γ the scaled distance. In order to demonstrate the stability and accuracy, we define the relative L^2-error as

$$
E_2 = \frac{\sqrt{\sum_x |u(x) - u_*(x)|^2}}{\sqrt{\sum_x |u_*(x)|^2}}, \tag{6.3}
$$

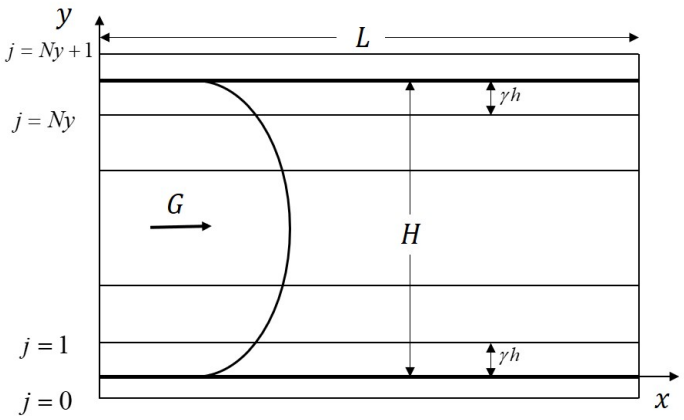

FIG. 3. *Configuration of the Poiseuille flow and the computational grid.*

where the summation is taken over the lattice nodes in the computational domain, u_* is the analytical solution (6.1), and u is the numerical solution of the vectorial LBM together with the boundary schemes (3.12) or (5.7). The convergence criterion is defined as

$$\frac{\sqrt{\sum_x |u(t_n, x) - u(t_n - 1000\Delta t, x)|^2}}{\sqrt{\sum_x |u(t_n, x)|^2}} < 10^{-10}. \tag{6.4}$$

Here $u(t_n, x)$ is the numerical solution at x and time $t_n = n\Delta t$ with Δt the time step size.

In this experiment, we set the scaled distance as $\gamma = 0.25,\ 0.5,\ 0.75$ and take different ν $(= 0.1,\ 0.01,\ 0.002)$ to test the accuracy and stability of the boundary schemes (3.12) and (5.7). Here we take $Ny = 10,\ 20,\ 40,\ 60,\ 80$, and the number of the meshes in the horizontal direction is set as $Nx = 2Ny$. The results based on the vector-type bounce-back scheme (3.12) are plotted in Figure 4. It is easy to see that the boundary scheme (3.12) is second-order accurate when the scaled distance $\gamma = 0.5$ and first-order accurate otherwise. The numerical results agree with the analysis in section 3. Figure 5 shows the numerical results of the LBM together with the boundary scheme (5.7). It is clear that all the cases have second-order accuracy, which validates the analysis in section 5.

6.2. Couette flow. In the Poiseuille flow, the boundaries are stationary so that the term containing the boundary velocity in the vector-type bounce-back scheme (3.12) and the boundary scheme (5.7) vanishes. To examine the schemes for nonzero velocity cases, we consider the Couette flow, which has the configuration of the Poiseuille flow. The problem has the analytical solution

$$u_1 = u(y) = 4U\left(1 - \frac{y}{H}\right)\frac{y}{H} + \frac{y}{H}U_b, \quad u_2 = 0. \tag{6.5}$$

Here the velocity at the top boundary is $U_b = 0.5U$ and other variables are the same as those of the Poiseuille flow. We apply the vector-type bounce-back scheme (3.12) and the boundary scheme (5.7) to the top and bottom wall boundary.

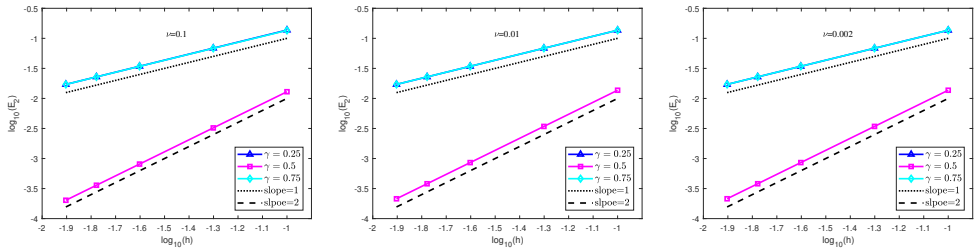

FIG. 4. *Convergence order of the vectorial LBM* (2.1) *together with the boundary scheme* (3.12) *for the Poiseuille flow. From left to right:* $\nu = 0.1,\ 0.01,\ 0.002$ *with* $\gamma = 0.25,\ 0.5,\ 0.75$.

