

One-step simplified lattice Boltzmann method of thermal flows under the Boussinesq approximationShenglei Qin (秦声雷)¹, Guoxiang Hou (侯国祥)^{1,*}, Liuming Yang (杨留名)^{2,3,†}
Xu Liu (刘旭)¹ and Haoze Luo (罗皓泽)¹¹*School of Naval Architecture and Ocean Engineering, Huazhong University of Science and Technology, Wuhan, Hubei 430074, China*²*Green & Smart River-Sea-Going Ship, Cruise and Yacht Research Center, Wuhan University of Technology, Wuhan 430063, China*³*Sanya Science and Education Innovation Park of Wuhan University of Technology, Sanya 572025, China*

(Received 25 April 2023; accepted 20 September 2023; published 12 October 2023)

In recent years, the simplified lattice Boltzmann method without evolution of distribution functions was developed, which adopts predictor-corrector steps to solve the constructed macroscopic equations. To directly solve the constructed macroscopic equations in a single step, we propose the present one-step simplified lattice Boltzmann method and apply it to simulate thermal flows under the Boussinesq approximation. The present method is derived by reconstructing the evolution equation of the lattice Boltzmann method and constructing nonequilibrium distribution functions. This method inherits the advantages of the simplified lattice Boltzmann method, such as low virtual memory cost, convenient boundary treatment, and good numerical stability at relaxation time close to 0.5. In addition, compared to the traditional artificial compressible method (ACM), the present method is more efficient in computation when a small time step is applied in the ACM to ensure numerical stability. Several numerical examples, including natural convection in a square cavity, the porous plate problem, and natural convection in a concentric annulus, are conducted to test the accuracy of the present method. The results show that this method can accurately simulate thermal flow problems and has good numerical stability.

DOI: [10.1103/PhysRevE.108.045305](https://doi.org/10.1103/PhysRevE.108.045305)**I. INTRODUCTION**

In general, heat transfer can be classified into three categories, namely, the heat conduction, heat convection, and heat radiation. As one of the basic ways of transfer, heat convection is a thermal exchange phenomenon accompanied by mass transfer, which can be further and roughly categorized as forced convection and natural convection. The forced convection is caused by external factors like pumps, winds, and fans, while the natural convection is spontaneous and generated by inner factors usually referring to species molar concentration difference or temperature difference. Numerically studying natural convection is of great importance due to its wide existence in engineering applications [1,2] such as nuclear reactor equipment, electronic devices, solar thermal systems, insulation of naval architecture cabin, and so on. Modeling the natural convection depends on the corresponding assumptions. For the fully compressible flows, the convection can be governed by Navier-Stokes equations and state equation [3], while the thermal Boussinesq approximation [4] is suitable for the incompressible flows which ignore the energy loss caused by heat dissipation of viscous friction and the compression work of pressure stress.

Among those varieties of numerical methods, the lattice Boltzmann method (LBM) [5–7] is a promising tool to simulate the thermal flows, which may be attributed to its simplicity in programming, extensibility in modeling, and good performance in parallel computing. In the applications of

thermal flows, the lattice Boltzmann (LB) model can be generally divided into three categories as multispeed (MS) model [8,9], hybrid model [10,11], double distribution function (DDF), or multidistribution function (MDF) model [12–14]. In terms of areas out of the thermal flows, the LBM has also been applied to solve various nonlinear partial differential equations, such as convection-diffusion equation [15] (CDE), Burgers equation [16], Poisson equation [17], and so on. With the development of the last 30 years, although the LBM has achieved a great success in the area of computational fluid dynamics (CFD), it has been criticized for several aspects. First of all, tracking and storing the distribution functions will burden extra virtual memory cost in computation. For example, including macroscopic variables required to be stored such as velocity, the D2Q9 model [7] requires storing extra nine distribution functions in each computational node. Second, the extra work of converting physical boundary conditions into boundary conditions of distribution functions sometimes is inconvenient, especially for the case of those complex and curved boundaries. Third, the numerical stability is still a great challenge when the relaxation time is close to 0.5, which is usually related to cases of high Reynolds number.

Given the above limitations of the LBM, Chen *et al.* [18] first proposed a simplified lattice Boltzmann method (SLBM) without evolution of distribution function to simulate incompressible flows. In their work, constructed macroscopic equations are derived through Chapman-Enskog (CE) expansion, and solved by two steps, namely, predictor and corrector steps. In each step, density and momentum are explicitly obtained by macroscopic variables contained in the equilibrium distribution functions, which makes it possible to avoid tracking and storing distribution functions. This immediately

*Corresponding author: houguoxiang@163.com†Corresponding author: yanglm_hust@foxmail.com;
yangliuming@whut.edu.cn

brings two merits, namely, low virtual memory cost and direct implementation of boundary conditions. In addition, compared to the single-relaxation-time (SRT) LBM, the SLBM is considered to have a better performance on numerical stability [18,19]. Up to now, inheriting the good scalability of modeling from the LBM, the SLBM has been extended to multiple fields. In the field of multiphase flows, Chen *et al.* proposed simplified multiphase lattice Boltzmann method [20] (SMLBM) to simulate liquid-gas problems and phase-field-simplified lattice Boltzmann method [21] (PF-SLBM) to simulate solid-liquid phase change problems. In the application of moving boundaries, immersed boundary-simplified lattice Boltzmann method [22] (IB-SLBM) and immersed boundary-simplified thermal lattice Boltzmann method [23] (IB-STLBM) couple the immersed boundary method (IBM) and the SLBM to simulate flows with immersed objectives. For other application areas, one may refer to [24–26].

However, as pointed out by Qin [27], the repeated work of calculating macroscopic variables and dealing with boundary conditions in the predictor and corrector steps of the SLBM cuts down the computational efficiency. This encourages us to develop the present one-step simplified lattice Boltzmann method (OSLBM). In this paper, by reconstructing the evolution equation of the SRT collision model of the LBM and constructing the unknown nonequilibrium distribution functions, we develop a one-step simplified lattice Boltzmann method of thermal flows under the Boussinesq approximation. This one-step simplified lattice Boltzmann method inherits the merits of the SLBM such as low virtual memory cost of directly storing the macroscopic variables, convenient boundary treatment, and good numerical stability when the relaxation time is close to 0.5. The rest of the paper is organized as follows. The basic governing equations and numerical models of the LBM are given in Secs. II A and II B, respectively. The reconstructed evolution equation, the constructed nonequilibrium distribution functions and the present one-step simplified lattice Boltzmann method of the thermal flows are derived in Sec. II C. Section II D introduces the boundary treatment of the present method. Then the computational efficiency and numerical stability of the present method are compared with that of the traditional artificial compressible method, LBM, SLBM in Sec. II E. In Sec. III, we present numerical examples, including natural convection in a square cavity, the porous plate problem, natural convection in a concentric annulus, to test the accuracy of the present method. A brief summary is given in Sec. IV.

II. NUMERICAL METHODS

A. Macroscopic governing equations

Under the condition of low Mach number and continuity, it is widely accepted that the governing equations of the velocity field are the incompressible Navier-Stokes equations which can be given as

$$\nabla \cdot \mathbf{u} = 0, \quad (1)$$

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u} \mathbf{u}) = -\nabla p + \nu \nabla \cdot [\nabla \rho \mathbf{u} + \nabla^T(\rho \mathbf{u})] + \mathbf{F}. \quad (2)$$

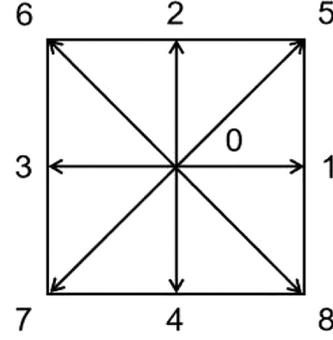


FIG. 1. Sketch of the D2Q9 model.

ρ , \mathbf{u} , and p are the macroscopic density, velocity, and pressure, respectively. ν is the kinematic viscosity and \mathbf{F} is the body force. In the well-known Boussinesq approximation [4], the density ρ can be approximated as a constant ρ_0 , while the temperature influence on density perturbation is only considered in the buoyancy force. So Eq. (2) can be written as

$$\begin{aligned} \frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot (\mathbf{u} \mathbf{u}) \\ = -\frac{\nabla p}{\rho_0} + \nu \nabla \cdot [\nabla \mathbf{u} + \nabla^T(\mathbf{u})] - \mathbf{g} \beta' (T - T_m), \end{aligned} \quad (3)$$

where T_m is the average temperature and \mathbf{g} is the gravity acceleration. T is the temperature. β' is the coefficient of thermal expansion. In the meantime, the heat dissipation of viscosity and the compression work of pressure are ignored, which gives the governing equation of the temperature field by the following passive-scalar equation:

$$\frac{\partial T}{\partial t} + \nabla \cdot (T \mathbf{u}) = \chi \nabla^2 T, \quad (4)$$

where χ is the thermal diffusivity.

B. The lattice Boltzmann method

In this paper, a widely used D2Q9 model [7] (Fig. 1) is applied for both temperature and velocity fields. The discrete velocity \mathbf{e}_i of nine directions and weight coefficients ω_i are displayed as

$$\mathbf{e}_i = (e_{ix}, e_{iy})^T = \begin{cases} (0, 0)^T, & i = 0, \\ c(\cos[(i-1)\frac{\pi}{2}], \sin[(i-1)\frac{\pi}{2}])^T, & i = 1-4, \\ \sqrt{2}c(\cos[(2i-1)\frac{\pi}{4}], \sin[(2i-1)\frac{\pi}{4}])^T, & i = 5-8, \end{cases} \quad (5)$$

$$\omega_{i=0} = \frac{4}{9}, \quad \omega_{i=1-4} = \frac{1}{9}, \quad \omega_{i=5-8} = \frac{1}{36}, \quad (6)$$

where $c = \delta x / \delta t = 1$ is the lattice speed, and δx and δt are lattice spacing and time step, respectively.

To simulate the temperature field, a single-relaxation-time lattice Boltzmann equation [28] can be given as

$$f_i(\mathbf{x} + \mathbf{e}_i \delta t, t + \delta t) - f_i(\mathbf{x}, t) = -\frac{1}{\tau_f} [f_i(\mathbf{x}, t) - f_i^{eq}(\mathbf{x}, t)], \quad (7)$$

where $f_i(\mathbf{x}, t)$ is the energy or temperature distribution function at space coordinate \mathbf{x} and time level t , and f_i^{eq} is the

equilibrium distribution function which can be expressed as

$$f_i^{\text{eq}} = \omega_i T \left(1 + \frac{\mathbf{e}_i \cdot \mathbf{u}}{c_s^2} \right). \quad (8)$$

The relaxation time τ_f is related to the thermal diffusivity as $\chi = c_s^2(\tau_f - 0.5)\delta t$. $c_s = c/\sqrt{3}$ is the lattice speed of sound. For convenience, the moment conditions are given as

$$\sum_i f_i = \sum_i f_i^{\text{eq}} = T, \quad (9)$$

$$\sum_i f_i^{\text{neq}} = \sum_i \epsilon f_i^{(1)} = 0, \quad (10)$$

$$\sum_i \mathbf{e}_i f_i^{\text{eq}} = T \mathbf{u}. \quad (11)$$

$f_i^{\text{neq}} = f_i - f_i^{\text{eq}}$ is the nonequilibrium part of the distribution function. It is usually assumed to be $f_i^{\text{neq}} = \epsilon f_i^{(1)} + \epsilon^2 f_i^{(2)}$ in the Chapman-Enskog analysis where ϵ is a small parameter proportion to the Knudsen number.

