Journal of Computational Physics 329 (2017) 1–15

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

Single-node second-order boundary schemes for the lattice Boltzmann method

Weifeng Zhao [a], Wen-An Yong [b,a,*]

[a] Beijing Computational Science Research Center, Beijing 100094, China
[b] Zhou Pei-Yuan Center for Applied Mathematics, Tsinghua University, Beijing 100084, China

ARTICLE INFO

Article history:
Received 19 June 2016
Received in revised form 10 September 2016
Accepted 23 October 2016
Available online 3 November 2016

Keywords:
Lattice Boltzmann equations
Boundary schemes
Dirichlet boundary conditions
Maxwell iteration

ABSTRACT

Based on our recently developed Maxwell iteration for the lattice Boltzmann method, we propose a class of single-node boundary schemes for Dirichlet boundary conditions of the Navier–Stokes equations. The schemes all have second-order accuracy for both straight and curved boundaries. The accuracy and stability of two specific schemes are examined through several numerical experiments. The results validate the second-order accuracy and show that a boundary scheme with a convex combination of distribution functions has better stability.

1. Introduction

The lattice Boltzmann equation (LBE) is an explicit time marching finite difference scheme of the continuous Boltzmann equation in space and time [1,2]. Its kinetic nature and simplicity make the LBE a popular mesoscopic method in the field of computational fluid dynamics (CFD) [3–5]. Unlike the conventional numerical methods in CFD which directly obtain macroscopic quantities such as velocity and pressure [6,7], the LBE provides distribution functions (DFs) at each time-space point and the macroscopic quantities are then obtained by low order velocity moments of the DFs. This gives rise to a fundamental challenge for the LBE in treating boundary conditions (BCs), because the mesoscopic DFs coming from the boundary are required at the fluid nodes next to the boundary (see Fig. 1) while the given BCs usually only involve the macroscopic variables, such as the Dirichlet BCs or pressure BCs. Thus, an important task associated with the LBE is to construct accurate and simple boundary schemes consistent with the macroscopic BCs and compatible with the LBE.

This paper focuses on the Dirichlet BC

$$\boldsymbol{u}(\boldsymbol{x},t)=\boldsymbol{\phi}(\boldsymbol{x},t),\quad \boldsymbol{x}\in\partial\Omega, \tag{1}$$

for the incompressible Navier–Stokes equations. Here $\boldsymbol{u}(\boldsymbol{x},t)$ is the macroscopic fluid velocity at position $\boldsymbol{x}$ and time t, $\boldsymbol{\phi}(\boldsymbol{x},t)$ is a given function of $\boldsymbol{x}$ and t, and the boundary $\partial\Omega$ is often curved (as illustrated in Fig. 1) in complex flows such as flows in porous media and two-phase flows. This kind of BCs is the most often encountered in applications and therefore various different boundary schemes have been proposed for it in the LBE literature [8–19]. Many of these boundary schemes involve a number of neighboring lattice nodes. For example, that proposed by Lallemand and Luo in [17] involves the three

* Corresponding author at: Zhou Pei-Yuan Center for Applied Mathematics, Tsinghua University, Beijing 100084, China.
E-mail addresses: zhaoweifeng@csrc.ac.cn (W. Zhao), wayong@tsinghua.edu.cn (W.-A. Yong).

http://dx.doi.org/10.1016/j.jcp.2016.10.049
0021-9991/

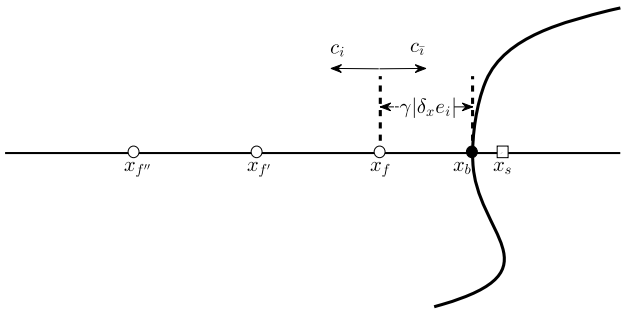

Fig. 1. Illustration of a curved boundary located between two lattice nodes arbitrarily. The thin solid straight line is the grid line and the thick curved line is the boundary. White circles (○) are the fluid nodes, the black circle (●) is the intersection of the boundary and the grid line, and the square box (□) is out of the fluid region.

nodes $\boldsymbol{x}_f$, $\boldsymbol{x}_{f'}$ and $\boldsymbol{x}_{f''}$ in Fig. 1. Other examples can be found in [13–16,19]. This class of boundary schemes usually employs interpolations of the DFs or fluid velocities at several neighboring nodes to increase the accuracy. As commented in [18], this class of schemes does not apply to the situation where there are no enough neighboring nodes (see Fig. 1 in [18] for example). Even if the single-node bounce-back scheme can be used to remedy this drawback, the accuracy of numerical results may degrade everywhere in the domain [18].

Another class of boundary schemes involves only the current lattice node and is free of interpolations. A typical example is the widely used bounce-back scheme [8,9], which only has first-order accuracy except for the case where the boundary locates at the middle of two neighboring nodes. Other single-node boundary schemes can be found in [10–12,18,20,21]. The schemes in [10–12] are of second-order accuracy but only for straight boundaries, while that proposed in [18] uses the DFs of all directions and needs to compute, at each boundary node $\boldsymbol{x}_f$, the inverse of a matrix with entries given by complicated formulas. Consequently, the one-point scheme [18] seems more complicated than the interpolation based schemes. In addition, those recently developed in [20,21] are for convection diffusion equations with complex boundary conditions and the idea can be extended to the Navier–Stokes equations. We remark that the derivation of the boundary schemes in [12,18,20] relies heavily on the Chapman–Enskog expansion or asymptotic analysis.

The aim of this work is to propose a class of single-node boundary schemes with second-order accuracy for the Dirichlet boundary condition Eq. (1) on curved boundaries. We start with a general linear combination of the pre-collision and equilibrium DFs, including the single-node schemes in [20] as a special case. Then with the recently developed Maxwell iteration for the LBE [22], we obtain three restrictions on the combination coefficients to have second-order accuracy. Based on the restrictions, two specific boundary schemes are presented. Moreover, the accuracy and stability of two specific schemes are analyzed through several numerical experiments. The results validate the second-order accuracy and show that the scheme with a convex combination of the DFs has better stability.

This paper is organized as follows. Section 2 introduces the multi-relaxation-time LBE and the two-relaxation-time (TRT) model with nine velocities on the two-dimensional square lattice (D2Q9 model). In Section 3, we use the Maxwell iteration to obtain an expansion of the DFs. Section 4 presents the main results of this work. A class of single-node boundary schemes with second-order accuracy is proposed and two specific schemes are constructed. The accuracy and stability of those boundary schemes are investigated with several numerical experiments in Section 5. Some conclusions and remarks are given in Section 6. The paper ends with an appendix containing details of the Maxwell iteration.

2. Two-relaxation-time lattice Boltzmann equations

Lattice Boltzmann equations (LBEs) can be directly derived from the kinetic equation with the coupling discretization of time and space [1,2]. Specifically, the discrete time is $\{t_n | n = 0, 1, 2, \ldots\}$ with time step δ_t and the coupled space consists of a finite and symmetric discrete set $\mathbb{V} := \{\boldsymbol{c}_i\} = \{-\boldsymbol{c}_i\}$ and a d-dimensional lattice space $\mathbb{Z}^d$ with a lattice size δ_x. The lattice $\mathbb{Z}^d$, the discrete velocity set $\mathbb{V}$, and discrete time are coupled as the following:

$$\boldsymbol{x}_j + \boldsymbol{c}_i\delta_t \in \mathbb{Z}^d \quad \forall \boldsymbol{x}_j \in \mathbb{Z}^d \text{ and } \boldsymbol{c}_i \in \mathbb{V}, \tag{2}$$

such that the particle transport term in the original Boltzmann equation, $\partial_t f + \boldsymbol{\xi} \cdot \nabla f$, is modeled by hopping from one lattice node $\boldsymbol{x}_j$ to another $\boldsymbol{x}_j + \boldsymbol{c}_i\delta_t$.

The LBEs can be concisely written in a vector form:

$$\mathbf{f}(\boldsymbol{x}_j + \boldsymbol{c}\delta_t, t_n + \delta_t) - \mathbf{f}(\boldsymbol{x}_j, t_n) = \boldsymbol{\Omega}(\boldsymbol{x}_j, t_n), \tag{3}$$

where $\mathbf{f}$ and $\boldsymbol{\Omega}$ are Q-dimensional vectors representing the discrete-velocity particle distribution and the collision term, respectively. Specifically, they are defined as

$$\mathbf{f}(\boldsymbol{x}_j + \boldsymbol{c}\delta_t,\, t_n + \delta_t) := (f_0(\boldsymbol{x}_j,\, t_n + \delta_t)),\; f_1(\boldsymbol{x}_j + \boldsymbol{c}_1\delta_t,\, t_n + \delta_t),\; \ldots,\; f_q(\boldsymbol{x}_j + \boldsymbol{c}_q\delta_t,\, t_n + \delta_t))^\dagger,$$
$$\mathbf{f}(\boldsymbol{x}_j,\, t_n) := (f_0(\boldsymbol{x}_j,\, t_n),\; f_1(\boldsymbol{x}_j,\, t_n),\; \ldots,\; f_q(\boldsymbol{x}_j,\, t_n))^\dagger,$$
$$\boldsymbol{\Omega}(\boldsymbol{x}_j,\, t_n) := (\Omega_0(\boldsymbol{x}_j,\, t_n),\; \Omega_1(\boldsymbol{x}_j,\, t_n),\; \ldots,\; \Omega_q(\boldsymbol{x}_j,\, t_n))^\dagger,$$

where $q := (Q-1)$ is the number of non-zero velocities and † denotes the transpose; and it is always assumed that $\boldsymbol{c}_0 := \mathbf{0}$.