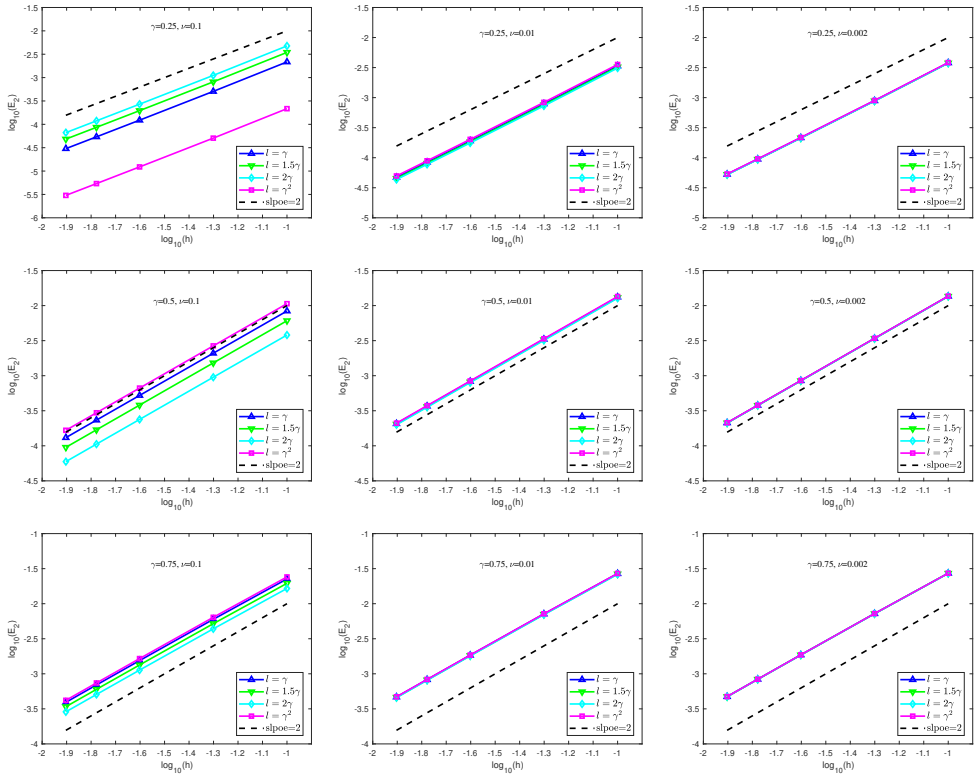

FIG. 5. *Convergence order of the vectorial LBM* (2.1) *together with the boundary scheme* (5.7) *for the Poiseuille flow. From top to bottom:* $\gamma = 0.25,\ 0.5,\ 0.75$ *with* $\nu = 0.1,\ 0.01,\ 0.002$ *and* $l = \gamma,\ 1.5\gamma,\ 2\gamma,\ \gamma^2$.

In this experiment, we still adopt the lattice size (6.2), relative L^2-error (6.3), and criterion convergence (6.4) and take $\gamma = 0.25,\ 0.5,\ 0.75$, $Ny = 10,\ 20,\ 40,\ 60,\ 80$, and $Nx = 2Ny$. The computational results in Figure 6 demonstrate that the vector-type bounce-back boundary scheme (3.12) has second-order accuracy only when $\gamma = 0.5$ and is first-order accurate otherwise. Figure 7 displays the numerical results of the vectorial LBM (2.1) together with the boundary scheme (5.7). The results illustrate that the boundary scheme (5.7) always has second-order accuracy. They both validate the analyses in sections 3 and 5.

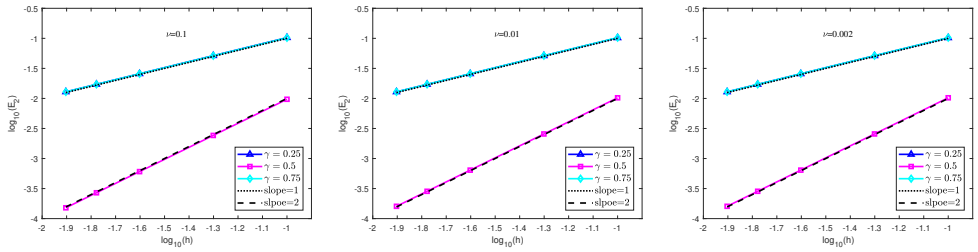

FIG. 6. *Convergence order of the vectorial LBM* (2.1) *together with the vector-type bounce-back boundary scheme* (3.12) *for the Couette flow. From left to right:* $\nu = 0.1,\ 0.01,\ 0.002$ *with* $\gamma = 0.25,\ 0.5,\ 0.75$.