Similarly, to simulate the velocity field, a single-relaxation-time Boltzmann method [29,30] of solving incompressible NS equations is given as

$$\begin{aligned} & g_i(\mathbf{x} + \mathbf{e}_i \delta t, t + \delta t) - g_i(\mathbf{x}, t) \\ &= -\frac{1}{\tau_g} [g_i(\mathbf{x}, t) - g_i^{\text{eq}}(\mathbf{x}, t)] + \delta t G_i(\mathbf{x}, t), \end{aligned} \quad (12)$$

where g_i is the distribution function of the velocity field. τ_g is the relaxation time and it is related to the kinematic viscosity as $\nu = c_s^2(\tau_g - 0.5)\delta t$. The equilibrium distribution function g_i^{eq} is designed as

$$g_i^{\text{eq}} = \begin{cases} \frac{\rho}{c_s^2}(\omega_0 - 1) + \rho_0 \omega_0 \left(\frac{\mathbf{e}_i \cdot \mathbf{u}}{c_s^2} + \frac{\mathbf{u} \cdot \mathbf{u} \cdot \mathbf{Q}}{2c_s^4} \right), & i = 0 \\ \frac{\rho}{c_s^2} \omega_i + \rho_0 \omega_i \left(\frac{\mathbf{e}_i \cdot \mathbf{u}}{c_s^2} + \frac{\mathbf{u} \cdot \mathbf{u} \cdot \mathbf{Q}}{2c_s^4} \right), & i = 1-8 \end{cases} \quad (13)$$

with second-order tensor $\mathbf{Q} = \mathbf{e}_i \mathbf{e}_i - c_s^2 \mathbf{I}$. \mathbf{I} is the identity tensor. G_i is a source term that usually includes the external force as

$$G_i = \left(1 - \frac{1}{2\tau_g} \right) \omega_i \frac{\mathbf{e}_i \cdot \mathbf{F}}{c_s^2}. \quad (14)$$

For the incompressible flows, the density ρ of fluid is usually approximated as a constant ρ_0 . And with the Boussinesq approximation, the temperature effect on the velocity field can be implemented through the buoyancy force $-\rho_0 \mathbf{g} \beta' (T - T_m)$. The moment conditions of this method are also given below:

$$\sum_i g_i = \sum_i g_i^{\text{eq}} = 0, \quad (15)$$

$$\sum_i \mathbf{e}_i g_i = \rho_0 \mathbf{u} - \frac{\delta t}{2} \mathbf{F}, \quad \sum_i \mathbf{e}_i g_i^{\text{eq}} = \rho_0 \mathbf{u}, \quad (16)$$

$$\sum_i \mathbf{e}_i \mathbf{e}_i g_i^{\text{eq}} = \rho_0 \mathbf{u} \mathbf{u} + p \mathbf{I}, \quad (17)$$

$$\sum_i g_i^{\text{neq}} = \sum_i \epsilon g_i^{(1)} = \sum_i \epsilon^2 g_i^{(2)} = 0, \quad (18)$$

$$\sum_i \mathbf{e}_i g_i^{\text{neq}} = \sum_i \epsilon \mathbf{e}_i g_i^{(1)} = -\frac{\delta t}{2} \mathbf{F}, \quad \sum_i \mathbf{e}_i \epsilon^2 g_i^{(2)} = 0, \quad (19)$$

$$\sum_i G_i = 0, \quad (20)$$

$$\sum_i \mathbf{e}_i G_i = \left(1 - \frac{1}{2\tau_g} \right) \mathbf{F}, \quad (21)$$

$$\Lambda = \sum_i \mathbf{e}_i \mathbf{e}_i G_i = 0, \quad (22)$$

$$\begin{aligned} \epsilon \Pi^{(1)} &= \sum_i \mathbf{e}_i \mathbf{e}_i \epsilon g_i^{(1)} \\ &= -\tau_g \delta t c_s^2 \rho_0 \{ \nabla(\mathbf{u}) + [\nabla(\mathbf{u})]^T + [\nabla \cdot (\mathbf{u})] \mathbf{I} \}, \end{aligned} \quad (23)$$

where $g_i^{\text{neq}} = g_i - g_i^{\text{eq}} = \epsilon g_i^{(1)} + \epsilon^2 g_i^{(2)}$ is the nonequilibrium distribution function and $\nabla = \epsilon \nabla_1$ is used in the Chapman-Enskog analysis.

C. The one-step simplified lattice Boltzmann method

To derive the present one-step simplified lattice Boltzmann method, we reconstruct the general evolution equation of the LBM by using the CE expansion in this subsection. The above LBM evolution equations (7) and (12) can be written in a general form as

$$\begin{aligned} & h_i(\mathbf{x} + \mathbf{e}_i \delta t, t + \delta t) - h_i(\mathbf{x}, t) \\ &= -\frac{1}{\tau} [h_i(\mathbf{x}, t) - h_i^{\text{eq}}(\mathbf{x}, t)] + \delta t F_i, \end{aligned} \quad (24)$$

where h_i represents f_i or g_i . τ is the relaxation time. $\delta t F_i$ is the source term and can be replaced by $\delta t G_i$. Notice that the source term can be set as zero for the cases without source term. For example, when h_i represents f_i , F_i will be set as zero. Next, we will analyze the above evolution equation by the Chapman-Enskog expansion which is given as follows:

$$h_i = h_i^{\text{eq}} + h_i^{\text{neq}} = h_i^{(0)} + \epsilon h_i^{(1)} + \epsilon^2 h_i^{(2)}, \quad (25)$$

$$\partial_t = \epsilon \partial_{t_1} + \epsilon^2 \partial_{t_2}, \quad (26)$$

$$\partial_{\mathbf{x}} = \epsilon \partial_{\mathbf{x}_1}, \quad (27)$$

$$\mathbf{F} = \epsilon \mathbf{F}_1, \quad F_i = \epsilon F_{1i}. \quad (28)$$

Substituting Eqs. (25)–(28) into Eq. (24) and applying the Taylor expansion, we have

$$O(\epsilon^0): \quad h_i^{(0)} = h_i^{\text{eq}}, \quad (29)$$

$$O(\epsilon^1): \quad D_{1i} h_i^{(0)} = -\frac{1}{\tau \delta t} h_i^{(1)} + F_{1i}, \quad (30)$$

$$\begin{aligned} O(\epsilon^2): \quad & \partial_{t_2} h_i^{(0)} + D_{1i} \left(1 - \frac{1}{2\tau} \right) h_i^{(1)} \\ &= -\frac{1}{\tau \delta t} h_i^{(2)} - \frac{\delta t}{2} D_{1i} F_{1i}. \end{aligned} \quad (31)$$

Notice that Eq. (30) is used to obtain Eq. (31) and D_{1i} denotes $\partial_{t_1} + \mathbf{e}_i \cdot \partial_{\mathbf{x}_1}$. With Eq. (30) $\times \epsilon$ + Eq. (31) $\times \epsilon^2$, we can obtain

$$D_i h_i^{\text{eq}} + \epsilon^2 D_{1i} \left(1 - \frac{1}{2\tau} \right) h_i^{(1)} = -\frac{1}{\tau \delta t} h_i^{\text{neq}} + F_i - 0.5 \delta t \epsilon^2 D_{1i} F_{1i}, \quad (32)$$

where D_i represents $\partial_t + \mathbf{e}_i \cdot \partial_{\mathbf{x}}$. With the Taylor expansion, $D_i h_i^{\text{eq}}$ can be written as

$$\delta t D_i h_i^{\text{eq}} = h_i^{\text{eq}}(\mathbf{x}, t) - h_i^{\text{eq}}(\mathbf{x} - \mathbf{e}_i \delta t, t - \delta t) + 0.5 \delta t^2 D_i^2 h_i^{\text{eq}} + O(\delta t^3). \tag{33}$$

Substituting Eq. (33) into (32) we have

$$\frac{h_i^{\text{eq}}(\mathbf{x}, t) - h_i^{\text{eq}}(\mathbf{x} - \mathbf{e}_i \delta t, t - \delta t)}{\delta t} + 0.5 \delta t D_i^2 h_i^{\text{eq}} + O(\delta t^2) + \epsilon^2 D_{1i} \left(1 - \frac{1}{2\tau}\right) h_i^{(1)} = -\frac{1}{\tau \delta t} h_i^{\text{neq}} + F_i - 0.5 \delta t \epsilon^2 D_{1i} F_{1i}. \tag{34}$$

The expression of $D_i^2 h_i^{\text{eq}}$ can be given from Eq. (32) as

$$D_i^2 h_i^{\text{eq}} = -\epsilon^2 D_{1i} \frac{1}{\tau \delta t} h_i^{(1)} + \epsilon^2 D_{1i} F_{1i}, \tag{35}$$

where terms like $\epsilon^3 D_{1i}^2 (1 - 1/2\tau) h_i^{(1)}$, $\epsilon^4 \partial_{t_2} D_{1i} (1 - 1/2\tau) h_i^{(1)}$, $-\epsilon^3 \partial_{t_2} h_i^{(1)} / \tau \delta t$, etc., are of $O(\epsilon^3)$ at least and are neglected. Substituting Eq. (35) into (34) yields

$$\frac{h_i^{\text{eq}}(\mathbf{x}, t) - h_i^{\text{eq}}(\mathbf{x} - \mathbf{e}_i \delta t, t - \delta t)}{\delta t} + \epsilon^2 D_{1i} \left(1 - \frac{1}{\tau}\right) h_i^{(1)} = -\frac{1}{\tau \delta t} h_i^{\text{neq}} + F_i - \delta t \epsilon^2 D_{1i} F_{1i} + O(\delta t^2). \tag{36}$$

Similarly, according to the Taylor expansion, we can obtain the following equations:

$$\epsilon^2 D_{1i} \left(1 - \frac{1}{\tau}\right) h_i^{(1)} = \epsilon \left[\left(1 - \frac{1}{\tau}\right) h_i^{(1)}(\mathbf{x}, t) - \left(1 - \frac{1}{\tau}\right) h_i^{(1)}(\mathbf{x} - \mathbf{e}_i \delta t, t - \delta t) + O(\delta t^3) \right] / \delta t, \tag{37}$$

$$\delta t \epsilon^2 D_{1i} F_{1i} = \epsilon [F_{1i}(\mathbf{x}, t) - F_{1i}(\mathbf{x} - \mathbf{e}_i \delta t, t - \delta t) + O(\delta t^2)]. \tag{38}$$

Notice that $0.5 \epsilon \delta t^2 D_i^2 (1 - \frac{1}{\tau}) h_i^{(1)}$, $\epsilon^3 \partial_{t_2} h_i^{(1)}$, and $\epsilon^3 \partial_{t_2} F_{1i}$ are in the order of $O(\epsilon^3)$ and neglected in Eqs. (37) and (38). Substituting Eqs. (37) and (38) into (36) gives

$$\begin{aligned} & \frac{h_i^{\text{eq}}(\mathbf{x}, t) - h_i^{\text{eq}}(\mathbf{x} - \mathbf{e}_i \delta t, t - \delta t)}{\delta t} + \epsilon \frac{\left(1 - \frac{1}{\tau}\right) h_i^{(1)}(\mathbf{x}, t) - \left(1 - \frac{1}{\tau}\right) h_i^{(1)}(\mathbf{x} - \mathbf{e}_i \delta t, t - \delta t)}{\delta t} \\ &= -\frac{1}{\tau \delta t} h_i^{\text{neq}}(\mathbf{x}, t) + F_i - \epsilon [F_{1i}(\mathbf{x}, t) - F_{1i}(\mathbf{x} - \mathbf{e}_i \delta t, t - \delta t)] + O(\delta t^2). \end{aligned} \tag{39}$$

And it can be rewritten as

$$\begin{aligned} & \frac{h_i^{\text{eq}}(\mathbf{x}, t) - h_i^{\text{eq}}(\mathbf{x} - \mathbf{e}_i \delta t, t - \delta t)}{\delta t} + \epsilon \frac{\left(1 - \frac{1}{\tau}\right) h_i^{(1)}(\mathbf{x}, t) - \left(1 - \frac{1}{\tau}\right) h_i^{(1)}(\mathbf{x} - \mathbf{e}_i \delta t, t - \delta t)}{\delta t} \\ &= -\frac{1}{\tau \delta t} h_i^{\text{neq}}(\mathbf{x}, t) + F_i(\mathbf{x} - \mathbf{e}_i \delta t, t - \delta t) + O(\delta t^2), \end{aligned} \tag{40}$$

which only has a truncation error of $O(\delta t^2)$.