It is natural to execute the collision process in the moment space, because relaxation of moments is directly related to dissipation processes in hydrodynamic systems. To cast the collision in terms of relaxation process of moments is the essence of the LBE with multiple-relaxation-time (MRT) proposed by d'Humières [23]. In the MRT model, the collision term in the LBE (3) can be written in general as the following:

$$\boldsymbol{\Omega} = -\mathsf{M}^{-1}\cdot\mathsf{S}\cdot\left[\mathbf{m} - \mathbf{m}^{(0)}\right], \tag{4}$$

where $\mathbf{m}$ and $\mathbf{m}^{(0)}$ are Q-dimensional vectors of moments and their equilibria, respectively; S is a diagonal matrix with non-negative diagonal elements (relaxation rates); and M is the transformation matrix which maps the distribution functions $\{f_i\}$ to the moments $\{m_i\}$, *i.e.*,

$$\mathbf{m} := \mathsf{M}\cdot\mathbf{f}. \tag{5}$$

To be concrete, we will use the nine-velocity model on two-dimensional square lattice, *i.e.*, the D2Q9 model, as a specific model in what follows. For the D2Q9 model, the discrete velocities are: $\boldsymbol{c}_i = \boldsymbol{e}_i c$, $\boldsymbol{e}_0 = (0,\,0)$, $\boldsymbol{e}_1 = -\boldsymbol{e}_3 = (1,\,0)$, $\boldsymbol{e}_2 = -\boldsymbol{e}_4 = (0,\,1)$, $\boldsymbol{e}_5 = -\boldsymbol{e}_7 = (1,\,1)$, and $\boldsymbol{e}_6 = -\boldsymbol{e}_8 = (-1,\,1)$. Here $c := \delta_x/\delta_t$, and the nine moments are

$$\mathbf{m} := \left(\rho,\, e,\, \varepsilon,\, j_x,\, q_x,\, j_y,\, q_y,\, p_{xx},\, p_{xy}\right)^\dagger, \tag{6}$$

where ρ is the zeroth-order moment and the mass density; e is the second-order moment related to energy, ε is the fourth-order moment related to the energy square; j_x and j_y are the first-order moments corresponding to the x- and y-component of momentum, respectively; q_x and q_y are the third-order moments corresponding to the x- and y-component of energy flux, respectively; and p_{xx} and p_{xy} are the second-order moments related to the diagonal and off-diagonal component of the stress tensor, respectively [24]. While the moments are somewhat uniquely determined, their ordering in Eq. (6) is arbitrary. With the ordering specified by Eq. (6), the transformation matrix M is given as [24]:

$$\mathsf{M} = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ -4 & -1 & -1 & -1 & -1 & 2 & 2 & 2 & 2 \\ 4 & -2 & -2 & -2 & -2 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & -1 & 0 & 1 & -1 & -1 & 1 \\ 0 & -2 & 0 & 2 & 0 & 1 & -1 & -1 & 1 \\ 0 & 0 & 1 & 0 & -1 & 1 & 1 & -1 & -1 \\ 0 & 0 & -2 & 0 & 2 & 1 & 1 & -1 & -1 \\ 0 & 1 & -1 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & 1 & -1 \end{pmatrix}. \tag{7}$$

Correspondingly, the diagonal matrix S composed of the relaxation rates $\{s_i\}$ is given in the following form:

$$\mathsf{S} := \mathrm{diag}\,(s_0,\, s_1,\, s_2,\, s_3,\, s_4,\, s_5,\, s_6,\, s_7,\, s_8) \tag{8}$$

with $0 \le s_i \le 2$ for $i = 1, 2, 4, 6, 7, 8$ [24,25].

If we set the relaxation rates to s_ν for the even-order nonconserved moments (*i.e.*, e, ε, p_{xx} and p_{xy}) and to

$$s_q = 8\frac{2 - s_\nu}{8 - s_\nu} \tag{9}$$

for the odd-order ones (*i.e.*, q_x and q_y) [26–28], then the MRT model becomes a two-relaxation-time (TRT) model. It was found in [27,28] that the relation (9) enforces the Dirichlet boundary conditions in $\boldsymbol{u}$ exactly and minimizes the slip velocity for the Poiseuille flow. With the TRT collision term, the LBE can be written in the distribution function space as the following concise form [26–28],

$$\begin{aligned} & f_i(\boldsymbol{x} + \boldsymbol{c}_i\delta_t, t + \delta_t) - f_i(\boldsymbol{x}, t) \\ =& -\frac{s_\nu}{2}\left[f_i(\boldsymbol{x}, t) + f_{\bar{i}}(\boldsymbol{x}, t) - \left(f_i^{(eq)}(\boldsymbol{x}, t) + f_{\bar{i}}^{(eq)}(\boldsymbol{x}, t)\right)\right] \\ & -\frac{s_q}{2}\left[f_i(\boldsymbol{x}, t) - f_{\bar{i}}(\boldsymbol{x}, t) - \left(f_i^{(eq)}(\boldsymbol{x}, t) - f_{\bar{i}}^{(eq)}(\boldsymbol{x}, t)\right)\right], \end{aligned} \tag{10}$$

where $f_{\bar{i}}$ is the distribution function corresponding to the velocity $\boldsymbol{c}_{\bar{i}} = -\boldsymbol{c}_i$. For the equilibria $f_i^{(eq)}$, we choose the following incompressible model [29]

$$f_i^{(eq)} = \omega_i \left\{ \rho + \rho_0 \left[\frac{\boldsymbol{c}_i \cdot \boldsymbol{u}}{c_s^2} + \frac{(\boldsymbol{c}_i \cdot \boldsymbol{u})^2}{2c_s^4} - \frac{u^2}{2c_s^2} \right] \right\}. \tag{11}$$

Here $\omega_0 = 4/9$, $\omega_{1,2,3,4} = 1/9$ and $\omega_{5,6,7,8} = 1/36$; $c_s = c/\sqrt{3}$ is the sound speed; ρ_0 is the mean density; ρ and $\boldsymbol{u}$ are the flow density and velocity given by

$$\rho = \sum_i f_i = \sum_i f_i^{(eq)}, \tag{12a}$$

$$\rho_0 \boldsymbol{u} = \sum_i \boldsymbol{c}_i f_i = \sum_i \boldsymbol{c}_i f_i^{(eq)}. \tag{12b}$$

3. Maxwell iteration for the TRT model

In this section, we derive an asymptotic expression of the LB solution f_i in terms of the equilibrium distribution $f_i^{(eq)}$ by using the Maxwell iteration recently developed in [22] for the LB equations. To begin with, we introduce the diffusive scaling

$$h := \delta_x \quad \text{and} \quad \delta_t = \eta h^2 \tag{13}$$

with η an adjustable parameter. This scaling means

$$c = \frac{\delta_x}{\delta_t} = \frac{1}{\eta h}, \qquad c_s = \frac{1}{\sqrt{3}\eta h},$$

and

$$f_i^{(eq)} = \omega_i \left\{ \rho + \rho_0 \left[3\eta h \boldsymbol{e}_i \cdot \boldsymbol{u} + \eta^2 h^2 \frac{9(\boldsymbol{e}_i \cdot \boldsymbol{u})^2 - 3u^2}{2} \right] \right\}. \tag{14}$$

Under the diffusive scaling above, the left-hand side of the LBE (10) can be expanded as

$$f_i(\boldsymbol{x} + \boldsymbol{e}_i h, t + \eta h^2) - f_i(\boldsymbol{x}, t) \sim \sum_{s=1}^{\infty} h^s D_{i,s} f_i(\boldsymbol{x}, t),$$

where the differential operator $D_{i,s}$ is defined as

$$D_{i,s} = \sum_{m+2n=s} \frac{(\boldsymbol{e}_i \cdot \nabla)^m (\eta \partial_t)^n}{m!n!}$$

with m and n two nonnegative integers. Thus the LBE (10) can be written as

$$\begin{aligned} &\sum_{s=1}^{\infty} h^s D_{i,s} f_i(\boldsymbol{x}, t) \\ &\sim -s_\nu \left[\frac{1}{2}\left(f_i(\boldsymbol{x}, t) + f_{\bar{i}}(\boldsymbol{x}, t)\right) - \frac{1}{2}\left(f_i^{(eq)}(\boldsymbol{x}, t) + f_{\bar{i}}^{(eq)}(\boldsymbol{x}, t)\right) \right] \\ &\quad - s_q \left[\frac{1}{2}\left(f_i(\boldsymbol{x}, t) - f_{\bar{i}}(\boldsymbol{x}, t)\right) - \frac{1}{2}\left(f_i^{(eq)}(\boldsymbol{x}, t) - f_{\bar{i}}^{(eq)}(\boldsymbol{x}, t)\right) \right]. \end{aligned} \tag{15}$$

From this we use the Maxwell iteration in Appendix to derive

$$\begin{aligned} f_i = f_i^{(eq)} &- \frac{b}{1-a^2} h (\boldsymbol{e}_i \cdot \nabla)(f_i^{(eq)} + a f_{\bar{i}}^{(eq)}) - \frac{1}{1-a^2} h^2 \left[(\frac{1}{2}b - a^2 b^2)(\boldsymbol{e}_i \cdot \nabla)^2 + b\eta\partial_t \right] f_i^{(eq)} \\ &+ \frac{ab}{1-a^2} h^2 \left[\frac{1}{2}(\boldsymbol{e}_i \cdot \nabla)^2 + \eta\partial_t \right] f_{\bar{i}}^{(eq)} + O(h^3), \end{aligned} \tag{16}$$

where

$$a = \frac{s_\nu - s_q}{s_\nu + s_q} \quad \text{and} \quad b = \frac{2}{s_\nu + s_q}.$$

Taking the first-order velocity moment of Eq. (16) and using Eq. (14) gives

$$\rho_0 \boldsymbol{u} = \rho_0 \boldsymbol{u} - \sum_i \left[\boldsymbol{c}_i \frac{b}{1-a^2} h \left(\boldsymbol{e}_i \cdot \nabla\right) f_i^{(eq)} \right] - \sum_i \left[\boldsymbol{c}_i \frac{ab}{1-a^2} h \left(\boldsymbol{e}_i \cdot \nabla\right) f_{\bar{i}}^{(eq)} \right] + O(h^2). \tag{17}$$