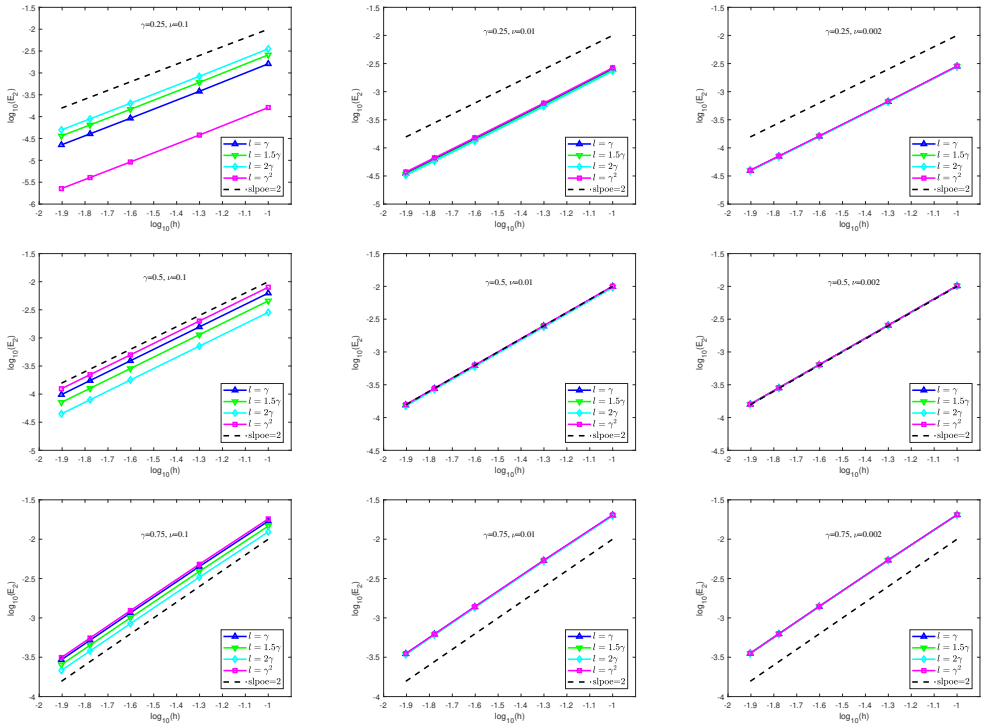

FIG. 7. *Convergence order of the vectorial LBM* (2.1) *together with the boundary scheme* (5.7) *for the Couette flow. From top to bottom:* $\gamma = 0.25,\ 0.5,\ 0.75$ *with* $\nu = 0.1,\ 0.01,\ 0.002$ *and* $l = \gamma,\ 1.5\gamma,\ 2\gamma,\ \gamma^2$.

6.3. Taylor–Green vortex flow in an irregular domain. To examine the applicability and utility of the boundary scheme (5.7) dealing with general curved boundaries, we consider the Taylor–Green vortex flow without external forces in an irregular domain as our third problem. The irregular domain is chosen as that in [18],

$$\Omega = \{(x, y)\ |P(x, y) \le 0\}$$

with

$$P(x, y) = x^4 - 12x^3 + 49x^2 - 81x + 2y^4 - 22y^3 + 84y^2 - 137y + 122$$

as shown in Figure 8.