Duo to the value and the moment conditions of $f_i^{(1)}(\mathbf{x} - \mathbf{e}_i \delta t, t - \delta t)$, $g_i^{(1)}(\mathbf{x} - \mathbf{e}_i \delta t, t - \delta t)$ are still unknown, we cannot directly take the zeroth-order or first-order moment of Eq. (40) to derive the macroscopic variables. To fix this, the parts of nonequilibrium distribution functions in the scale of $O(\epsilon)$ are constructed and named ‘‘the constructed nonequilibrium functions’’ in the following.

Considering that Eq. (7) has no source term, the source term F_{1i} should disappear in Eq. (30). According to this, $f_i^{(1)}$ can be given as

$$f_i^{(1)} = -\tau_f \delta t D_{1i} f_i^{\text{eq}} = -\tau_f \delta t (\partial_{t_1} + \mathbf{e}_i \cdot \nabla_1) f_i^{\text{eq}}. \tag{41}$$

With Eq. (8) we can obtain

$$\frac{\partial f_i^{\text{eq}}}{\partial t_1} = \omega_i \left[\frac{\partial T}{\partial t_1} + \frac{\mathbf{e}_i \cdot \partial(T\mathbf{u})}{c_s^2 \partial t_1} \right], \tag{42}$$

$$\mathbf{e}_i \cdot \nabla_1 f_i^{\text{eq}} = \omega_i \left[\mathbf{e}_i \cdot \nabla_1 T + \frac{1}{c_s^2} (\mathbf{e}_i \mathbf{e}_i) : \nabla_1(T\mathbf{u}) \right]. \tag{43}$$

In the meantime, taking the zeroth-order moment of Eq. (41) gives

$$\frac{\partial T}{\partial t_1} + \nabla_1 \cdot (T\mathbf{u}) = 0. \tag{44}$$

Substituting Eq. (44) into (42) we have

$$\frac{\partial f_i^{\text{eq}}}{\partial t_1} = \omega_i \left[-\nabla_1 \cdot (T\mathbf{u}) + \frac{\mathbf{e}_i \cdot \partial(T\mathbf{u})}{c_s^2 \partial t_1} \right]. \tag{45}$$

With Eqs. (43) and (45) substituted in Eq. (41), we have

$$f_i^{(1)} = -\tau_f \delta t \omega_i \left[-\nabla_1 \cdot (\mathbf{T}\mathbf{u}) + \frac{\mathbf{e}_i \cdot \partial(\mathbf{T}\mathbf{u})}{c_s^2 \partial t_1} + \mathbf{e}_i \cdot \nabla_1 T + \frac{1}{c_s^2} (\mathbf{e}_i \mathbf{e}_i) : \nabla_1 (\mathbf{T}\mathbf{u}) \right]. \quad (46)$$

Given the complexity of this equation, the simplification is presented as follows. First, through algebraic operation, we can derive $\sum_i \omega_i [-\nabla_1 \cdot (\mathbf{T}\mathbf{u}) + (\mathbf{e}_i \mathbf{e}_i) : \nabla_1 (\mathbf{T}\mathbf{u}) / c_s^2] = 0$ and $\sum_i \omega_i \mathbf{e}_i [-\nabla_1 \cdot (\mathbf{T}\mathbf{u}) + (\mathbf{e}_i \mathbf{e}_i) : \nabla_1 (\mathbf{T}\mathbf{u}) / c_s^2] = 0$. In the meanwhile, $\sum_i f_i^{(1)}$ and $\sum_i \mathbf{e}_i f_i^{(1)}$ are only required in the zeroth-order moment of Eq. (32). This means that $-\nabla_1 \cdot (\mathbf{T}\mathbf{u}) + (\mathbf{e}_i \mathbf{e}_i) : \nabla_1 (\mathbf{T}\mathbf{u}) / c_s^2$ can be neglected in Eq. (46). Second, we can derive $\sum_i \omega_i \mathbf{e}_i \cdot \partial_1 (\mathbf{T}\mathbf{u}) / c_s^2 = 0$ and $\partial_{x_1} \cdot \sum_i \omega_i [\mathbf{e}_i \mathbf{e}_i \cdot \partial_1 (\mathbf{T}\mathbf{u}) / c_s^2] = \partial_1 \nabla_1 \cdot (\mathbf{T}\mathbf{u})$ which are related to $D_{1i} f_i^{(1)}$ in the zeroth-order moment of Eq. (32). Considering $\partial_1 \nabla_1 \cdot (\mathbf{T}\mathbf{u})$ is a truncation error in the Chapman-Enskog analysis when recovering the original LBM to Eq. (4), we do not recommend to remain $[\mathbf{e}_i \cdot \partial(\mathbf{T}\mathbf{u})] / (c_s^2 \partial t_1)$ in Eq. (46). To sum up, the constructed nonequilibrium function for the temperature field $f_i^{\text{neq},*}$ can be finally given as

$$f_i^{\text{neq},*} = \epsilon f_i^{(1)} = -\tau_f \delta t \omega_i \mathbf{e}_i \cdot \nabla T. \quad (47)$$

Here we can see that, through algebraic operation, both of Eqs. (46) and (47) satisfy the moment condition of $\sum_i \epsilon f_i^{(1)} = 0$.

Similarly, according to Eq. (30), $g_i^{(1)}$ can be given as

$$g_i^{(1)} = -\tau_g \delta t (\partial t_1 + \mathbf{e}_i \cdot \nabla_1) g_i^{\text{eq}} + \tau_g \delta t G_{1i}. \quad (48)$$

With Eq. (13), we can obtain

$$\frac{\partial g_i^{\text{eq}}}{\partial t_1} = \begin{cases} \frac{\omega_0 - 1}{c_s^2} \frac{\partial p}{\partial t_1} - \omega_0 \rho_0 \frac{\mathbf{I} : \partial(\mathbf{u}\mathbf{u})}{2c_s^2 \partial t_1}, & i = 0 \\ \frac{\omega_i}{c_s^2} \frac{\partial p}{\partial t_1} + \omega_i \rho_0 \left[\frac{\mathbf{e}_i \cdot \partial \mathbf{u}}{c_s^2 \partial t_1} + \frac{\mathbf{Q} : \partial(\mathbf{u}\mathbf{u})}{2c_s^4 \partial t_1} \right], & i = 1-8 \end{cases} \quad (49)$$

$$\mathbf{e}_i \cdot \nabla_1 g_i^{\text{eq}} = \begin{cases} 0, & i = 0 \\ \frac{\omega_i}{c_s^2} \mathbf{e}_i \cdot \nabla_1 p + \omega_i \rho_0 \left\{ \frac{\mathbf{e}_i \cdot \nabla_1 \mathbf{u}}{c_s^2} + \frac{\mathbf{Q} : [\mathbf{e}_i \cdot \nabla_1 (\mathbf{u}\mathbf{u})]}{2c_s^4} \right\}, & i = 1-8 \end{cases} \quad (50)$$

where $\mathbf{e}_0 = \binom{0}{0}$ is considered. In the original LBM, to recover the incompressible NS equations, the terms of $O(\delta t \text{Ma}^2)$ are neglected in the analysis on $g_i^{(1)}$, where $\frac{\partial p}{\partial t_1}$ and $\mathbf{u}\mathbf{u}$ are supposed to be of $O(\text{Ma}^2)$ under the incompressible condition [30]. In the meantime, taking the first-order moment of Eq. (48) gives

$$\rho_0 \frac{\partial \mathbf{u}}{\partial t_1} + \rho_0 \nabla_1 \cdot (\mathbf{u}\mathbf{u}) + \nabla_1 p = \mathbf{F}_1. \quad (51)$$

With the above condition of $O(\delta t \text{Ma}^2)$ and Eq. (51), we can rewrite Eqs. (49) and (50) as

$$\frac{\partial g_i^{\text{eq}}}{\partial t_1} = \begin{cases} 0, & i = 0 \\ \frac{\omega_i}{c_s^2} [\mathbf{e}_i \cdot (\mathbf{F}_1 - \nabla_1 p)], & i = 1-8 \end{cases} \quad (52)$$

$$\mathbf{e}_i \cdot \nabla_1 g_i^{\text{eq}} = \begin{cases} 0, & i = 0 \\ \frac{\omega_i}{c_s^2} [\mathbf{e}_i \cdot \nabla_1 p + \rho_0 \mathbf{e}_i \mathbf{e}_i : \nabla_1 \mathbf{u}], & i = 1-8. \end{cases} \quad (53)$$

With Eqs. (48), (52), and (53), the constructed nonequilibrium function $g_i^{\text{neq},*}$ for the velocity field is given as

$$g_i^{\text{neq},*} = \epsilon g_i^{(1)} = -\tau_g \delta t \frac{\omega_i}{c_s^2} [\mathbf{e}_i \cdot \mathbf{F} + \rho_0 \mathbf{e}_i \mathbf{e}_i : \nabla \mathbf{u}] + \tau_g \delta t G_i. \quad (54)$$

Through algebra operation, it is not difficult to prove that the constructed nonequilibrium function (54) satisfies the moment conditions of $\sum_i g_i^{(1)} = 0$, $\sum_i \mathbf{e}_i g_i^{(1)} = -\frac{\delta t}{2} \mathbf{F}_1$, and $\sum_i \mathbf{e}_i \mathbf{e}_i g_i^{(1)} = \Pi^{(1)}$.

According to the zeroth moment of Eq. (40) where $\sum_i F_i(\mathbf{x} - \mathbf{e}_i \delta t, t - \delta t)$ should be deleted, and considering the moment conditions in Eqs. (9) and (10), we can easily derive the present one-step simplified method of the temperature field as

$$T(\mathbf{x}, t) = \sum_i \left[f_i^{\text{eq}}(\mathbf{x} - \mathbf{e}_i \delta t, t - \delta t) + \left(1 - \frac{1}{\tau_f} \right) \times f_i^{\text{neq},*}(\mathbf{x} - \mathbf{e}_i \delta t, t - \delta t) \right], \quad (55)$$

where $f_i^{\text{neq},*}$ is given in Eq. (47). Similarly according to the first moment of Eq. (40) and moment conditions in Eqs. (16) and (19), the present one-step simplified method of the velocity field is derived as

$$\rho_0 \mathbf{u}(\mathbf{x}, t) = \sum_i \mathbf{e}_i \left\{ g_i^{\text{eq}}(\mathbf{x} - \mathbf{e}_i \delta t, t - \delta t) + \left(1 - \frac{1}{\tau_g} \right) \times g_i^{\text{neq},*}(\mathbf{x} - \mathbf{e}_i \delta t, t - \delta t) + \delta t G_i(\mathbf{x} - \mathbf{e}_i \delta t, t - \delta t) \right\} + \frac{\delta t}{2} \mathbf{F}(\mathbf{x}, t). \quad (56)$$

In the original LBM, the pressure p can be evaluated as

$$p = \frac{c_s^2}{1 - \omega_0} \left(\sum_{i \neq 0} g_i^{\text{eq}} - \rho_0 \omega_0 \frac{\mathbf{u} \cdot \mathbf{u}}{2c_s^2} \right), \quad (57)$$

$$p = \frac{c_s^2}{1 - \omega_0} \left(\sum_{i \neq 0} g_i - \rho_0 \omega_0 \frac{\mathbf{u} \cdot \mathbf{u}}{2c_s^2} \right), \quad (58)$$

from which we can easily derive

$$\sum_{i \neq 0} g_i^{\text{neq}} = \sum_{i \neq 0} g_i - \sum_{i \neq 0} g_i^{\text{eq}} = 0. \quad (59)$$