Recalling that

$$\sum_i \boldsymbol{c}_i \boldsymbol{c}_i f_i^{(eq)} = \rho_0 \boldsymbol{u}\boldsymbol{u} + \rho c_s^2 \mathrm{I}$$

with I the 2×2 identity matrix, it follows from Eq. (17) that

$$\begin{aligned} O(h^2) &= \frac{b}{1-a^2} \eta h^2 \sum_i \boldsymbol{c}_i \left(\boldsymbol{c}_i \cdot \nabla\right) f_i^{(eq)} + \frac{ab}{1-a^2} \eta h^2 \sum_{\bar{i}} \boldsymbol{c}_{\bar{i}} \left(\boldsymbol{c}_{\bar{i}} \cdot \nabla\right) f_{\bar{i}}^{(eq)} \\ &= \frac{b}{1-a} \eta h^2 \nabla \cdot \left(\rho_0 \boldsymbol{u}\boldsymbol{u} + \rho c_s^2 \mathrm{I}\right) \\ &= \frac{b}{1-a} \eta c_s^2 h^2 \nabla \rho + O(h^2) \\ &= \frac{b}{3\eta(1-a)} \nabla \rho + O(h^2), \end{aligned}$$

which gives

$$\nabla \rho = O(h^2). \tag{18}$$

On the other hand, we may expand the macroscopic density as

$$\rho = \sum_{k \geq 0} h^k \rho^{(k)}$$

and take the leading term $\rho^{(0)} = \rho_0$ according to the asymptotic analysis in [30–32]. Then we have

$$\partial_t \rho = O(h). \tag{19}$$

Using the relations (14), (18) and (19), we can simplify the expansion (16) as

$$f_i = f_i^{(eq)} - 3\eta h^2 \tau \left(\boldsymbol{e}_i \cdot \nabla\right) \left(\omega_i \rho_0 \boldsymbol{e}_i \cdot \boldsymbol{u}\right) + O(h^3), \tag{20}$$

where

$$\tau := \frac{b}{1+a} = \frac{1}{s_\nu}. \tag{21}$$

Relations (18)–(20) will be used to construct our boundary schemes in the next section.

We conclude this section by mentioning that higher-order expansions of f_i can also be obtained from the Maxwell iteration and can be used to obtain the Euler or Navier–Stokes equations [22]. We will not do it here, since the relations (18)–(20) will be shown to be enough in deriving second-order boundary schemes.

4. Single-node schemes for Dirichlet boundary conditions

In this section, we construct a class of single-node second-order boundary schemes for Dirichlet boundary conditions (1). Inspired by the classical bounce-back scheme [8,9], our boundary schemes are of the following simple form:

$$f_i(\boldsymbol{x}_f, t+\delta_t) = a_1 f_{\bar{i}}(\boldsymbol{x}_f, t) + a_2 f_i(\boldsymbol{x}_f, t) + a_3 f_i^{(eq)}(\boldsymbol{x}_f, t) + a_4 f_{\bar{i}}^{(eq)}(\boldsymbol{x}_f, t) + a_5 \omega_i \rho_0 \frac{\mathbf{c}_i \cdot \boldsymbol{\phi}}{c_s^2} \tag{22}$$

with five coefficients $a_k (k = 1, 2, \ldots, 5)$ to be determined. In (22), $\boldsymbol{x}_f$ is the interior lattice node next to the boundary point $\boldsymbol{x}_b$ (see Fig. 1), and $\boldsymbol{\phi} := \boldsymbol{\phi}(\boldsymbol{x}_b, t)$ is the given boundary velocity. About such boundary schemes, we make the following remark.

Remark 1. The general form (22) contains the classical bounce-back scheme and the recently proposed single-node boundary schemes in [20] for convection–diffusion equations. But it is different substantially from those in [20], because the equilibrium DFs appear explicitly in (22). It is also different from the one-point scheme proposed in [18], for it involves the i-th and $\bar{i}$-th DFs only while the latter uses the DFs of all directions. Moreover, the one-point scheme needs to compute, at each boundary node $\boldsymbol{x}_f$, the inverse of a matrix with entries given by complicated formulas.

To determine the coefficients, we refer to Fig. 1 and introduce the scaled distance between $\boldsymbol{x}_f$ and $\boldsymbol{x}_b$ as

$$\gamma = \frac{|\boldsymbol{x}_b - \boldsymbol{x}_f|}{|\boldsymbol{x}_s - \boldsymbol{x}_f|} = \frac{|\boldsymbol{x}_b - \boldsymbol{x}_f|}{|h\boldsymbol{e}_i|}. \tag{23}$$

Then we have

$$\boldsymbol{x}_f = \boldsymbol{x}_b + \gamma h \boldsymbol{e}_i,$$

where $\boldsymbol{e}_i = \eta h \boldsymbol{c}_i$ points to $\boldsymbol{x}_f$ from $\boldsymbol{x}_b$.

Next we expand each term in Eq. (22) at the boundary point $(\boldsymbol{x}_b, t)$. For $f_i(\boldsymbol{x}_f, t+\delta_t)$, we use Eq. (20) to expand it at $(\boldsymbol{x}_b, t)$ as

$$\begin{aligned}
f_i(\boldsymbol{x}_f, t+\delta_t) =& f_i(\boldsymbol{x}_b + \gamma h \boldsymbol{e}_i, t + \eta h^2) \\
=& f_i(\boldsymbol{x}_b, t) + \eta h^2 \partial_t f_i(\boldsymbol{x}_b, t) + h\gamma \boldsymbol{e}_i \cdot \nabla f_i(\boldsymbol{x}_b, t) + \frac{1}{2} h^2 \gamma^2 (\boldsymbol{e}_i \cdot \nabla)^2 f_i(\boldsymbol{x}_b, t) + O(h^3) \\
=& f_i^{(eq)}(\boldsymbol{x}_b, t) - 3\eta h^2 \tau (\boldsymbol{e}_i \cdot \nabla)(\omega_i \rho_0 \boldsymbol{e}_i \cdot \boldsymbol{u})(\boldsymbol{x}_b, t) + \eta h^2 \partial_t f_i^{(eq)}(\boldsymbol{x}_b, t) \\
&+ h\gamma (\boldsymbol{e}_i \cdot \nabla) \left[f_i^{(eq)}(\boldsymbol{x}_b, t) - 3\eta h^2 \tau (\boldsymbol{e}_i \cdot \nabla)(\omega_i \rho_0 \boldsymbol{e}_i \cdot \boldsymbol{u})(\boldsymbol{x}_b, t) \right] \\
&+ \frac{1}{2} h^2 \gamma^2 (\boldsymbol{e}_i \cdot \nabla)^2 f_i^{(eq)}(\boldsymbol{x}_b, t) + O(h^3).
\end{aligned}$$

With Eqs. (18) and (19), this can be simplified as

$$\begin{aligned}
f_i(\boldsymbol{x}_f, t+\delta_t) =& f_i^{(eq)}(\boldsymbol{x}_b, t) - 3\eta h^2 \tau (\boldsymbol{e}_i \cdot \nabla)(\omega_i \rho_0 \boldsymbol{e}_i \cdot \boldsymbol{u})(\boldsymbol{x}_b, t) \\
&+ 3\eta h^2 \gamma (\boldsymbol{e}_i \cdot \nabla)(\omega_i \rho_0 \boldsymbol{e}_i \cdot \boldsymbol{u})(\boldsymbol{x}_b, t) + O(h^3) \\
=& f_i^{(eq)}(\boldsymbol{x}_b, t) - 3\eta h^2 (\tau - \gamma)(\boldsymbol{e}_i \cdot \nabla)(\omega_i \rho_0 \boldsymbol{e}_i \cdot \boldsymbol{u})(\boldsymbol{x}_b, t) + O(h^3).
\end{aligned} \tag{24}$$

Similarly, we have

$$\begin{aligned}
f_{\bar{i}}(\boldsymbol{x}_f, t) =& f_{\bar{i}}(\boldsymbol{x}_b - \gamma h \boldsymbol{e}_{\bar{i}}, t) \\
=& f_{\bar{i}}(\boldsymbol{x}_b, t) - \gamma h (\boldsymbol{e}_{\bar{i}} \cdot \nabla) f_{\bar{i}}(\boldsymbol{x}_b, t) + \frac{1}{2} \gamma^2 h^2 (\boldsymbol{e}_{\bar{i}} \cdot \nabla)^2 f_{\bar{i}}(\boldsymbol{x}_b, t) + O(h^3) \\
=& f_{\bar{i}}^{(eq)}(\boldsymbol{x}_b, t) - 3\eta h^2 \tau (\boldsymbol{e}_{\bar{i}} \cdot \nabla)(\omega_i \rho_0 \boldsymbol{e}_{\bar{i}} \cdot \boldsymbol{u})(\boldsymbol{x}_b, t) - 3\eta\gamma h^2 (\boldsymbol{e}_{\bar{i}} \cdot \nabla)(\omega_i \rho_0 \boldsymbol{e}_{\bar{i}} \cdot \boldsymbol{u})(\boldsymbol{x}_b, t) + O(h^3) \\
=& f_{\bar{i}}^{(eq)}(\boldsymbol{x}_b, t) - 3\eta h^2 (\tau + \gamma)(\boldsymbol{e}_i \cdot \nabla)(\omega_i \rho_0 \boldsymbol{e}_i \cdot \boldsymbol{u})(\boldsymbol{x}_b, t) + O(h^3),
\end{aligned} \tag{25}$$

$$\begin{aligned}
f_i(\boldsymbol{x}_f, t) =& f_i(\boldsymbol{x}_b + \gamma h \boldsymbol{e}_i, t) \\
=& f_i^{(eq)}(\boldsymbol{x}_b, t) - 3\eta h^2 (\tau - \gamma)(\boldsymbol{e}_i \cdot \nabla)(\omega_i \rho_0 \boldsymbol{e}_i \cdot \boldsymbol{u})(\boldsymbol{x}_b, t) + O(h^3),
\end{aligned} \tag{26}$$

$$\begin{aligned}
f_i^{(eq)}(\boldsymbol{x}_f, t) =& f_i^{(eq)}(\boldsymbol{x}_b + \gamma h \boldsymbol{e}_i, t) \\
=& f_i^{(eq)}(\boldsymbol{x}_b, t) + 3\eta\gamma h^2 (\boldsymbol{e}_i \cdot \nabla)(\omega_i \rho_0 \boldsymbol{e}_i \cdot \boldsymbol{u})(\boldsymbol{x}_b, t) + O(h^3),
\end{aligned} \tag{27}$$

and

$$f_{\bar{i}}^{(eq)}(\boldsymbol{x}_f, t) = f_{\bar{i}}^{(eq)}(\boldsymbol{x}_b - \gamma \boldsymbol{e}_{\bar{i}} h, t) = f_{\bar{i}}^{(eq)}(\boldsymbol{x}_b, t) - 3\eta\gamma h^2 (\boldsymbol{e}_i \cdot \nabla)(\omega_i \rho_0 \boldsymbol{e}_i \cdot \boldsymbol{u})(\boldsymbol{x}_b, t) + O(h^3). \tag{28}$$