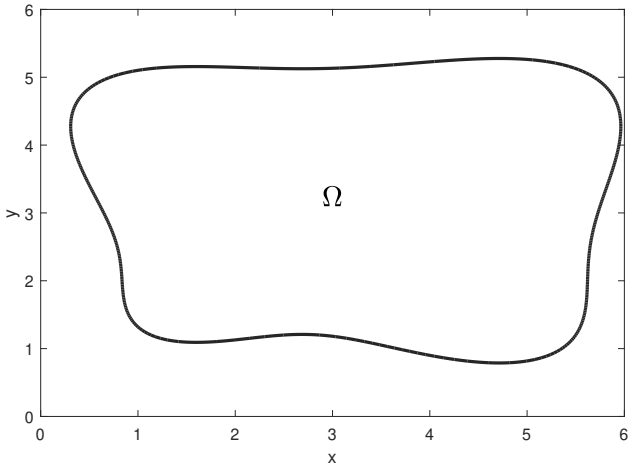

FIG. 8. *Illustration of the irregular domain. The fluid domain is that enclosed with the thick curve.*

This problem has analytic solutions

$$
\begin{aligned}
u(x,y,t) &= -U_0 \cos(k_1 x)\sin(k_2 y)\, e^{-\nu t\left(k_1^2+k_2^2\right)},\\
v(x,y,t) &= U_0 \frac{k_1}{k_2}\sin(k_1 x)\cos(k_2 y)\, e^{-\nu t\left(k_1^2+k_2^2\right)}
\end{aligned}
\tag{6.6}
$$

with pressure

$$
P(x,y,t) = -\frac{1}{4}U_0^2\left(\cos(2k_1 x) + \frac{k_1^2}{k_2^2}\cos(2k_2 y)\right) e^{-2\nu t\left(k_1^2+k_2^2\right)}, \tag{6.7}
$$

where k_1 and k_2 are the wave numbers in x and y directions, respectively. We choose $U_0 = 1.0, k_1 = k_2 = \pi/2$ and the maximal time for computation is set to $T = 1$. The relative L^2-error is defined as before:

$$
E_2 = \frac{\sqrt{\sum_x |u(T,x) - u_*(T,x)|^2}}{\sqrt{\sum_x |u_*(T,x)|^2}}, \tag{6.8}
$$

where the summation is taken over the lattice nodes in the computational domain at time T $(= 1)$, u_* is the analytical solution (6.6), and u is the numerical solution of the vectorial LBM (2.1) with the boundary scheme (5.7).

To examine the stability and accuracy of the boundary scheme, we take different spatial mesh sizes $h = 6/(192*1),\ 6/(192*2),\ 6/(192*3),\ 6/(192*4)$. The numerical results with different ν $(= 0.01,\ 0.002,\ 0.0002,\ 0.00002)$ are given in Figure 9. One can see that, for all the cases, the desired second-order accuracy is achieved. These results show the utility of our scheme (5.7) for unsteady problems with curved boundaries.

7. Conclusions and remarks. In this paper, we propose a vector-type bounce-back boundary scheme (3.12) for the LBM (2.1). By using the Maxwell iteration, we show that the new boundary scheme is second-order accurate when the boundary is