According to Eqs. (40), (57), and (59), the pressure p can be calculated as

$$p(\mathbf{x}, t) = \frac{c_s^2}{1 - \omega_0} \left\{ \sum_{i \neq 0} g_i^{\text{eq}}(\mathbf{x} - \mathbf{e}_i \delta t, t - \delta t) + \sum_{i \neq 0} \left[1 - \frac{1}{\tau_g} \right] g_i^{\text{neq},*}(\mathbf{x} - \mathbf{e}_i \delta t, t - \delta t) \right\}$$

$$+ \sum_{i \neq 0} \delta t G_i(\mathbf{x} - \mathbf{e}_i \delta t, t - \delta t) - \rho_0 \omega_0 \frac{\mathbf{u}(\mathbf{x}, t) \cdot \mathbf{u}(\mathbf{x}, t)}{2c_s^2} \Bigg\}. \quad (60)$$

To compute the space derivatives used in the above equations, the following second-order isotropic discretization schemes [30] are adopted as

$$\nabla \varphi(\mathbf{x}) = \sum_{i \neq 0} \frac{\omega_i \mathbf{e}_i \varphi(\mathbf{x} + \mathbf{e}_i \delta t)}{c_s^2 \delta t} \quad (61)$$

and

$$\nabla^2 \varphi(\mathbf{x}) = \sum_{i \neq 0} \frac{2\omega_i [\varphi(\mathbf{x} + \mathbf{e}_i \delta t) - \varphi(\mathbf{x})]}{c_s^2 \delta t^2}, \quad (62)$$

where φ can be replaced by any other macroscopic variables.

Finally, the computational procedure of the present one-step simplified lattice Boltzmann method is briefly given as follows.

- (1) Initialize the temperature and velocity field.
- (2) Update the temperature T and calculate the source term according to Eqs. (55) and (14), respectively.
- (3) Calculate the buoyancy force.
- (4) Update velocity and pressure by Eqs. (56) and (60).
- (5) Implement appropriate boundary conditions.
- (6) Return to step 2 until the computation satisfies demand.

D. Boundary treatment

Given that the spatial derivatives are required to calculate the constructed nonequilibrium functions $f_i^{\text{neq},*}$, $g_i^{\text{neq},*}$ in Eqs. (55), (56), and (60), the nonequilibrium parts on the boundaries are still unknown. Here are two choices to solve this problem.

For the first one, we can adopt a linear extrapolation scheme [18] for the nonequilibrium parts on the boundaries, namely, setting

$$f_i^{\text{neq},*}(\mathbf{x}_b) = 2f_i^{\text{neq},*}(\mathbf{x}_{f1}) - f_i^{\text{neq},*}(\mathbf{x}_{f2}) \quad (63)$$

and

$$g_i^{\text{neq},*}(\mathbf{x}_b) = 2g_i^{\text{neq},*}(\mathbf{x}_{f1}) - g_i^{\text{neq},*}(\mathbf{x}_{f2}), \quad (64)$$

where \mathbf{x}_b represents the location of boundary. \mathbf{x}_{f1} and \mathbf{x}_{f2} represent the locations of the first and second layers of the inner domain, respectively.

For the second one, we can directly set the spatial derivatives' value according to the boundary conditions and

calculate the unknown spatial derivatives through the Taylor expansion. Taking the bottom wall of internal flows as an example, if the no-slip condition is assumed, we can set $-\partial_y u_y = \partial_x u_x = \partial_x u_y = 0$, where $\partial_y u_y = -\partial_x u_x$ satisfies the divergence-free velocity condition of the incompressible flow. The unknown spatial derivatives can be calculated as

$$\partial_y u_x = \frac{4u_x(\mathbf{x}_{f1}) - 3u_{xw} - u_x(\mathbf{x}_{f2})}{2\delta x}, \quad (65)$$

where u_{xw} is the horizontal velocity of the wall.

E. Computational efficiency and numerical stability

For the traditional numerical methods to simulate incompressible viscous flows, it can be classified into two categories as strictly incompressible methods and weakly compressible methods. One of the strictly incompressible methods is the vorticity-stream function method, in which vorticity and stream functions are introduced into the incompressible Navier-Stokes (NS) equations to remove the pressure term in the NS equations. The application of this method is restricted due to the cost in solving the Poisson equation of the stream function and the boundary condition of the vorticity function. Another approach of the strictly incompressible methods is directly solving the incompressible NS equations. Due to that the pressure can not be explicitly updated in the momentum equation, it has to additionally solve the Poisson equation of the pressure. The representative methods of this approach are projection method [31], semi-implicit method for the pressure-linked equation (SIMPLE) [32], pressure implicit with split operator (PISO) [33], etc. In the weakly compressible method, namely, the artificial compressible method (ACM) [34], it assumes that the effect of compressibility can be ignored under the condition of Mach number $Ma < 0.3$. So, the term of weak compressibility $\delta p / \partial t$, where $\delta = \rho / p$, is added into the continuity equation of the incompressible NS equations. Then the pressure can be explicitly updated in the algorithm. Compared to the strictly incompressible methods, due to the slow convergence of the Poisson equation, the algorithm of the ACM is more simple and efficient. But owing to the compressibility, the pressure wave brings a challenge to the stability of numerical simulations [35].

Through the CE expansion analysis, the incompressible LB model used in this paper neglects the term $\delta p / \partial t$ in the CE expansion of recovering the continuity equation, which means the incompressible LBM and the present OSLBM are in the category of the artificial compressible method (without other specified, we use "ACM" to represent the traditional artificial compressible method in the following). Given the fact that the LBM and the OSLBM can be considered as artificial compressible methods, it is valuable to compare the present method with the ACM in terms of computational efficiency and numerical stability. The governing equations and finite difference schemes of the ACM in two dimensions are given below:

$$\frac{\partial u_x}{\partial t} + \frac{\partial u_x^2}{\partial x} + \frac{\partial (u_x u_y)}{\partial y} = -\frac{\partial p}{\partial x} + \nu \left(\frac{\partial^2 u_x}{\partial x^2} + \frac{\partial^2 u_x}{\partial y^2} \right), \quad (66)$$

$$\frac{\partial u_y}{\partial t} + \frac{\partial u_y^2}{\partial y} + \frac{\partial (u_x u_y)}{\partial x} = -\frac{\partial p}{\partial y} + \nu \left(\frac{\partial^2 u_y}{\partial x^2} + \frac{\partial^2 u_y}{\partial y^2} \right), \quad (67)$$

$$\delta \frac{\partial p}{\partial t} + \frac{\partial u_x}{\partial x} + \frac{\partial u_y}{\partial y} = 0, \quad (68)$$

$$\frac{\partial u_x}{\partial t} = \frac{u_x^{n+1} - u_x^n}{\delta t}, \quad \frac{\partial u_y}{\partial t} = \frac{u_y^{n+1} - u_y^n}{\delta t}, \quad \frac{\partial p}{\partial t} = \frac{p^{n+1} - p^n}{\delta t}, \quad (69)$$

$$\frac{\partial u_{x,ij}^2}{\partial x} = \frac{(u_{x,ij} + u_{x,i+1j})(u_{x,ij} + u_{x,i+1j}) - (u_{x,ij} + u_{x,i-1j})(u_{x,ij} + u_{x,i-1j})}{4\delta x}, \quad (70)$$

$$\frac{\partial u_{y,ij}^2}{\partial y} = \frac{(u_{y,ij} + u_{y,ij+1})(u_{y,ij} + u_{y,ij+1}) - (u_{y,ij} + u_{y,ij-1})(u_{y,ij} + u_{y,ij-1})}{4\delta y}, \quad (71)$$

$$\frac{\partial (u_{x,ij}u_{y,ij})}{\partial y} = \frac{(u_{x,ij} + u_{x,i+1j})(u_{y,ij} + u_{y,ij+1}) - (u_{x,ij} + u_{x,i-1j})(u_{y,ij-1} + u_{y,ij+1})}{4\delta y}, \quad (72)$$

$$\frac{\partial (u_{x,ij}u_{y,ij})}{\partial x} = \frac{(u_{x,ij} + u_{x,i+1j})(u_{y,ij} + u_{y,ij+1}) - (u_{x,i-1j} + u_{x,i-1j+1})(u_{y,i-1j} + u_{y,ij})}{4\delta x}, \quad (73)$$

$$\nabla^2 u_{x,ij} = \frac{u_{x,i+1j} - 2u_{x,ij} + u_{x,i-1j}}{(\delta x)^2} + \frac{u_{x,ij+1} - 2u_{x,ij} + u_{x,ij-1}}{(\delta y)^2}, \quad (74)$$

$$\nabla^2 u_{y,ij} = \frac{u_{y,i+1j} - 2u_{y,ij} + u_{y,i-1j}}{(\delta x)^2} + \frac{u_{y,ij+1} - 2u_{y,ij} + u_{y,ij-1}}{(\delta y)^2}, \quad (75)$$

$$\frac{\partial u_{x,ij}}{\partial x} = \frac{u_{x,ij} - u_{x,i-1j}}{\delta x}, \quad \frac{\partial u_{y,ij}}{\partial y} = \frac{u_{y,ij} - u_{y,ij-1}}{\delta y}, \quad (76)$$

where the superscript n represents the n th step of the iteration while the subscripts i and j represent the location of the grid point. The staggered grid is used in the ACM within this paper.

In the following, the typical lid-driven cavity [36] is chosen as a benchmark, in which the length of the square is set as $L_0 = 1$ and the horizontal speed of the top lid is $u_0 = 0.1$. In this simulation, no-slip boundary conditions are applied in all the methods and the typical nonequilibrium extrapolation scheme [37] is applied in the LBM to treat the boundaries of the distribution function, and δ is set as $\frac{1}{3}$ in the ACM.

To learn the computational efficiency of the OSLBM and ACM, a set of meshes 50×50 , 100×100 , 200×200 , 400×400 are adopted to simulate the lid-driven cavity at Reynolds number $\text{Re} = u_0 L_0 / \nu = 5000$. The results are given in Fig. 2, where t is the time cost of the computation and t^* is the physical time of the flow process. Notice that the time step δt is equal to the space step δx in the OSLBM due to the LB model, while the time step can be modified in the ACM. But the time step is usually much smaller than the space step to ensure numerical stability. In the cases of 50×50 and 100×100 , setting $\delta t = 0.1\delta x$ leads to divergence. So the time cost of ACM with $\delta t = 0.1\delta x$ is not plotted in Figs. 2(a) and 2(b). From Fig. 2, we can see that, by simulating the same length time of the flow process, the computation efficiency of the ACM depends on the time step. When $\delta t = 0.1\delta x$, compared to the OSLBM, ACM can save 30%–31% computational time where the percentage is not sensitive to mesh size. For $\delta t = 0.05\delta x$ and $0.01\delta x$, the ACM will cost more computational time about 41% and 600%, respectively, in our tests. As for the LBM and the SLBM, our previous work [38] has shown that, compared to the OSLBM, the LBM can save 40%–50% computational time. While the improved efficiency of the OSLBM, compared to the SLBM, is 25%–35%. The results of numerical stability are given in Tables I and II where

the convergent criterion is set as

$$\text{Error}_v = \frac{\sum_{N_x \times N_y} |(\sqrt{u_x^2 + u_y^2})^{n+1} - (\sqrt{u_x^2 + u_y^2})^n|}{\sum_{N_x \times N_y} (\sqrt{u_x^2 + u_y^2})^{n+1}} < 10^{-6}. \quad (77)$$

We can see that, when $\delta t = \delta x$, all the cases are divergent for the ACM. And through reducing the time step, the problems of stability can be alleviated. While both of the SLBM and the OSLBM have good performance on numerical stability even for relaxation time τ close to 0.5.