With these expansions, we consider

$$
\begin{aligned}
R_i &:= f_i(\boldsymbol{x}_f, t+\delta_t) - a_1 f_{\bar{i}}(\boldsymbol{x}_f, t) - a_2 f_i(\boldsymbol{x}_f, t) - a_3 f_i^{(eq)}(\boldsymbol{x}_f, t) - a_4 f_{\bar{i}}^{(eq)}(\boldsymbol{x}_f, t) - a_5 \omega_i \rho_0 \frac{\mathbf{c}_i \cdot \boldsymbol{\phi}}{c_s^2} \\
&= f_i^{(eq)}(\boldsymbol{x}_b, t) - 3\eta h^2(\tau-\gamma)\,(\boldsymbol{e}_i\cdot\nabla)\,(\omega_i\rho_0\boldsymbol{e}_i\cdot\boldsymbol{u})\,(\boldsymbol{x}_b,t) \\
&\quad - a_1\left[f_{\bar{i}}^{(eq)}(\boldsymbol{x}_b, t) - 3\eta h^2(\tau+\gamma)\,(\boldsymbol{e}_i\cdot\nabla)\,(\omega_i\rho_0\boldsymbol{e}_i\cdot\boldsymbol{u})\,(\boldsymbol{x}_b,t)\right] \\
&\quad - a_2\left[f_i^{(eq)}(\boldsymbol{x}_b, t) - 3\eta h^2(\tau-\gamma)\,(\boldsymbol{e}_i\cdot\nabla)\,(\omega_i\rho_0\boldsymbol{e}_i\cdot\boldsymbol{u})\,(\boldsymbol{x}_b,t)\right] \\
&\quad - a_3\left[f_i^{(eq)}(\boldsymbol{x}_b, t) + 3\eta\gamma h^2\,(\boldsymbol{e}_i\cdot\nabla)\,(\omega_i\rho_0\boldsymbol{e}_i\cdot\boldsymbol{u})\,(\boldsymbol{x}_b,t)\right] \\
&\quad - a_4\left[f_{\bar{i}}^{(eq)}(\boldsymbol{x}_b, t) - 3\eta\gamma h^2\,(\boldsymbol{e}_i\cdot\nabla)\,(\omega_i\rho_0\boldsymbol{e}_i\cdot\boldsymbol{u})\,(\boldsymbol{x}_b,t)\right] - 3\eta h a_5\omega_i\rho_0\mathbf{e}_i\cdot\boldsymbol{\phi} + O(h^3), \\
&= (1-a_2-a_3) f_i^{(eq)}(\boldsymbol{x}_b, t) - (a_1+a_4) f_{\bar{i}}^{(eq)}(\boldsymbol{x}_b, t) \\
&\quad + 3\eta h^2\left[(\gamma-\tau) + a_1(\tau+\gamma) - a_2(\gamma-\tau) - a_3\gamma + a_4\gamma\right] \\
&\quad\quad \times (\boldsymbol{e}_i\cdot\nabla)\,(\omega_i\rho_0\boldsymbol{e}_i\cdot\boldsymbol{u})\,(\boldsymbol{x}_b,t) - 3\eta h a_5\omega_i\rho_0\mathbf{e}_i\cdot\boldsymbol{\phi} + O(h^3) \\
&= (1-a_1-a_2-a_3-a_4)\omega_i\left\{\rho(\boldsymbol{x}_b,t) + \eta^2h^2\rho_0\left[\frac{9\,(\boldsymbol{e}_i\cdot\boldsymbol{u}(\boldsymbol{x}_b,t))^2}{2} - \frac{3u^2(\boldsymbol{x}_b,t)}{2}\right]\right\} \\
&\quad + 3\eta h(1+a_1-a_2-a_3+a_4)\omega_i\rho_0\boldsymbol{e}_i\cdot\boldsymbol{u}(\boldsymbol{x}_b,t) - 3\eta h a_5\omega_i\rho_0\boldsymbol{e}_i\cdot\boldsymbol{\phi} \\
&\quad + 3\eta h^2\left[(\gamma-\tau) + a_1(\tau+\gamma) - a_2(\gamma-\tau) - a_3\gamma + a_4\gamma\right](\boldsymbol{e}_i\cdot\nabla)\,(\omega_i\rho_0\boldsymbol{e}_i\cdot\boldsymbol{u})\,(\boldsymbol{x}_b,t) + O(h^3).
\end{aligned}
\tag{29}
$$

From the first-order term $O(h)$ in R_i, we can see that, to be consistent with the Dirichlet BC Eq. (1), it is required that

$$1 + a_1 - a_2 - a_3 + a_4 = a_5.$$

Furthermore, we set the coefficients of orders $O(1)$ and $O(h^2)$ in R_i to be zero to have second-order accuracy. In this way, we obtain the following constraints for the coefficients a_k:

$$1 - a_1 - a_2 - a_3 - a_4 = 0, \tag{30a}$$

$$1 + a_1 - a_2 - a_3 + a_4 - a_5 = 0, \tag{30b}$$

$$\tau(1 - a_1 - a_2) - \gamma(1 + a_1 - a_2 - a_3 + a_4) = 0. \tag{30c}$$

These are our basic equations to construct second-order single-node schemes of the form (22).

Before proceeding, we make the following remarks on the above constraints.

Remark 2. (1). The first two constraints in Eq. (30) ensure first-order accuracy and, together with the third one, second-order accuracy.

(2). The classical bounce-back scheme [8,9]

$$f_i(\boldsymbol{x}_f, t+\delta_t) = f_{\bar{i}}^*(\boldsymbol{x}_f, t) + 2\omega_i\rho_0\frac{\mathbf{c}_i\cdot\boldsymbol{\phi}}{c_s^2}$$

is a simple example of the first-order schemes for $\gamma \neq 1/2$. Indeed, because the post-collision distribution is

$$
\begin{aligned}
f_{\bar{i}}^*(\boldsymbol{x}_f, t) &:= f_{\bar{i}}(\boldsymbol{x}_f, t) - s_\nu\left[\frac{1}{2}\left(f_i(\boldsymbol{x}_f,t) + f_{\bar{i}}(\boldsymbol{x}_f,t)\right) - \frac{1}{2}\left(f_i^{(eq)}(\boldsymbol{x}_f,t) + f_{\bar{i}}^{(eq)}(\boldsymbol{x}_f,t)\right)\right] \\
&\quad - s_q\left[\frac{1}{2}\left(f_{\bar{i}}(\boldsymbol{x}_f,t) - f_i(\boldsymbol{x}_f,t)\right) - \frac{1}{2}\left(f_{\bar{i}}^{(eq)}(\boldsymbol{x}_f,t) - f_i^{(eq)}(\boldsymbol{x}_f,t)\right)\right] \\
&= (1 - \frac{1}{2}s_\nu - \frac{1}{2}s_q) f_{\bar{i}}(\boldsymbol{x}_f,t) - (\frac{1}{2}s_\nu - \frac{1}{2}s_q) f_i(\boldsymbol{x}_f,t) + (\frac{1}{2}s_\nu - \frac{1}{2}s_q) f_i^{(eq)}(\boldsymbol{x}_f,t) \\
&\quad + (\frac{1}{2}s_\nu + \frac{1}{2}s_q) f_{\bar{i}}^{(eq)}(\boldsymbol{x}_f,t),
\end{aligned}
\tag{31}
$$

the classical bounce-back scheme can be written as (22) with

$$a_1 = 1 - a_4, \quad a_2 = -a_3, \quad a_3 = \frac{s_\nu - s_q}{2}, \quad a_4 = \frac{s_\nu + s_q}{2}, \quad a_5 = 2. \tag{32}$$

Obviously, they satisfy the first two equations in Eq. (30), while the third equation is satisfied only if $\gamma = 1/2$.

In order to obtain specific second-order schemes, we turn to determine the coefficients a_k from Eqs. (30), which contain five parameters with only three equations. Thus, we can construct a class of schemes with two adjustable parameters. Precisely, we can express a_2, a_3 and a_5 as

$$a_2 = 1 - a_1 - \frac{\gamma}{\tau}a_5, \quad a_3 = \frac{\gamma}{\tau}a_5 - a_4, \quad a_5 = 2(a_1 + a_4) \tag{33}$$

with a_1 and a_4 free. Once a_1 and a_4 are fixed, we obtain a second-order scheme (22). In principle, a_1 and a_4 can be specified arbitrarily.

Here we construct two schemes. The first one is motivated by the bounce-back scheme and we choose a_1 and a_4 similar to those in Eq. (32):

$$a_1 = 1 - a_4, \qquad a_4 = \gamma(s_\nu + s_q). \tag{34}$$

With this choice, we obtain the following second-order scheme

$$f_i(\boldsymbol{x}_f, t + \delta_t) = a_1 f_{\bar{i}}(\boldsymbol{x}_f, t) + a_2 f_i(\boldsymbol{x}_f, t) + a_3 f_i^{(eq)}(\boldsymbol{x}_f, t) + a_4 f_{\bar{i}}^{(eq)}(\boldsymbol{x}_f, t) + 2\omega_i \rho_0 \frac{\mathbf{c}_i \cdot \boldsymbol{\phi}}{c_s^2} \tag{35}$$

with

$$a_1 = 1 - a_4, \quad a_2 = -a_3, \quad a_3 = \gamma(s_\nu - s_q), \quad a_4 = \gamma(s_\nu + s_q). \tag{36}$$

This scheme can be further written in the concise form:

$$f_i(\boldsymbol{x}_f, t + \delta_t) = 2\gamma f_{\bar{i}}^*(\boldsymbol{x}_f, t) + (1 - 2\gamma) f_{\bar{i}}(\boldsymbol{x}_f, t) + 2\omega_i \rho_0 \frac{\mathbf{c}_i \cdot \boldsymbol{\phi}}{c_s^2}, \tag{37}$$

where $f_{\bar{i}}^*(\boldsymbol{x}_f, t)$ is the post-collision distribution in Eq. (31). This is our first scheme and it reduces to the classical bounce-back scheme when $\gamma = 1/2$. Note that, when $\gamma > 1/2$, the above scheme is not a convex combination of $f_{\bar{i}}^*(\boldsymbol{x}_f, t)$ and $f_{\bar{i}}(\boldsymbol{x}_f, t)$, which may cause numerical instability.