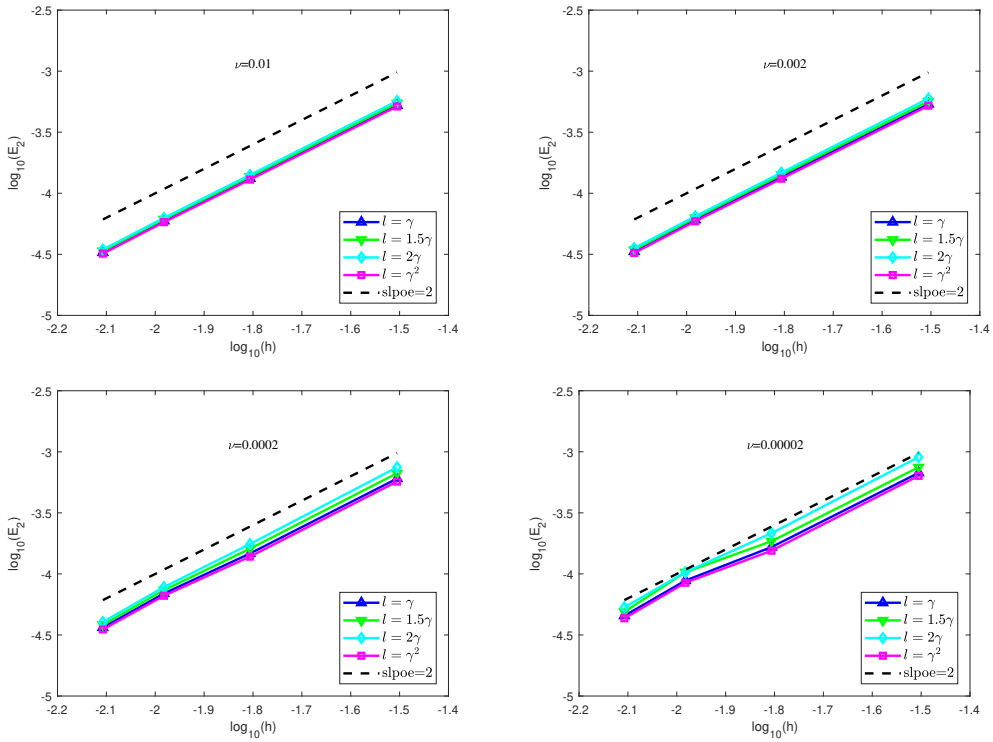

FIG. 9. *Convergence order of the vectorial LBM* (2.1) *together with the boundary scheme* (5.7) *for the Taylor–Green vortex flow in an irregular domain. From top to bottom and from left to right:* $\nu = 0.01,\ 0.002,\ 0.0002,\ 0.00002$ *with* $l = \gamma,\ 1.5\gamma,\ 2\gamma,\ \gamma^2$.

located at the middle of two neighboring lattice nodes. We also analyze the numerical stability of the new boundary scheme. Based on the vector-type boundary scheme, we derive a parameterized second-order boundary scheme (5.7) with accuracy independent of the location of boundary. Numerical experiments are conducted to validate the accuracy and stability of the boundary schemes with both straight and curved boundaries.

It should be noted that the construction of the vector-type bounce-back boundary scheme takes advantage of the vector nature of the distribution functions and seems novel. In addition, the stability analysis crucially utilizes the subtle structural property (4.9) of the vectorial LB model.

Finally, we point out that the construction ideas of the boundary schemes may be applied to the vectorial lattice Boltzmann schemes simulating strong nonlinear waves [12], the BGK-FVS schemes for the incompressible Navier–Stokes equations [7], and the "vector kinetic" numerical approach determining permeability tensors of porous media [19]. These are left for the future work.

REFERENCES

[1] C. K. AIDUN AND J. R. CLAUSEN, *Lattice-Boltzmann method for complex flows*, Annu. Rev. Fluid Mech., 42 (2010), pp. 439–472.

[2] S. ANSUMALI, I. V. KARLIN, AND H. C. ÖTTINGER, *Minimal entropic kinetic models for hydrodynamics*, Europhys. Lett., 63 (2003), pp. 798–804.