III. NUMERICAL TESTS

In this section, several typical numerical examples, namely, the natural convection in a square cavity, the porous plate problem, and the natural convection in a concentric annulus, are conducted to validate the accuracy of the present method. Unless otherwise specified, the convergent criterion is set as

$$\text{Error}_v = \frac{\sum_{N_x \times N_y} |(\sqrt{u_x^2 + u_y^2})^{n+1} - (\sqrt{u_x^2 + u_y^2})^n|}{\sum_{N_x \times N_y} (\sqrt{u_x^2 + u_y^2})^{n+1}} < 10^{-8}, \quad (78)$$

$$\text{Error}_T = \frac{\sum_{N_x \times N_y} |T^{n+1} - T^n|}{\sum_{N_x \times N_y} |T^{n+1}|} < 10^{-8}, \quad (79)$$

where the superscript n represents the n th step of the iteration.

A. Natural convection in a square cavity

As depicted in Fig. 3, the physical configuration is the phenomenon of heat convection induced by the temperature difference between the left and right walls. No-slip boundary conditions are applied to all the fixed walls. The left wall is set as a hot wall with temperature $T_H = 1$ while the right one is a

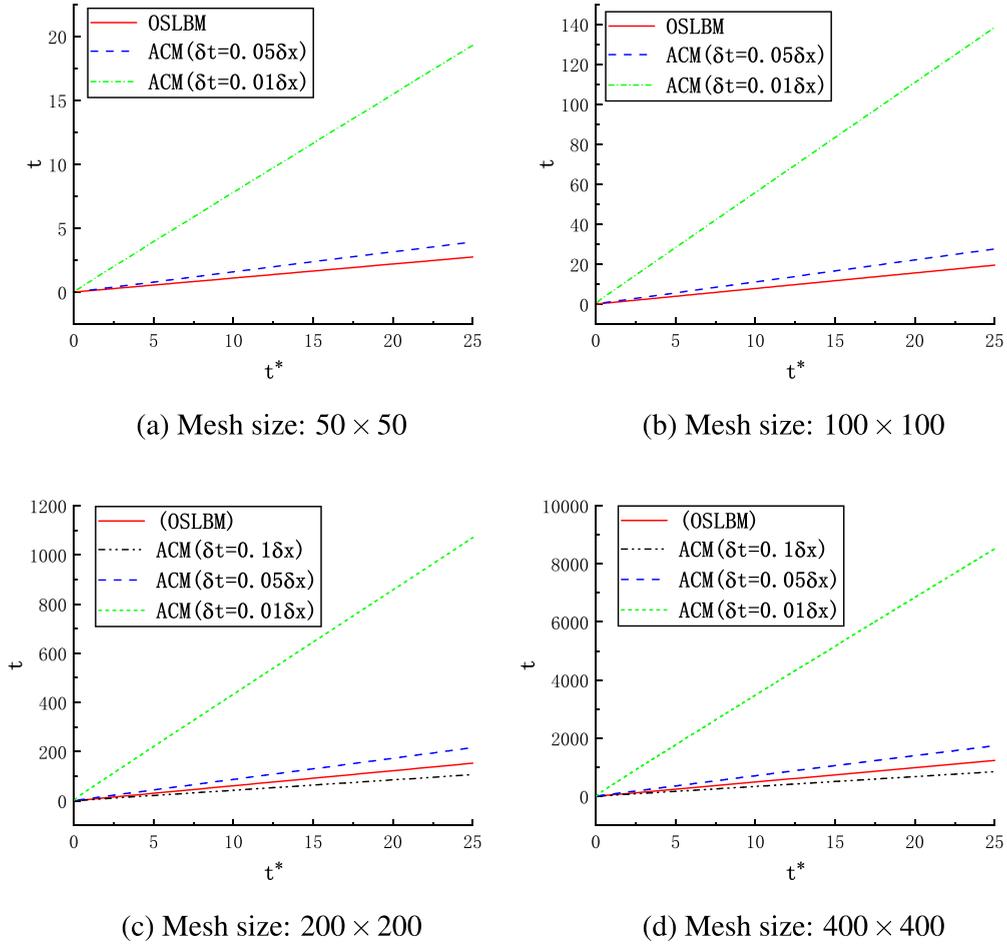


FIG. 2. Computational efficiency comparisons between the OSLBM and the ACM with different mesh sizes at $Re = 5000$. t is the time cost of the computation while t^* is the physical time of the flow process. δt is the time step and δx is the space step of the ACM

cold wall with temperature $T_c = 0$. There is no heat flux across the top and bottom walls. The length of the square cavity is set as $L = 1$. Usually, this convection flow depends on two crucial nondimensional numbers, namely, the Prandtl number Pr and

Rayleigh number Ra defined as

$$Pr = \frac{\nu}{\chi}, \tag{80}$$

TABLE I. Numerical stability tests for ACM with different mesh sizes. “c” represents the convergence while “d” represents divergence, and “-” represents the results that are not directly divergent but do not meet the convergent criterion.

Re	Mesh size	ACM($\delta t = \delta x$)	ACM($\delta t = 0.1\delta x$)	ACM($\delta t = 0.05\delta x$)	ACM($\delta t = 0.01\delta x$)
3200	10 × 10	d	d	d	c
	50 × 50	d	d	c	c
	100 × 100	d	d	c	c
	150 × 150	d	c	c	c
	200 × 200	d	c	c	c
	300 × 300	d	c	c	c
5000	10 × 10	d	d	d	c
	50 × 50	d	d	-	c
	100 × 100	d	d	c	c
	150 × 150	d	d	c	c
	200 × 200	d	c	c	c
	300 × 300	d	c	c	c

TABLE II. Numerical stability tests for LBM, SLBM, and OSLBM with different mesh sizes. ‘‘c’’ represents the convergence while ‘‘d’’ represents divergence, and ‘‘-’’ represents the results that are not directly divergent but do not meet the convergent criterion.

Re	Mesh size	LBM	SLBM	OSLBM
3200	$10 \times 10(\tau = 0.5009)$	d	c	c
	$50 \times 50(\tau = 0.505)$	-	c	c
	$100 \times 100(\tau = 0.509)$	-	c	c
	$150 \times 150(\tau = 0.514)$	-	c	c
	$200 \times 200(\tau = 0.519)$	-	c	c
5000	$300 \times 300(\tau = 0.528)$	c	c	c
	$10 \times 10(\tau = 0.5006)$	d	c	c
	$50 \times 50(\tau = 0.503)$	-	c	c
	$100 \times 100(\tau = 0.506)$	-	c	c
	$150 \times 150(\tau = 0.509)$	-	c	c
	$200 \times 200(\tau = 0.512)$	-	c	c
	$300 \times 300(\tau = 0.518)$	c	c	c

$$Ra = \frac{\beta' g \Delta T L^3}{\nu \chi} = \frac{V_c^2 L^2}{\nu \chi}, \quad (81)$$

where g is the magnitude of the gravity acceleration and $\Delta T = T_H - T_C$ is the temperature difference. $V_c = \sqrt{\beta' g L \Delta T}$ is the characteristic velocity. In this simulation, the constant parameters are set as $g = 9.8$, $Pr = 0.71$, $V_c = 0.1$.

Before the accuracy tests, a stability test of high Rayleigh number $Ra = 10^8$ with a mesh size of 10×10 is conducted, in which $\tau_f = 0.5003$ and $\tau_g = 0.5004$ are unstable for the single-relaxation-time LBM. However, the present numerical simulation is convergent which indicates that the present method is quite effective and stable at relaxation time close to 0.5.

Next, a series of numerical tests with different Rayleigh numbers at $Ra = 10^3, 10^4, 10^5, 10^6$ with mesh sizes of $100 \times 100, 150 \times 150, 200 \times 200, 300 \times 300$, respectively, are performed to validate the accuracy.

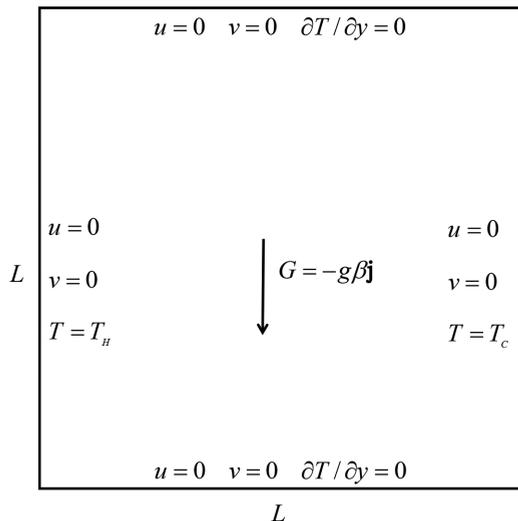


FIG. 3. The illustration of the natural convection in a square cavity.

TABLE III. Comparison of numerical velocities from different methods.

Ra		10^3	10^4	10^5	10^6
u_{max}^*	Present	3.645	16.181	34.911	65.334
	Chen [26]	3.648	16.137	34.566	64.173
	TLBFS [39]	3.640	16.140	34.870	64.838
	DQ [40]	3.649	16.190	34.736	64.755
	Davis [40]	3.634	16.182	34.810	65.330
y	Present	0.810	0.820	0.855	0.850
	Chen [26]	0.810	0.820	0.855	0.850
	TLBF [39]	0.815	0.825	0.855	0.850
	DQ [40]	0.815	0.825	0.855	0.850
	Davis [41]	0.813	0.823	0.855	0.851
v_{max}^*	Present	3.690	19.611	68.668	221.335
	Chen [26]	3.699	19.594	68.351	218.560
	TLBFS [39]	3.708	19.670	68.850	220.920
	DQ [40]	3.698	19.638	68.640	220.640
	Davis [41]	3.679	19.509	68.220	216.750
x	Present	0.180	0.120	0.065	0.037
	Chen [26]	0.180	0.120	0.065	0.040
	TLBFS [39]	0.180	0.118	0.065	0.038
	DQ [40]	0.180	0.120	0.065	0.035
	Davis [41]	0.179	0.120	0.066	0.039

Figure 4 gives the numerical results of the isotherms at different Rayleigh numbers. At small Rayleigh numbers, the isotherms are almost vertical, indicating that the heat transfer is mainly caused by the conduction between the hot and cold walls. While with the increase of Ra , more areas of vertical isotherms are occupied by the horizontal ones which demonstrates that heat convection starts to take the lead. Figure 5 exhibits the numerical results of streamlines. As it can be seen that a single vortex is formed when the Ra is small. Then the vortex is divided into two centrosymmetric vortices at $Ra = 10^5$. When Ra continues growing, a third vortex shows up squeezing the other ones towards the wall. These observations are in accordance with the physical phenomenon and the reported results [26,42].