To avoid the non-convex combination, we propose the following scheme

$$f_i(\boldsymbol{x}_f, t + \delta_t) = \frac{2\gamma}{1 + 2\gamma} f_i^*(\boldsymbol{x}_f, t) + \frac{1}{1 + 2\gamma} f_{\bar{i}}(\boldsymbol{x}_f, t) + \frac{2}{1 + 2\gamma} \omega_i \rho_0 \frac{\mathbf{c}_i \cdot \boldsymbol{\phi}}{c_s^2}, \tag{38}$$

which is obviously same as Scheme (37) when $\gamma = 0$. With (31), it is direct to verify that the coefficients $a_1, a_2, \ldots, a_5$ corresponding to the last boundary scheme are

$$\begin{aligned} &a_1 = \frac{1}{1 + 2\gamma} - a_4, \quad a_2 = \gamma a_5 - a_3, \quad a_3 = \frac{\gamma}{1 + 2\gamma}(s_\nu + s_q), \\ &a_4 = \frac{\gamma}{1 + 2\gamma}(s_\nu - s_q), \quad a_5 = \frac{2}{1 + 2\gamma}. \end{aligned} \tag{39}$$

These coefficients can be easily shown to satisfy the constraints (30) and thereby the scheme has second-order accuracy. We can see that the single-node scheme (38) is a convex combination of the pre- and post-collision distributions for all $0 \le \gamma \le 1$ and therefore is expected to exhibit good stability. This will be demonstrated by numerical examples in the next section.

5. Numerical experiments

To validate the boundary schemes (37) and (38) for the Dirichlet boundary condition (1), we conduct numerical experiments for the following three problems: the Poiseuille flow with straight boundaries, and the Taylor–Green vortex flow and the Taylor–Couette flow with curve boundaries. All these flows are governed by the two-dimensional incompressible Navier–Stokes equations

$$\nabla \cdot (u, v) = 0, \qquad \partial_t (u, v) + (u, v) \cdot \nabla (u, v) + \nabla p = \nu \Delta (u, v) + \boldsymbol{F} \tag{40}$$

in proper domains, where ν is the kinematic viscosity and $\boldsymbol{F}$ is an external force. They all have analytical solutions. For each numerical experiment, we only need to specify the relaxation time τ and lattice size h, which determine all other parameters: $\delta_x = h, \delta_t = \eta h^2, \eta = (\tau - 1/2)/(3\nu), s_\nu = 1/\tau$ and s_q is determined by s_ν via Eq. (9).

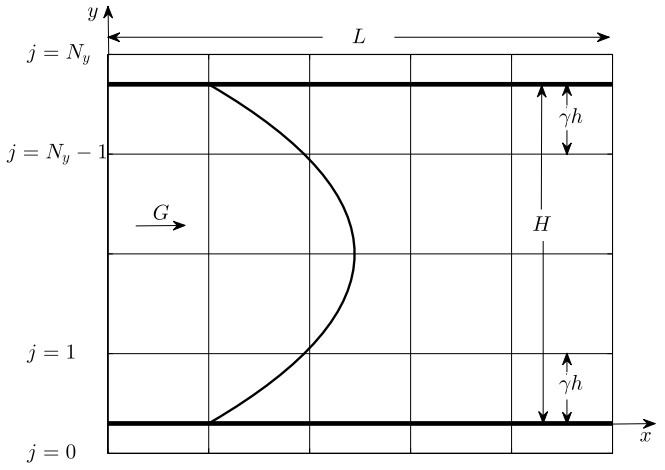

Fig. 2. Configuration of the Poiseuille flow in LBE simulations with an arbitrary γ.

5.1. Poiseuille flow

The first problem is the Poiseuille flow between two parallel no-slip walls driven by a constant body force $\boldsymbol{F} = G(1, 0)$ (see Fig. 2). This problem has the following analytical solution

$$u = u(y) = 4U(1 - \frac{y}{H})\frac{y}{H}, \quad v = 0, \tag{41}$$

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. Here we take

$$\nu = 0.003, \quad G = 0.8\nu, \quad H = 1.$$

In our computation, the horizontal direction is periodic. The boundary schemes are applied at the upper and lower straight boundaries. As illustrated in Fig. 2, N_y is the number of meshes in the vertical direction, and the lower and upper walls are located between $j = 0$ and $j = 1$, $j = N_y$ and $j = N_y - 1$, respectively. The lattice size is

$$h = \frac{H}{N_y - 2 + 2\gamma} \tag{42}$$

with γ the scaled distance. To demonstrate the accuracy and stability of the boundary schemes, we define the relative L^2-error as

$$E_r = \frac{\sqrt{\sum_{\boldsymbol{x}} |\boldsymbol{u}(\boldsymbol{x}) - \boldsymbol{u}^*(\boldsymbol{x})|^2}}{\sqrt{\sum_{\boldsymbol{x}} |\boldsymbol{u}^*(\boldsymbol{x})|^2}}, \tag{43}$$

where the summation is over all lattice nodes in the computational domain, $\boldsymbol{u}^* = (u, v)$ is the analytical solution (41), and $\boldsymbol{u}$ is the LB solution.

In our numerical experiments, we set $\gamma = 0.25, 0.75$ and 1, take different $\tau (= 0.505, 0.6, 1, 2, 3)$ and $N_y = 11, 21, 41, 61, 81$, and the number of meshes in the horizontal direction is $N_x = 2(N_y - 1)$. Note that the lattice size h is calculated by Eq. (42). Fig. 3 shows that with different γ and τ the convergence orders are all around 2 for both schemes. These show the second-order accuracy of the two specific schemes (37) and (38) for straight boundaries. From the figure, we can also see that Scheme (37) loses its stability when $\gamma > 0.5$, while Scheme (38) is stable almost for all tested γ and τ except for the cases where $\gamma = 0.25$, $\tau = 0.505$ and N_y is too small. This verifies the aforementioned statement that Scheme (37) may lose its stability when $\gamma > 1/2$ due to the non-convex combination of the distributions.

To further examine the stability of the boundary schemes (37) and (38) for different values of γ and τ, we fix $N_y = 41$ and set $\gamma = 0, 0.25, 0.5, 0.75$ and 1. For each γ, we find an interval of τ in which the LB solutions of the boundary schemes are convergent. Here by convergence we mean that

$$\frac{\sqrt{\sum_{\boldsymbol{x}} |\boldsymbol{u}(\boldsymbol{x}, t_n) - \boldsymbol{u}(\boldsymbol{x}, t_n - 1000\delta_t)|^2}}{\sqrt{\sum_{\boldsymbol{x}} |\boldsymbol{u}(\boldsymbol{x}, t_n)|^2}} < 10^{-10}. \tag{44}$$

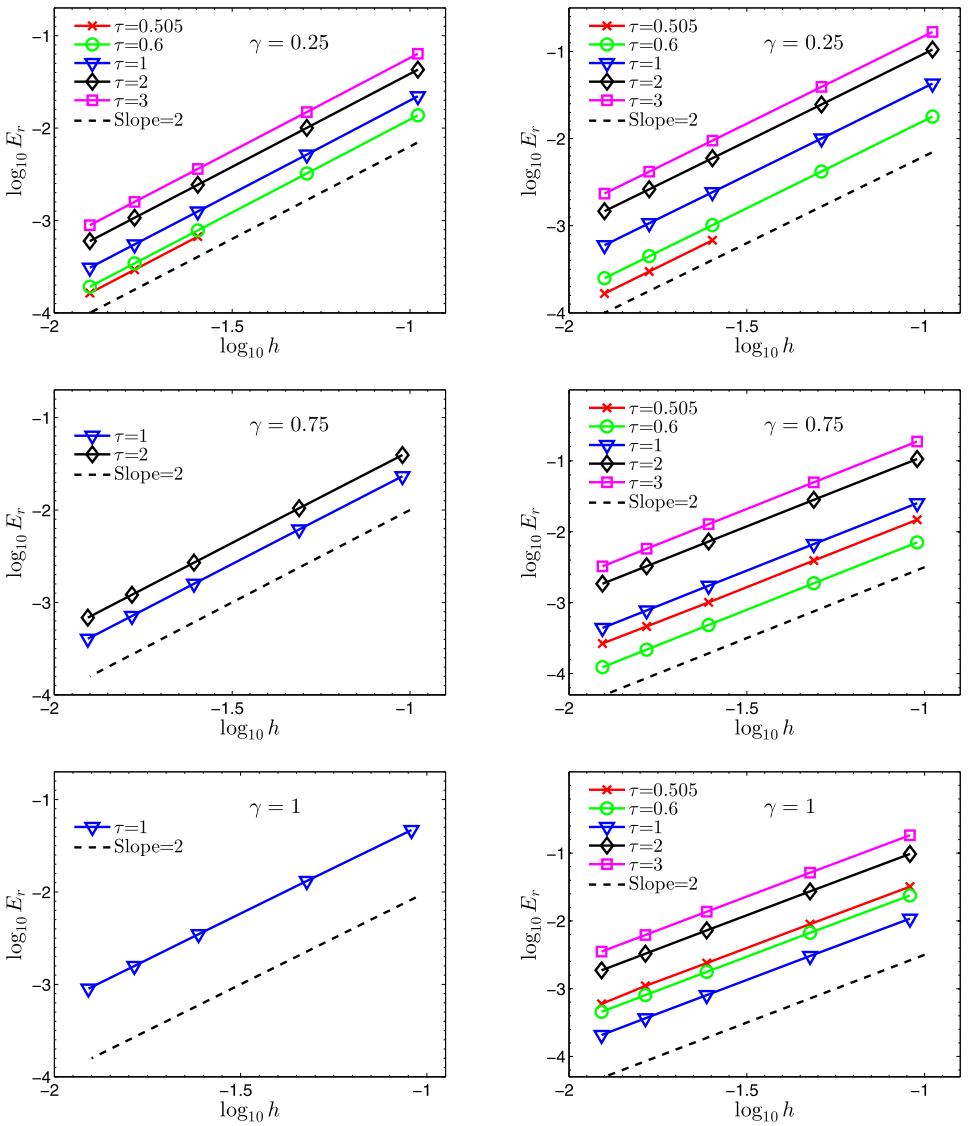

Fig. 3. Convergence order of the boundary schemes for the Poiseuille flow. From top to bottom: $\gamma = 0.25$, 0.75 and 1. Left: scheme (37), right: scheme (38).