[3] P. Bhatnagar, E. Gross, and M. Krook, *A model for collision processes in gases.* I. *Small amplitude processes in charged and neutral one-component systems*, Phys. Rev., 94 (1954), 511.
[4] R. Bianchini, *Strong convergence of a vector-BGK model to the incompressible Navier-Stokes equations*, J. Math. Pures Appl., 12 (2019), pp. 133–158.
[5] R. Bianchini and R. Natalini, *Convergence of a vector BGK approximation for the incompressible Navier-Stokes equations*, Kinet. Relat. Models, 12 (2019), pp. 133–158.
[6] F. Bouchut, *Construction of BGK models with a family of kinetic entropies for a given system of conservation laws*, J. Stat. Phys., 95 (1999), pp. 113–170.
[7] F. Bouchut, Y. Jobic, R. Natalini, R. Occelli, and V. Pavan, *Second-order entropy satisfying BGK-FVS schemes for incompressible Navier-Stokes equations*, SMAI J. Comput. Math., 4 (2018), pp. 1–56.
[8] M. Bouzidi, M. Firdaouss, and P. Lallemand, *Momentum transfer of a Boltzmann-lattice fluid with boundaries*, Phys. Fluids, 13 (2001), pp. 3452–3459.
[9] S. Chen and G. Doolen, *Lattice Boltzmann method for fluid flows*, Annu. Rev. Fluid Mech., 30 (1998), pp. 329–364.
[10] P. J. Dellar, *Lattice kinetic schemes for magnetohydrodynamics*, J. Comput. Phys., 179 (2002), pp. 95–126.
[11] D. d'Humières, *Generalized lattice-Boltzmann equations*, in Rarefied Gas Dynamics: Theory and Simulations, B. D. Shizgal and D. P. Weaver, eds., Progr. Aeronaut. Astronaut. 159, 1992, pp. 450–458.
[12] F. Dubois, *Simulation of strong nonlinear waves with vectorial lattice Boltzmann schemes*, Internat. J. Modern Phys. C, 2014, 1441014.
[13] F. Dubois, P. Lallemand, and M. M. Tekitek, *On anti bounce back boundary condition for lattice Boltzmann schemes*, Comput. Math. Appl., 79 (2020), pp. 555–575.
[14] I. Ginzburg, *Generic boundary conditions for lattice Boltzmann models and their application to advection and anisotropic dispersion equations*, Adv. Water Res., 28 (2005), pp. 1196–1216.
[15] I. Ginzburg, F. Verhaeghe, and D. d'Humières, *Two-relaxation-time lattice Boltzmann scheme: About parametrization, velocity, pressure and mixed boundary conditions*, Commun. Comput. Phys., 3 (2008), pp. 427–478.
[16] Z. Guo, M. Firdaouss, and P. Lallemand, *An extrapolation method for boundary conditions in lattice Boltzmann method*, Phys. Fluids, 13 (2001), pp. 3452–3459.
[17] Z. Guo and C. Shu, *Lattice Boltzmann Method and Its Applications in Engineering*, World Scientific, Singapore, 2013.
[18] J. Huang and W.-A. Yong, *Boundary conditions of the lattice Boltzmann method for convection-diffusion equations*, J. Comput. Phys., 300 (2015), pp. 70–91.
[19] Y. Jobic, P. Kumar, F. Topin, and R. Occelli, *Determining permeability tensors of porous media: A novel "vector kinetic" numerical approach*, Int. J. Multiph. Flow, 110 (2019), pp. 198–217.
[20] M. Junk and Z. Yang, *Convergence of lattice Boltzmann methods for Stokes flows in periodic and bounded domains*, Comput. Math. Appl., 55 (2008), pp. 1481–1491.
[21] M. Junk and Z. Yang, *Convergence of lattice Boltzmann methods for Navier-Stokes flows in periodic and bounded domains*, Numer. Math., 112 (2009), pp. 65–87.
[22] M. Junk and W.-A. Yong, *Weighted L2-stability of the lattice Boltzmann method*, SIAM J. Numer. Anal., 47 (2009), pp. 1651–1665.
[23] I. Karlin, A. Ferrante, and H. C. Öttinger, *Perfect entropy functions of the lattice Boltzmann method*, Europhys. Lett., 47 (1999), pp. 182–188.
[24] I. Karlin, A. Gorban, S. Succi, and V. Boffi, *Maximum entropy principle for lattice kinetic equations*, Phys. Rev. Lett., 81 (1998).
[25] A. J. C. Ladd, *Numerical simulations of particulate suspensions via a discretized Boltzmann equation. Part* 1. *Theoretical foundation*, J. Fluid Mech., 271 (1994), pp. 285–309.
[26] A. J. C. Ladd, *Numerical simulations of particulate suspensions via a discretized Boltzmann equation. Part* 2. *Numerical results*, J. Fluid Mech., 271 (1994), pp. 311–339.
[27] S. Succi, *The Lattice Boltzmann Equation: For Fluid Dynamics and Beyond*, Oxford University Press, Oxford, 2001.
[28] S. Succi, R. Benzi, A. Calí, and M. Vergassola, *The Lattice Boltzmann Equation: Theory and Application*, Springer, New York, 1992, pp. 187–203.
[29] M. Zhang, W. Zhao, and P. Lin, *Lattice Boltzmann method for general convection-diffusion equations: MRT model and boundary schemes*, J. Comput. Phys., 389 (2019), pp. 147–163.
[30] J. Zhao, *Discrete-velocity vector-BGK models based numerical methods for the incompressible Navier-Stokes equations*, Commun. Comput. Phys., to appear, 2020.

[31] J. Zhao, Z. Zhang, and W.-A. Yong, *Approximation of the multi-dimensional incompressible Navier-Stokes equations by discrete-velocity vector-BGK models*, J. Math. Anal. Appl., 486 (2020), 123901.
[32] W. Zhao, J. Huang, and W.-A. Yong, *Boundary conditions for kinetic theory based models* I*: Lattice Boltzmann models*, Multiscale Model. Simul., 17 (2019), pp. 854–872.
[33] W. Zhao and W.-A. Yong, *Maxwell iteration for the lattice Boltzmann method with diffusive scaling*, Phys. Rev. E, 95 (2017), 033311.