To better quantify the performance of the present method, the maximum horizontal velocity $u_{max}^* = Lu_{max}/\chi$ with its location y along the vertical center line, and the maximum vertical velocity $v_{max}^* = Lv_{max}/\chi$ with its location x along the horizontal center line are listed and compared in Table III. Another important standard of estimating the numerical results is the convective rate of heat transfer described by the Nusselt number Nu . Usually the average Nusselt number \overline{Nu} of the whole computational domain is defined as

$$\overline{Nu} = \frac{L}{\chi \Delta T} \frac{1}{L^2} \iint_{\Omega} \left(u_x T - \chi \frac{\partial T}{\partial x} \right) d\Omega. \quad (82)$$

The local and average Nusselt numbers Nu and Nu_0 on the left wall are defined as

$$Nu = \frac{L}{\Delta T} \frac{\partial T}{\partial x}, \quad (83)$$

$$Nu_0 = \frac{1}{L \Delta T} \int_{x=0} \frac{\partial T}{\partial x} dy. \quad (84)$$

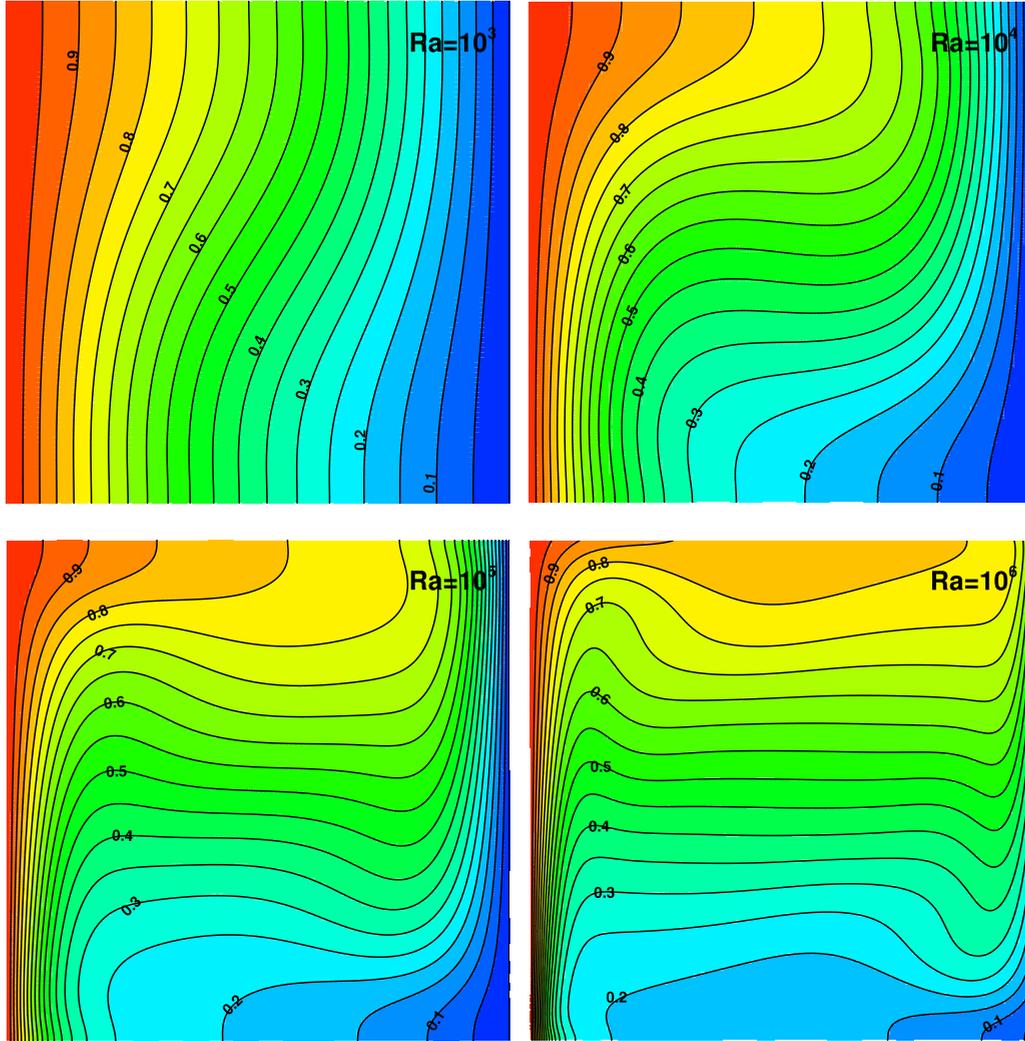


FIG. 4. Isotherms of the natural convection in a square cavity at different Ra.

Table IV gives the numerical results of average Nusselt numbers, the maximum and minimum values of the local Nusselt numbers with their corresponding positions, and compares them with available literature. All the results are in great accordance with reported values, implying that the present method performs well in accuracy.

B. The porous plate problem

The porous plate problem describes a phenomenon that fluids are sheared by two horizontal porous plates of different temperatures while another flow in the vertical direction is injected through the plates and leaves the domain with the initial rate. As illustrated in Fig. 6, the upper porous plate in cold temperature $T_c = 0$ is moving with a constant velocity $u_0 = 0.1$ in the horizontal direction, while the lower one in a higher temperature $T_H = 1$ is fixed in the bottom. The inject rate of the vertical flows is v_0 determined by the Reynolds number $Re = v_0 H / \nu$. $H = 1$ is the distance between the porous plates and $L = 2H$ is the length of the channel. To simulate this problem, the top and bottom boundaries of the fluid domain are corresponding to the exact velocity and temperature while the periodic boundary conditions are applied in the horizontal

direction. In the beginning, setting $Ra = 100$, $\tau_g = 1.2$, a series of numerical tests with $Re = 5, 20, 30$ at $Pr = 0.71$, and $Pr = 0.2, 0.8, 1.5$ at $Re = 10$ are conducted to examine the accuracy of the present method. Figure 7 displays the numerical horizontal velocity distributions and temperature distributions along the vertical direction of the channel, where solid lines represent the corresponding analytical solutions [44] given as

$$u = u_0 \left(\frac{e^{Re y/H} - 1}{e^{Re} - 1} \right), \tag{85}$$

$$T = T_H - \Delta T \left(\frac{e^{Pr Re y/H} - 1}{e^{Pr Re} - 1} \right). \tag{86}$$

As can be observed, all the numerical results fit the analytical solutions well. To further demonstrate it, the relative errors of different meshes at $Pr = 0.71$, $Re = 10$, $Ra = 100$ are drawn in Fig. 8, where the dotted lines are linear fitting lines. As we can see, the results have good linearity and the slopes of these fitting lines are around 2 implying the present method has second-order accuracy in space.

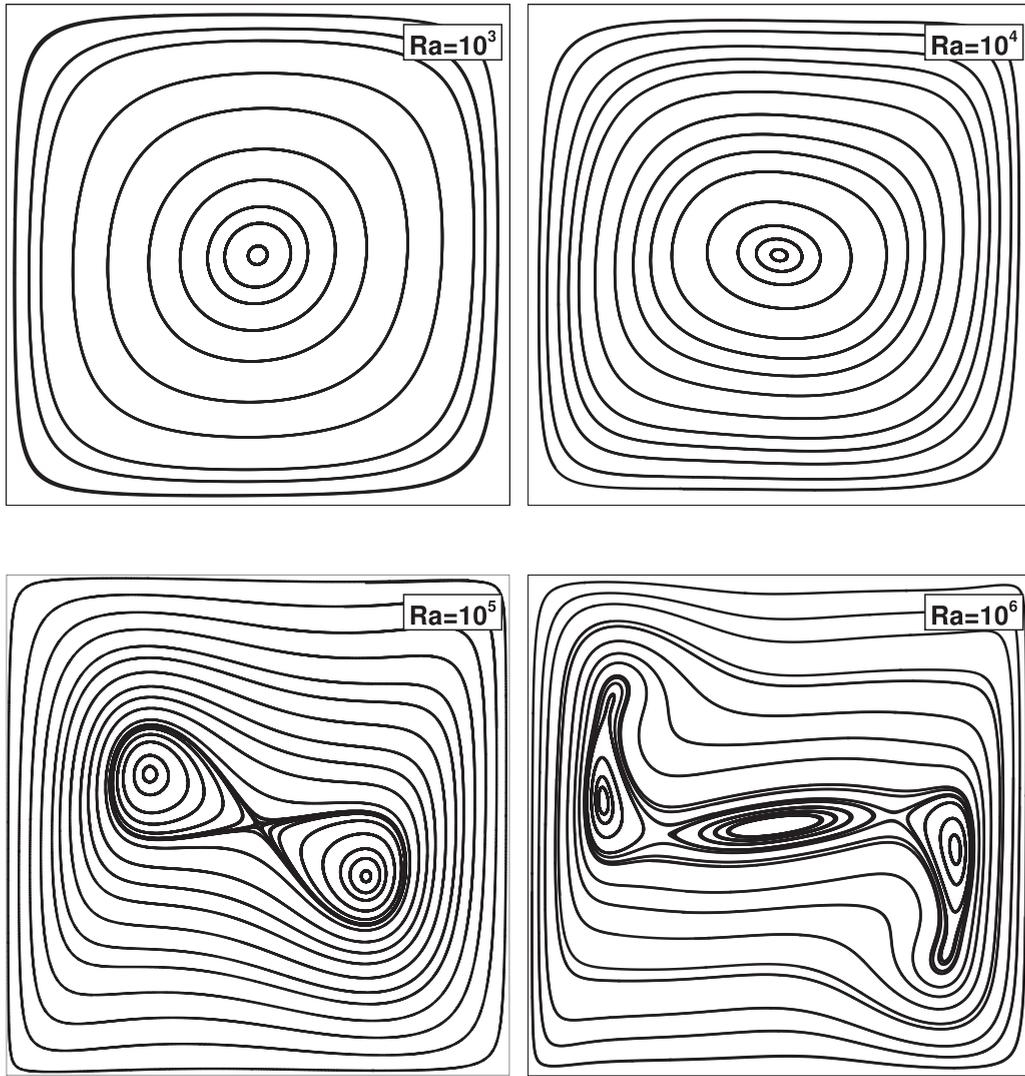


FIG. 5. The streamlines of the natural convection in a square cavity at different Ra.

C. Natural convection in a concentric annulus

In the previous numerical examples, simulations are carried on uniform meshes with straight boundaries, while it is also valuable to investigate the performance on curved

boundaries and nonuniform meshes. In the following, the natural convection in a concentric annulus is simulated to assess the performance of the present method. As illustrated in Fig. 9, the concentric annulus consists of two circular walls, where the inner one is a cold wall with temperature $T_c = 0$ and the higher temperature of the outer circle is $T_H = 1.0$. $R_i = 1$ is the inner radius and R_o is the radius of the outer circle. Related to the Rayleigh number, the characteristic length $L = R_o - R_i$ is the distance between the walls. $Ar = R_o/R_i = 2.6$ is the ratio of the radius.

In this simulation, a series of cases at $Ra = 10^2, 10^3, 3 \times 10^3, 6 \times 10^3, 10^4, 5 \times 10^4$ are carried on a 250×60 body-fitted O-type mesh [45] which is depicted in Fig. 10, and all the cases apply the no-slip boundary conditions on the circular walls. The rest of parameters are fixed at $Pr = 0.71, g = 9.8, v_c = 0.1$.

The convergent results of isotherms of the present simulation are given in Fig. 11, where the distributions of the isotherms are almost uniform and equidistant between the walls when the Rayleigh numbers are small. This can be attributed to the heat conduction overwhelming the convection.

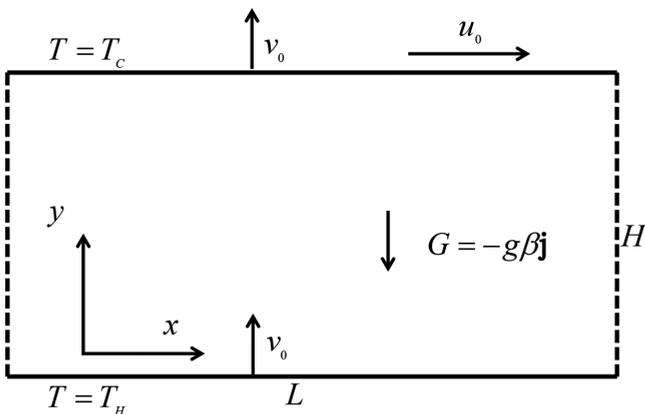


FIG. 6. The illustration of the porous plate problem.