Table 1
Stability intervals of τ with different γ for the schemes (37) and (38).

γ	0	0.25	0.5	0.75	1.0
Scheme (37)	[0.509, 3]	[0.504, 3]	[0.5001, 3]	[0.67, 2.7]	[0.83, 1.7]
Scheme (38)	[0.509, 3]	[0.503, 3]	[0.5001, 3]	[0.5001, 3]	[0.5001, 3]

Here $\boldsymbol{u}(\boldsymbol{x}, t_n)$ is the LB solution at the lattice node $\boldsymbol{x}$ and time t_n. To avoid too much computation time (for small τ) or too large Mach number Ma (for large τ), we only test τ in the interval [0.5001, 3]. With a large number of numerical experiments, we obtain the stability intervals of τ with different γ for the two boundary schemes (37) and (38). The results are given in Table 1. From Table 1, it can be seen that Scheme (37) is stable for large intervals of τ when $\gamma \leq 1/2$, while the stability interval shrinks as τ increases from 1/2 to 1. This once again verifies the aforementioned statement that the scheme (37) may lose its stability when $\gamma > 1/2$ due to the non-convex combination of the distributions. In contrast, Scheme (38) has large stability regions for all γ tested here, especially it is stable for all $\tau \in [0.5001, 3]$ when $\gamma \geq 1/2$. This may be attributed to the fact that the scheme is a convex combination of the distributions for all $0 \leq \gamma \leq 1$.

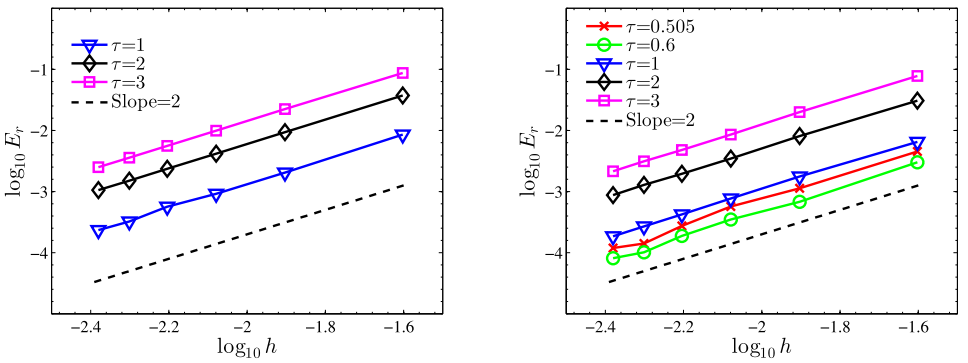

Fig. 4. Convergence order of the boundary schemes for the Taylor–Green vortex flow. Left: scheme (37), right: scheme (38).

5.2. Taylor–Green vortex flow in a circular domain

The second problem is the Taylor–Green vortex flow in the circular domain

$$\Omega := \left\{(x, y)|(x-\frac{1}{2})^2+(y-\frac{1}{2})^2 \le \frac{1}{16}\right\}$$

without external forces. This problem has analytic solutions

$$u = -U_0\cos(2\pi x)\sin(2\pi y)e^{-8\pi^2\nu t},$$
$$v = U_0\cos(2\pi y)\sin(2\pi x)e^{-8\pi^2\nu t},$$
$$p = p_0 - \frac{1}{4}U_0^2[\cos(4\pi x)+\cos(4\pi y)]e^{-16\pi^2\nu t}$$

with free parameters U_0 and p_0. In our numerical simulations reported below, we take the parameters as

$$\nu = 0.002, \quad U_0 = 0.05, \quad p_0 = \rho_0 c_s^2 \quad \text{with} \quad \rho_0 = 1.$$

The initial and boundary values are given by the above analytical solutions.

Let $\boldsymbol{u} = \boldsymbol{u}(\boldsymbol{x}, t)$ be the LB solution and $\boldsymbol{u}^* = (u, v)$ the above analytic solution. We define the relative L^2-error as

$$E_r = \frac{\sqrt{\sum_{\boldsymbol{x}}|\boldsymbol{u}(\boldsymbol{x}, T)-\boldsymbol{u}^*(\boldsymbol{x}, T)|^2}}{\sqrt{\sum_{\boldsymbol{x}}|\boldsymbol{u}^*(\boldsymbol{x}, T)|^2}} \tag{45}$$

at time $T = 1/U_0$, where the summation is over all lattice nodes in the circular domain Ω.

To examine the stability and accuracy of the boundary schemes (37) and (38), we take different $\tau(= 0.505, 0.6, 1, 2, 3)$ in the simulation with a number of spatial steps $h = 1/40$, $1/80$, $1/120$, $1/160$, $1/200$ and $1/240$. Fig. 4 shows that even with the curved boundary $\partial\Omega$, the convergence orders are all around 2 for both schemes with different τ. Moreover, numerical results show that the scheme (38) is stable for all τ taken here, while the scheme (37) is stable only for $\tau = 1, 2, 3$. This confirms our prediction that the scheme (38) has better stability than (37).

On the other hand, it seems that the error of the scheme (38) becomes larger with τ increasing from 0.6 to 3. This may be roughly explained by considering the Mach number

$$\mathrm{Ma} = \frac{U_0}{c_s} = \frac{(\tau-\frac{1}{2})U_0\delta_x}{\sqrt{3}\nu}$$

for the diffusive scaling. Namely, the Mach number increases as τ increases. Thus, when τ is large, the compressible effect will be prominent and thus the error will also increase. Also for this reason, the cases of $\tau > 3$ are not considered here due to the large Mach number.

5.3. Taylor–Couette flow between two concentric circular cylinders

In this flow, the outer cylinder has radius r_2 and is fixed, while the inner cylinder has radius $r_1(< r_2)$ and rotates with a constant angular velocity ω. There are no external forces. The flow has the steady analytic solution

$$(u, v) = \phi(r)(\cos\theta, \sin\theta)$$

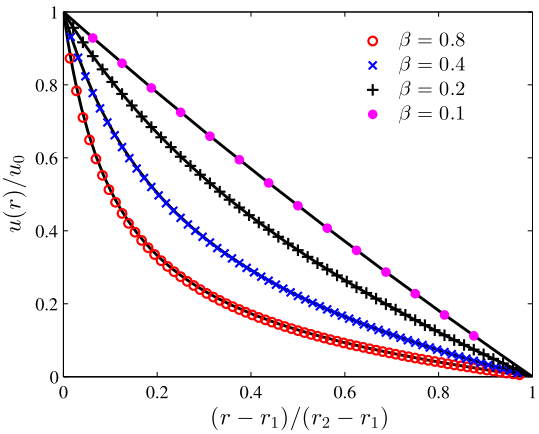

Fig. 5. Velocity profiles of the Taylor–Couette flow with different radius ratios β. The solid lines are the analytical solutions and the markers stand for the numerical results with the scheme (38).

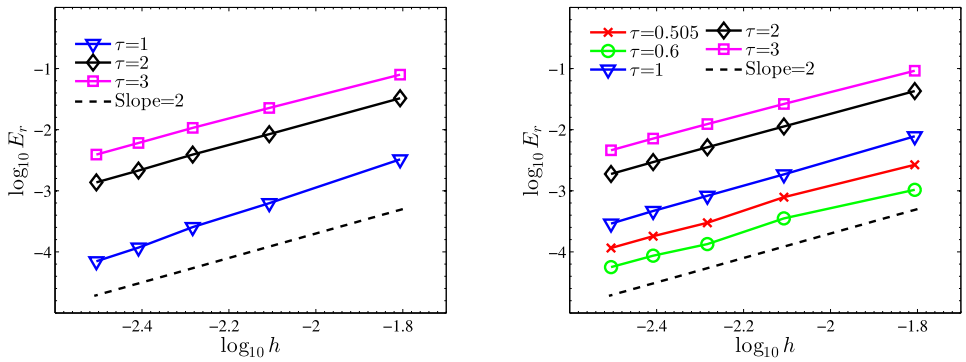

Fig. 6. Convergence order of the boundary schemes for the Taylor–Couette flow. Left: scheme (37), right: scheme (38).

with

$$\phi(r) = \frac{\phi_0 \beta}{1-\beta^2}\left(\frac{r_2}{r} - \frac{r}{r_2}\right)$$

in the polar coordinate (r, θ), where $\phi_0 = \omega r_1, \beta = r_1/r_2$ and $r \in [r_1, r_2]$.

In the simulation, the kinematic viscosity ν and the radius r_2 are fixed:

$$\nu = 0.02, \quad r_2 = 0.8$$

and the angular velocity ω is $0.1/r_1$. The boundary schemes are applied to the boundaries of the two cylinders. We set $\tau = 1$, $h = r_2/80 = 0.01$ and change the ratio β to test the stability of the two boundary schemes. As demonstrated in Fig. 5, numerical results of the boundary schemes are in excellent agreement with the analytical solutions. Here only the results with the scheme (38) are plotted for simplicity.

Moreover, we fix $\beta = 0.5$ and compute the relative L^2-error, as defined in Eq. (43), for different τ and $h(= 0.8/32, 0.8/64, 0.8/96, 0.8/128, 0.8/160)$ to test the stability and accuracy of the boundary schemes. From Fig. 6, one can see that the scheme (38) almost has second order accuracy for all the values of τ tested here, while (37) is stable only for $\tau = 1, 2, 3$. These findings are similar to those for the Taylor–Green vortex flow. Furthermore, based on the behaviors of the boundary schemes displayed in Figs. 4 and 6, we recommend to choose the relaxation rate $s_\nu = 1/\tau$ close to 1 in order to balance the stability, accuracy and efficiency.

6. Conclusions and remarks

In this work, we present a class of single-node second-order boundary schemes accompanying a TRT-LBE for the Dirichlet boundary condition of the Navier–Stokes equations defined in a domain with curved boundaries. The schemes are derived by using the recently developed Maxwell iteration [22] for the LBE. Unlike the interpolation based schemes for curved boundaries [13–15,17,16,19] which involve several lattice nodes, our schemes apply to the situations where not enough neighboring nodes are available [18]. In addition, our schemes are much simpler than the one-point scheme in [18].

Moreover, two specific boundary schemes are constructed and their accuracy and stability are investigated through several numerical examples. The results validate the second-order accuracy of both schemes and show that the scheme with a convex combination of DFs has better stability.

It is remarkable that only two groups of the five combination coefficients in (22) are singled out from the three constraints (30). Other single-node second-order boundary schemes could be constructed by fixing the five coefficients satisfying the constraints, which may give better boundary schemes.

Moreover, our idea in deriving the boundary schemes for the Dirichlet BCs is promising to be extended to Neumann BCs or more complicated BCs. Indeed, the key formula (20) contains derivatives of the fluid velocity. Thus, the stress tensor can be easily expressed in terms of the distribution functions. With this, we may construct schemes for outflow boundary conditions [36], boundary conditions for free interface problems [37], and the slip boundary conditions in rarefied flows [38,39]. We hope to report our progress in the near future.

Finally, an important issue would be to apply our boundary schemes to moving boundary problems. This seems doable but further investigations are needed, which is beyond the scope of this work. Indeed, when the boundary is moving, the additional work to do is to specify the unknown distribution functions at the lattice nodes that move out of the non-fluid region into the fluid region to become fluid nodes. This can be done by using the extrapolation methods as proposed in [17,40]. In this way, our single-node schemes could be used to handle moving boundary problems. However, this straightforward application needs to be examined carefully, because our boundary schemes do not ensure the mass conservation. In fact, it was observed in [40] that, for moving boundary problems, boundary schemes breaking the mass conservation may induce excessive perturbations of the flow fields.

Acknowledgements

The authors would like to thank Profs. Li-Shi Luo and Zhaoli Guo for valuable discussions. The second author was financially supported by the National Natural Science Foundation of China (NSFC 11471185) and by the Tsinghua University Initiative Scientific Research Program (20151080424).

Appendix

In this appendix, we use the Maxwell iteration [22,33–35] to derive the expansion (16) from Eq. (15).

First of all, we truncate the expansion in Eq. (15) up to $O(h^3)$ to get

$$\begin{aligned} & h(\boldsymbol{e}_i\cdot\nabla)f_i+\frac{1}{2}h^2(\boldsymbol{e}_i\cdot\nabla)^2 f_i+h^2\eta\partial_t f_i \\ =&-s_\nu\left[\frac{1}{2}(f_i+f_{\bar{i}})-\frac{1}{2}\left(f_i^{(eq)}+f_{\bar{i}}^{(eq)}\right)\right] \\ &-s_q\left[\frac{1}{2}(f_i-f_{\bar{i}})-\frac{1}{2}\left(f_i^{(eq)}-f_{\bar{i}}^{(eq)}\right)\right]+O(h^3). \end{aligned}$$

From this we deduce that

$$\begin{aligned} -\frac{1}{2}(s_\nu+s_q)f_i=&\frac{1}{2}(s_\nu-s_q)f_{\bar{i}}-\frac{1}{2}(s_\nu-s_q)f_{\bar{i}}^{(eq)}-\frac{1}{2}(s_\nu+s_q)f_i^{(eq)} \\ &+h(\boldsymbol{e}_i\cdot\nabla)f_i+\frac{1}{2}h^2(\boldsymbol{e}_i\cdot\nabla)^2 f_i+h^2\eta\partial_t f_i+O(h^3) \end{aligned}$$

and thereby

$$\begin{aligned} f_i=&-\frac{s_\nu-s_q}{s_\nu+s_q}f_{\bar{i}}+\frac{s_\nu-s_q}{s_\nu+s_q}f_{\bar{i}}^{(eq)}+f_i^{(eq)} \\ &-\frac{2}{s_\nu+s_q}\left[h(\boldsymbol{e}_i\cdot\nabla)f_i+\frac{1}{2}h^2(\boldsymbol{e}_i\cdot\nabla)^2 f_i+h^2\eta\partial_t f_i\right]+O(h^3). \end{aligned} \tag{46}$$

Set

$$a=\frac{s_\nu-s_q}{s_\nu+s_q},\quad b=\frac{2}{s_\nu+s_q}\quad\text{and}\quad K_i=-af_{\bar{i}}+af_{\bar{i}}^{(eq)}+f_i^{(eq)}.$$

Then Eq. (46) can be written as

$$f_i=K_i-b\left[h(\boldsymbol{e}_i\cdot\nabla)f_i+\frac{1}{2}h^2(\boldsymbol{e}_i\cdot\nabla)^2 f_i+h^2\eta\partial_t f_i\right]+O(h^3). \tag{47}$$

With the help of the Maxwell iteration [22,33–35], that is, substituting f_i in Eq. (47) into its right-hand side, we obtain

$$f_i = K_i - bh\,(\boldsymbol{e}_i\cdot\nabla)\,K_i + h^2\left[(b^2-\frac{1}{2}b)\,(\boldsymbol{e}_i\cdot\nabla)^2 - b\eta\partial_t\right]K_i + O(h^3).$$

Similarly, we have

$$\begin{aligned} f_{\bar{i}} &= K_{\bar{i}} - bh\,(\boldsymbol{e}_{\bar{i}}\cdot\nabla)\,K_{\bar{i}} + h^2\left[(b^2-\frac{1}{2}b)\,(\boldsymbol{e}_{\bar{i}}\cdot\nabla)^2 - b\eta\partial_t\right]K_{\bar{i}} + O(h^3)\\ &= -af_i + af_i^{(eq)} + f_{\bar{i}}^{(eq)} + bh\,(\boldsymbol{e}_i\cdot\nabla)\left(-af_i + af_i^{(eq)} + f_{\bar{i}}^{(eq)}\right)\\ &\quad + h^2\left[(b^2-\frac{1}{2}b)\,(\boldsymbol{e}_i\cdot\nabla)^2 - b\eta\partial_t\right]\left(-af_i + af_i^{(eq)} + f_{\bar{i}}^{(eq)}\right) + O(h^3). \end{aligned}$$

Substituting this into Eq. (47), we obtain

$$\begin{aligned} f_i &= -af_{\bar{i}} + af_{\bar{i}}^{(eq)} + f_i^{(eq)} - b\left[h\,(\boldsymbol{e}_i\cdot\nabla)\,f_i + \frac{1}{2}h^2\,(\boldsymbol{e}_i\cdot\nabla)^2\,f_i + h^2\eta\partial_t f_i\right] + O(h^3)\\ &= a^2 f_i - a^2 f_i^{(eq)} - af_{\bar{i}}^{(eq)} - abh\,(\boldsymbol{e}_i\cdot\nabla)\left(-af_i + af_i^{(eq)} + f_{\bar{i}}^{(eq)}\right)\\ &\quad - h^2 a\left[(b^2-\frac{1}{2}b)\,(\boldsymbol{e}_i\cdot\nabla)^2 - b\eta\partial_t\right]\left(-af_i + af_i^{(eq)} + f_{\bar{i}}^{(eq)}\right)\\ &\quad + af_{\bar{i}}^{(eq)} + f_i^{(eq)} - b\left[h\,(\boldsymbol{e}_i\cdot\nabla)\,f_i + \frac{1}{2}h^2\,(\boldsymbol{e}_i\cdot\nabla)^2\,f_i + h^2\eta\partial_t f_i\right] + O(h^3)\\ &= a^2 f_i + (1-a^2)\,f_i^{(eq)} - abh\,(\boldsymbol{e}_i\cdot\nabla)\left(af_i^{(eq)} + f_{\bar{i}}^{(eq)}\right)\\ &\quad - h^2 a\left[(b^2-\frac{1}{2}b)\,(\boldsymbol{e}_i\cdot\nabla)^2 - b\eta\partial_t\right]\left(af_i^{(eq)} + f_{\bar{i}}^{(eq)}\right) - (1-a^2)bh\,(\boldsymbol{e}_i\cdot\nabla)\,f_i\\ &\quad - h^2\left[(\frac{1}{2}b + \frac{1}{2}a^2b - a^2b^2)\,(\boldsymbol{e}_i\cdot\nabla)^2 + (b+a^2b)\eta\partial_t\right]f_i + O(h^3). \end{aligned}$$

Notice that $a^2 < 1$. The last equation further yields

$$\begin{aligned} f_i &= f_i^{(eq)} - \frac{ab}{1-a^2}h\,(\boldsymbol{e}_i\cdot\nabla)\,(af_i^{(eq)} + f_{\bar{i}}^{(eq)})\\ &\quad - \frac{a}{1-a^2}h^2\left[(b^2-\frac{1}{2}b)\,(\boldsymbol{e}_i\cdot\nabla)^2 - b\eta\partial_t\right](af_i^{(eq)} + f_{\bar{i}}^{(eq)}) - bh\,(\boldsymbol{e}_i\cdot\nabla)\,f_i\\ &\quad - \frac{1}{1-a^2}h^2\left[(\frac{1}{2}b + \frac{1}{2}a^2b - a^2b^2)\,(\boldsymbol{e}_i\cdot\nabla)^2 + (b+a^2b)\eta\partial_t\right]f_i + O(h^3). \end{aligned}$$