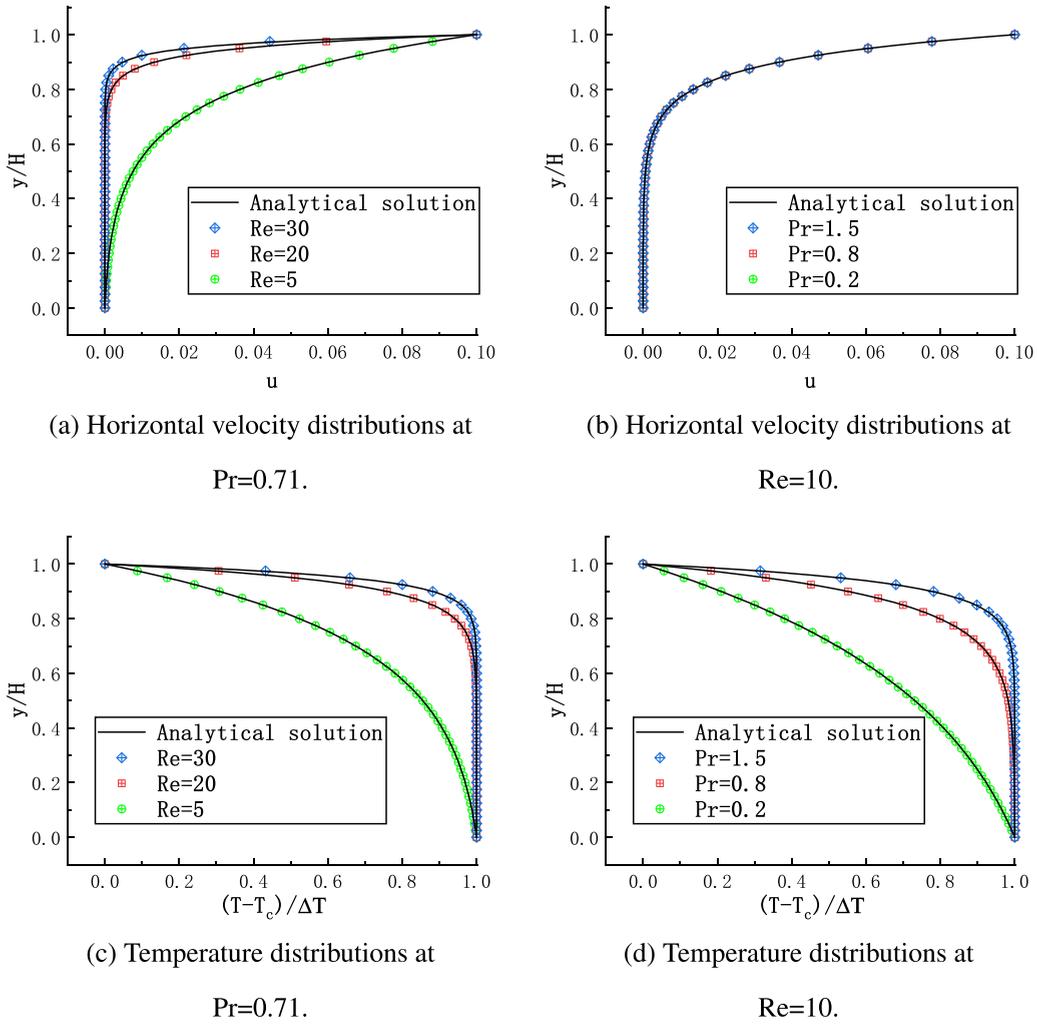


FIG. 7. Comparison of numerical results and analytical values.

Then, with the increment of the Rayleigh number, the heat convection will be the main factor of the temperature distribution, which leads to the nonuniform distributions of the isotherms drawn in Fig. 11 at large Ra. The above

observation of the simulation results is corresponding to the physics phenomenon. To better quantify this, Table V lists the heat conductivities of the inner and outer walls at different Ra, and compares the results with the reported values, where the

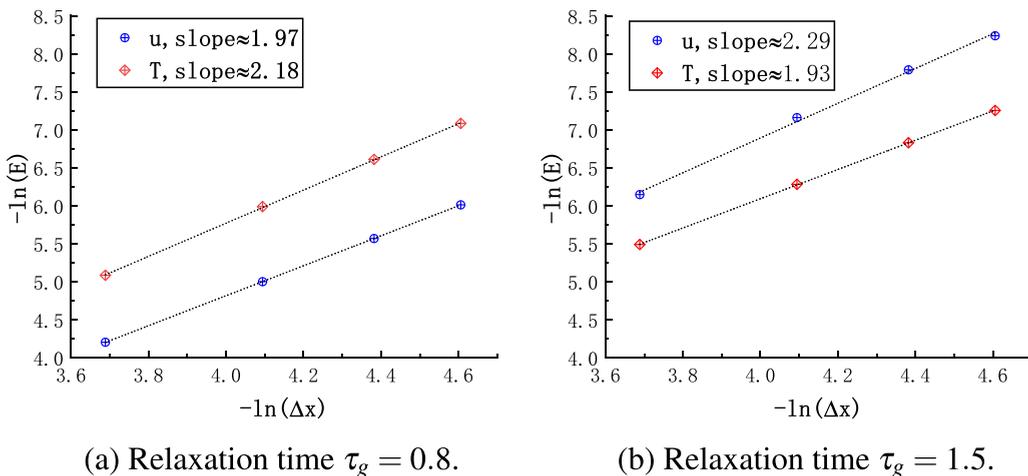


FIG. 8. Relative errors E of velocity and temperature versus grid spacing Δx .

TABLE IV. Comparison of Nusselt numbers from different methods.

Ra		10^3	10^4	10^5	10^6
\bar{Nu}	Present	1.119	2.251	4.535	8.859
	Chen [26]	1.118	2.243	4.498	8.713
	TLBM [43]				
	Davis [41]	1.116	2.234	4.510	8.798
Nu_0	Present	1.118	2.242	4.511	8.812
	Chen [26]	1.118	2.249	4.526	8.732
	TLBM [43]	1.117	2.235	4.504	8.767
	Davis [41]	1.116	2.242	4.532	8.928
Nu_{max}	Present	1.504	3.525	7.676	17.301
	Chen [26]	1.508	3.522	7.804	17.396
	TLBM [43]	1.501	3.507	7.658	17.288
	Davis [41]	1.501	3.545	7.761	18.076
y	Present	0.090	0.147	0.085	0.043
	Chen [26]	0.090	0.140	0.075	0.037
	TLBM [43]	0.086	0.148	0.088	0.044
	Davis [41]	0.087	0.149	0.085	0.046
Nu_{min}	Present	0.696	0.597	0.754	1.027
	Chen [26]	0.691	0.591	0.753	1.050
	TLBM [43]	0.698	0.584	0.728	0.998
	Davis [41]	0.694	0.592	0.736	1.005
y	Present	0.970	0.980	0.990	0.993
	Chen [26]	0.980	1.000	1.000	0.993
	TLBM [43]	0.953	0.984	0.990	0.990
	Davis [41]	1.000	1.000	1.000	1.000

heat conductivities are calculated as

$$\bar{k}_{eq} = \frac{\ln(Ar)}{2\pi} \oint \frac{\partial T}{\partial n} ds. \quad (87)$$

As can be seen, all the numerical results are in good agreement with the reference, implying the present method is reliable on nonuniform meshes and curved boundaries.

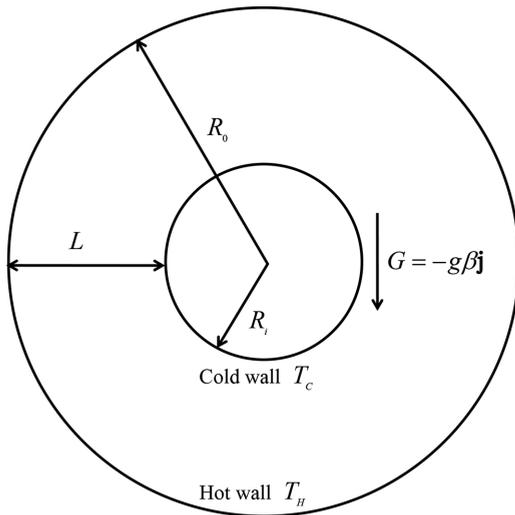


FIG. 9. The illustration of the natural convection in a concentric annulus.

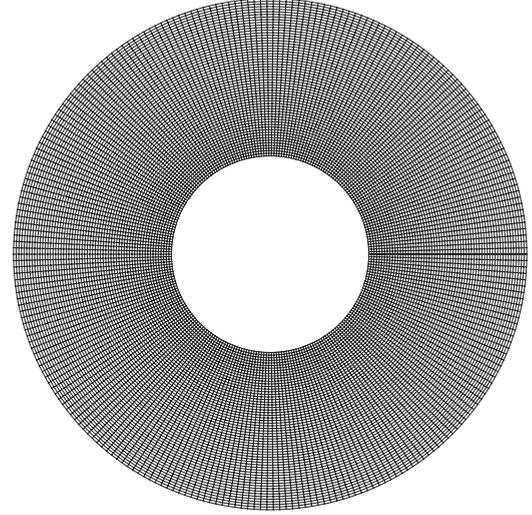


FIG. 10. The body-fitted O-type mesh applied in the computation.

IV. SUMMARY

In this paper, we derive a one-step simplified lattice Boltzmann method of thermal flows under the Boussinesq approximation. The derivation is based on two steps: reconstructing the evolution equation and constructing the nonequilibrium distribution functions. In reconstructing the evolution equation, we start from the general form of the SRT collision model of the LBM and apply the Chapman-Enskog expansion. By using the Taylor expansion to approximate derivatives like D_i , D_i^2 and neglecting high order terms larger than $O(\epsilon^2)$, we finally obtain the reconstructed evolution equation which only has a truncation error of $O(\delta t^2)$. To construct the nonequilibrium distribution functions, the equilibrium distribution functions and their derivatives with respect to space and time are used. We also use some moment conditions and assumptions to simplify the expressions. We obtain two sets of constructed nonequilibrium distribution functions, one for the temperature field and one for the velocity field. Finally, by using these constructed nonequilibrium distribution functions, the moments of the reconstructed evolution equation can be directly taken. Thus, the present OSLBM without tracking and storing distribution functions is derived.

Compared to the LBM, the computational efficiency of the OSLBM still needs to be improved. And considering it is based on the recovering macroscopic equations of the LBM, the present OSLBM is a solver at the macroscopic scale. But in the continuum regime, without tracking the distribution functions, the OSLBM has the advantage over virtual memory cost and boundary treatments. In addition, the OSLBM is more numerically stable with small relaxation time τ . Compared to the ACM, the OSLBM is not as efficient as ACM using large time step. However, due to the poor stability of the ACM, usually a small time step is required to ensure numerical stability which will cut down the computational efficiency of the ACM. This implies that the OSLBM is more efficient than the ACM using small time step.