Iterating this equation once again yields

$$\begin{aligned} f_i &= f_i^{(eq)} - \frac{ab}{1-a^2}h\,(\boldsymbol{e}_i\cdot\nabla)\,(af_i^{(eq)} + f_{\bar{i}}^{(eq)}) - \frac{a}{1-a^2}h^2\left[(b^2-\frac{1}{2}b)\,(\boldsymbol{e}_i\cdot\nabla)^2 - b\eta\partial_t\right](af_i^{(eq)} + f_{\bar{i}}^{(eq)})\\ &\quad - bh\,(\boldsymbol{e}_i\cdot\nabla)\,f_i^{(eq)} + \frac{ab^2}{1-a^2}h^2\,(\boldsymbol{e}_i\cdot\nabla)^2\,(af_i^{(eq)} + f_{\bar{i}}^{(eq)})\\ &\quad - \frac{1}{1-a^2}h^2\left[(\frac{1}{2}b + \frac{1}{2}a^2b - a^2b^2)\,(\boldsymbol{e}_i\cdot\nabla)^2 + (b+a^2b)\eta\partial_t\right]f_i^{(eq)} + O(h^3)\\ &= f_i^{(eq)} - \left(\frac{a^2b}{1-a^2} + b\right)h\,(\boldsymbol{e}_i\cdot\nabla)\,f_i^{(eq)} - \frac{ab}{1-a^2}h\,(\boldsymbol{e}_i\cdot\nabla)\,f_{\bar{i}}^{(eq)}\\ &\quad - \frac{1}{1-a^2}h^2\left[(a^2b^2 - \frac{1}{2}a^2b - a^2b^2 + \frac{1}{2}b + \frac{1}{2}a^2b - a^2b^2)\,(\boldsymbol{e}_i\cdot\nabla)^2 + (-a^2b + b + a^2b)\eta\partial_t\right]f_i^{(eq)}\\ &\quad - \frac{1}{1-a^2}h^2\left[(ab^2 - \frac{1}{2}ab - ab^2)\,(\boldsymbol{e}_i\cdot\nabla)^2 - ab\eta\partial_t\right]f_{\bar{i}}^{(eq)} + O(h^3)\\ &= f_i^{(eq)} - \frac{b}{1-a^2}h\,(\boldsymbol{e}_i\cdot\nabla)\,(f_i^{(eq)} + af_{\bar{i}}^{(eq)}) - \frac{1}{1-a^2}h^2\left[(\frac{1}{2}b - a^2b^2)\,(\boldsymbol{e}_i\cdot\nabla)^2 + b\eta\partial_t\right]f_i^{(eq)}\\ &\quad + \frac{ab}{1-a^2}h^2\left[\frac{1}{2}\,(\boldsymbol{e}_i\cdot\nabla)^2 + \eta\partial_t\right]f_{\bar{i}}^{(eq)} + O(h^3). \end{aligned}$$

Consequently, we arrive at the expansion

$$f_i = f_i^{(eq)} - \frac{b}{1-a^2} h \left(\boldsymbol{e}_i \cdot \nabla\right) \left(f_i^{(eq)} + a f_{\bar{i}}^{(eq)}\right) - \frac{1}{1-a^2} h^2 \left[\left(\frac{1}{2} b - a^2 b^2\right) \left(\boldsymbol{e}_i \cdot \nabla\right)^2 + b\eta\partial_t \right] f_i^{(eq)} + \frac{ab}{1-a^2} h^2 \left[\frac{1}{2} \left(\boldsymbol{e}_i \cdot \nabla\right)^2 + \eta\partial_t \right] f_{\bar{i}}^{(eq)} + O\left(h^3\right).$$

References

[1] X. He, L.-S. Luo, A priori derivation of the lattice Boltzmann equation, Phys. Rev. E 55 (6) (1997) R6333–R6336.
[2] X. He, L.-S. Luo, Theory of the lattice Boltzmann method: from the Boltzmann equation to the lattice Boltzmann equation, Phys. Rev. E 56 (6) (1997) 6811–6817.
[3] L.-S. Luo, M. Krafczyk, W. Shyy, in: R. Blockley, W. Shyy (Eds.), Encyclopedia of Aerospace Engineering, Wiley, New York, 2010, pp. 651–660, Chap. 56.
[4] D. Yu, R. Mei, L.-S. Luo, W. Shyy, Viscous flow computations with the method of lattice Boltzmann equation, Prog. Aerosp. Sci. 39 (5) (2003) 329–367.
[5] S. Chen, G.D. Doolen, Lattice Boltzmann method for fluid flows, Annu. Rev. Fluid Mech. 30 (1) (1998) 329–364.
[6] J.D. Anderson, Computational Fluid Dynamics, 1st edition, McGraw–Hill Education, New York, 1995.
[7] J. Blazek, Computational Fluid Dynamics: Principles and Applications, 2nd edition, Elsevier, New York, 2005.
[8] A.J.C. Ladd, Numerical simulations of particulate suspensions via a discretized Boltzmann equation. Part 1. Theoretical foundation, J. Fluid Mech. 271 (1994) 285–309.
[9] A.J.C. Ladd, Numerical simulations of particulate suspensions via a discretized Boltzmann equation. Part 2. Numerical results, J. Fluid Mech. 271 (1994) 311–339.
[10] D.R. Noble, S. Chen, J.G. Georgiadis, R.O. Buckius, A consistent hydrodynamic boundary condition for the lattice Boltzmann method, Phys. Fluids 7 (7) (1995) 203–209.
[11] T. Inamuro, M. Yoshino, F. Ogino, A non-slip boundary condition for lattice Boltzmann simulations, Phys. Fluids 7 (12) (1995) 2928–2930.
[12] I. Ginzburg, D. d'Humières, Local second-order boundary method for lattice Boltzmann models, J. Stat. Phys. 84 (5) (1996) 927–971.
[13] O. Filippova, D. Hänel, Grid refinement for lattice-BGK models, J. Comput. Phys. 147 (1998) 219–228.
[14] R. Mei, L.-S. Luo, W. Shyy, An accurate curved boundary treatment in lattice Boltzmann method, J. Comput. Phys. 155 (1999) 307–330.
[15] M. Bouzidi, M. Firdaouss, P. Lallemand, Momentum transfer of a Boltzmann-lattice fluid with boundaries, Phys. Fluids 13 (11) (2001) 3452–3459.
[16] Z. Guo, C. Zheng, B. Shi, An extrapolation method for boundary conditions in lattice Boltzmann method, Phys. Fluids 14 (2002) 2007–2010.
[17] P. Lallemand, L.-S. Luo, Lattice Boltzmann method for moving boundaries, J. Comput. Phys. 184 (2003) 406–421.
[18] M. Junk, Z. Yang, One-point boundary condition for the lattice Boltzmann method, Phys. Rev. E 72 (6) (2005) 066701.
[19] X. Yin, J. Zhang, An improved bounce-back scheme for complex boundary conditions in lattice Boltzmann method, J. Comput. Phys. 231 (2012) 4295–4303.
[20] J. Huang, W.-A. Yong, Boundary conditions of the lattice Boltzmann method for convection–diffusion equations, J. Comput. Phys. 300 (2015) 70–91.
[21] J. Huang, Z. Hu, W.-A. Yong, Second-order curved boundary treatments of the lattice Boltzmann method for convection–diffusion equations, J. Comput. Phys. 310 (2016) 26–44.
[22] W.-A. Yong, W. Zhao, L.-S. Luo, Theory of the lattice Boltzmann method: derivation of macroscopic equations via the Maxwell iteration, Phys. Rev. E 93 (2016) 033310.
[23] D. d'Humières, Rarefied gas dynamics: theory and simulations, in: B.D. Shizgal, D.P. Weave (Eds.), Prog. Astronaut. Aeronaut., vol. 159, AIAA, Washington, D.C., 1992, p. 450.
[24] P. Lallemand, L.-S. Luo, Theory of the lattice Boltzmann method: dispersion, dissipation, isotropy, Galilean invariance, and stability, Phys. Rev. E 61 (2000) 6546–6562.
[25] M. Junk, W.-A. Yong, Weighted L^2-stability of the lattice Boltzmann method, SIAM J. Numer. Anal. 47 (3) (2009) 1651–1665.
[26] I. Ginzburg, Equilibrium-type and link-type lattice Boltzmann models for generic advection and anisotropic-dispersion equation, Adv. Water Resour. 28 (11) (2005) 1171–1195.
[27] I. Ginzburg, F. Verhaeghe, D. d'Humières, Two-relaxation-time lattice Boltzmann scheme: about parameterization, velocity, pressure and mixed boundary conditions, Commun. Comput. Phys. 3 (2008) 427–478.
[28] I. Ginzburg, F. Verhaeghe, D. d'Humières, Study of simple hydrodynamic solutions with the two-relation-times lattice Boltzmann scheme, Commun. Comput. Phys. 3 (2008) 519–581.
[29] X. He, L.-S. Luo, Lattice Boltzmann model for the incompressible Navier–Stokes equation, J. Stat. Phys. 88 (3) (1997) 927–944.
[30] M. Junk, W.-A. Yong, Rigorous Navier–Stokes limit of the lattice Boltzmann equation, Asymptot. Anal. 35 (2) (2003) 165–185.
[31] M. Junk, A. Klar, L.-S. Luo, Asymptotic analysis of the lattice Boltzmann equation, J. Comput. Phys. 210 (2) (2005) 676–704.
[32] Z. Yang, W.-A. Yong, Asymptotic analysis of the lattice Boltzmann method for generalized Newtonian fluid flows, Multiscale Model. Simul. 12 (3) (2014) 1028–1045.
[33] I. Müller, T. Ruggeri, Rational Extended Thermodynamics, Springer Tracts Natl. Philos., vol. 37, Springer, Heidelberg, 1998.
[34] W.-A. Yong, Diffusive relaxation limit of multi-dimensional isentropic hydrodynamical models for semiconductors, SIAM J. Appl. Math. 64 (5) (2004) 1737–1748.
[35] Z. Yang, W.-A. Yong, Validity of the Chapman–Enskog expansion for a class of hyperbolic relaxation systems, J. Differ. Equ. 258 (8) (2015) 2745–2766.
[36] S. Dong, G.E. Karniadakis, C. Chryssostomidis, A robust and accurate outflow boundary condition for incompressible flow simulations on severely-truncated unbounded domains, J. Comput. Phys. 261 (2014) 83–105.
[37] S. Bogner, R. Ammer, U. Rüde, Boundary conditions for free interfaces with the lattice Boltzmann method, J. Comput. Phys. 297 (2015) 1–12.
[38] D.A. Lockerby, J.M. Reese, Velocity boundary condition at solid walls in rarefied gas calculations, Phys. Rev. E 70 (2004) 017303.
[39] T. Reis, P.J. Dellar, Lattice Boltzmann simulations of pressure-driven flows in microchannels using Navier–Maxwell slip boundary conditions, Phys. Fluids 24 (2012) 112001.
[40] P.-H. Kao, R.-J. Yang, An investigation into curved and moving boundary treatments in the lattice Boltzmann method, J. Comput. Phys. 227 (2008) 5671–5690.