To test the accuracy of the OSLBM, several numerical examples are conducted. In the test of the natural convection

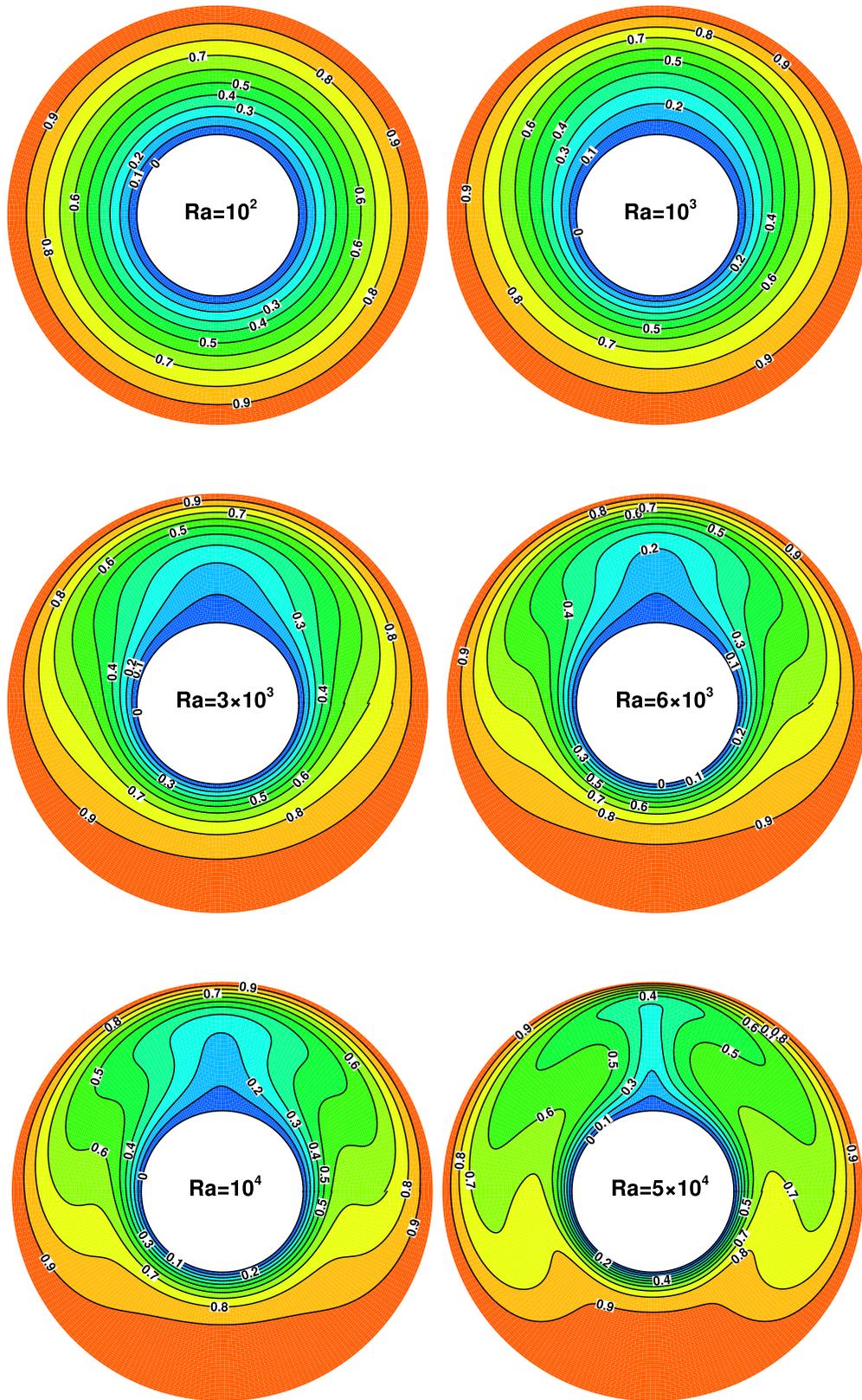


FIG. 11. Isotherms of the natural convection in a concentric annulus at different Ra.

in a square cavity, cases of different Rayleigh numbers Ra are simulated. We compare the Nusselt number Nu , the maximum velocity, and their corresponding positions with the results of other numerical methods. The results show that the method

proposed in this paper has good performance on accuracy. In addition, the test of coarse mesh at high Ra indicates good numerical stability at relaxation time close to 0.5. In the test of the porous plate problem, numerical cases at different Re

TABLE V. Comparison of heat conductivities on the walls.

Ra	Inner cylinder $\overline{k_{\text{eqi}}}$				Outer cylinder $\overline{k_{\text{eqo}}}$			
	Wang <i>et al.</i> [39]	Shu [46]	Chen <i>et al.</i> [26]	Present	Wang <i>et al.</i> [39]	Shu [46]	Chen <i>et al.</i> [26]	Present
10^2	1.002	1.001	0.986	1.004	1.002	1.001	1.007	0.999
10^3	1.076	1.082	1.091	1.085	1.078	1.082	1.082	1.081
3×10^3	1.381	1.397	1.393	1.402	1.384	1.397	1.403	1.395
6×10^3	1.695	1.715	1.698	1.725	1.701	1.715	1.727	1.714
10^4	1.960	1.979	1.947	1.992	1.960	1.979	1.994	1.980
5×10^4	2.941	2.958	2.909	2.998	2.941	2.958	2.975	3.005

and Pr are conducted and the velocity and temperature distributions are compared with the analytical solutions. The results of relative errors versus grid spacing prove that the present method has second order of accuracy in space. The last numerical example of natural convection in a concentric annulus is conducted to study the performance on nonuniform mesh and curved boundary. Different cases of Ra are carried on a 250×60 body-fitted O-type mesh, in which good agreement with other methods shows that the present method is reliable to apply the nonuniform mesh and curved boundaries.

ACKNOWLEDGMENTS

We acknowledge the financial support from the National Natural Science Foundation of China (Grant No. 51979115), and the special project in Ship Of Green Ecology and Environmental Protection (Grant No. 2020307). L. Yang is supported by National Natural Science Foundation of China (Grant No. 52201374), Hainan Provincial Natural Science Foundation of China (Grant No. 522QN343), and the Fundamental Research Funds for the Central Universities (Grant No. WUT:223173002).

- [1] A. Bairi, E. Zarco-Pernia, and J.-M. García de María, A review on natural convection in enclosures for engineering applications. The particular case of the parallelogrammic diode cavity, *Appl. Therm. Eng.* **63**, 304 (2014).
- [2] P. Mayeli and G. J. Sheard, Buoyancy-driven flows beyond the Boussinesq approximation: A brief review, *Int. Commun. Heat Mass Transfer* **125**, 105316 (2021).
- [3] M. König and M. Weber Sutter, Applying common equations of state to three reference fluids: Water, carbon dioxide, and helium, *Chem. Ing. Tech.* **94**, 493 (2022).
- [4] X. Shan, Simulation of Rayleigh-Bénard convection using a lattice Boltzmann method, *Phys. Rev. E* **55**, 2780 (1997).
- [5] S. Chen and G. D. Doolen, Lattice Boltzmann method for fluid flows, *Annu. Rev. Fluid Mech.* **30**, 329 (1998).
- [6] S. Chen, H. Chen, D. Martínez, and W. Matthaeus, Lattice boltzmann model for simulation of magnetohydrodynamics, *Phys. Rev. Lett.* **67**, 3776 (1991).
- [7] Y. H. Qian, D. D'Humières, and P. Lallemand, Lattice BGK models for Navier-Stokes equation, *Europhys. Lett.* **17**, 479 (1992).
- [8] F. J. Alexander, S. Chen, and J. D. Sterling, Lattice Boltzmann thermohydrodynamics, *Phys. Rev. E* **47**, R2249 (1993).
- [9] N. Cao, S. Chen, S. Jin, and D. Martínez, Physical symmetry and lattice symmetry in the lattice Boltzmann method, *Phys. Rev. E* **55**, R21 (1997).
- [10] S. Chakraborty and D. Chatterjee, An enthalpy-based hybrid lattice-boltzmann method for modelling solid-liquid phase transition in the presence of convective transport, *J. Fluid Mech.* **592**, 155 (2007).
- [11] O. Filippova and D. Hänel, A novel lattice BGK approach for low Mach number combustion, *J. Comput. Phys.* **158**, 139 (2000).
- [12] A. Bartoloni, LBE simulations of Rayleigh-Benard convection on the APE100 parallel processor, *Int. J. Mod. Phys. C* **04**, 993 (1993).
- [13] J. Eggels and J. Somers, Numerical simulation of free convective flow using the lattice-Boltzmann scheme, *Int. J. Heat Fluid Flow* **16**, 357 (1995).
- [14] X. He, S. Chen, and G. D. Doolen, A novel thermal model for the lattice Boltzmann method in incompressible limit, *J. Comput. Phys.* **146**, 282 (1998).
- [15] R. G. M. van der Sman, Galilean invariant lattice Boltzmann scheme for natural convection on square and rectangular lattices, *Phys. Rev. E* **74**, 026705 (2006).
- [16] Q. Li, Z. Chai, and B. Shi, A novel lattice Boltzmann model for the coupled viscous Burgers' equations, *Appl. Math. Comput.* **250**, 948 (2015).
- [17] Z. Chai and B. Shi, A novel lattice Boltzmann model for the Poisson equation, *Appl. Math. Model.* **32**, 2050 (2008).
- [18] Z. Chen, C. Shu, Y. Wang, L. M. Yang, and D. Tan, A simplified lattice Boltzmann method without evolution of distribution function, *Adv. Appl. Math. Mech.* **9**, 1 (2017).
- [19] Z. Chen, C. Shu, D. Tan, and C. Wu, On improvements of simplified and highly stable lattice Boltzmann method: Formulations, boundary treatment, and stability analysis, *Int. J. Numer. Methods Fluids* **87**, 161 (2007).
- [20] Z. Chen, C. Shu, D. Tan, X. D. Niu, and Q. Z. Li, Simplified multiphase lattice Boltzmann method for simulating multiphase flows with large density ratios and complex interfaces, *Phys. Rev. E* **98**, 063314 (2018).
- [21] Z. Chen, C. Shu, L. M. Yang, X. Zhao, and N. Y. Liu, Phase-field-simplified lattice Boltzmann method for modeling solid-liquid phase change, *Phys. Rev. E* **103**, 023308 (2021).
- [22] Z. Chen, C. Shu, and D. Tan, Immersed boundary-simplified lattice Boltzmann method for incompressible viscous flows, *Phys. Fluids* **30**, 053601 (2018).
- [23] Z. Chen, C. Shu, L. M. Yang, X. Zhao, and N. Y. Liu, Immersed boundary-simplified thermal lattice Boltzmann method for incompressible thermal flows, *Phys. Fluids* **32**, 013605 (2020).

- [24] Z. Chen and C. Shu, Simplified lattice Boltzmann method for non-Newtonian power-law fluid flows, *Int. J. Numer. Methods Fluids* **92**, 38 (2020).
- [25] Y. Gao, Y. Yu, L. Yang, S. Qin, and G. Hou, Development of a coupled simplified lattice Boltzmann method for thermal flows, *Comput. Fluids* **229**, 105042 (2021).
- [26] Z. Chen, C. Shu, and D. Tan, A simplified thermal lattice Boltzmann method without evolution of distribution functions, *Int. J. Heat Mass Transfer* **105**, 741 (2017).
- [27] S. Qin, L. Yang, G. Hou, Y. Gao, and W. Guo, A one-step simplified lattice Boltzmann method without evolution of distribution functions, *Int. J. Numer. Methods Fluids* **94**, 1001 (2022).
- [28] J. Lu, H. Lei, and C. Dai, A unified thermal lattice Boltzmann equation for conjugate heat transfer problem, *Int. J. Heat Mass Transfer* **126**, 1275 (2018).
- [29] H. Liang, J. Xu, J. Chen, H. Wang, Z. Chai, and B. Shi, Phase-field-based lattice Boltzmann modeling of large-density-ratio two-phase flows, *Phys. Rev. E* **97**, 033309 (2018).
- [30] H. Liang, B. C. Shi, Z. L. Guo, and Z. H. Chai, Phase-field-based multiple-relaxation-time lattice Boltzmann model for incompressible multiphase flows, *Phys. Rev. E* **89**, 053320 (2014).
- [31] D. Kwak, C. Kiris, and C. S. Kim, Computational challenges of viscous incompressible flows, *Comput. Fluids* **34**, 283 (2005).
- [32] S. Patankar and D. Spalding, A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows, *Int. J. Heat Mass Transfer* **15**, 1787 (1972).
- [33] R. Issa, Solution of the implicitly discretised fluid flow equations by operator-splitting, *J. Comput. Phys.* **62**, 40 (1986).
- [34] A. J. Chorin, A numerical method for solving incompressible viscous flow problems, *J. Comput. Phys.* **2**, 12 (1967).
- [35] J. Lu, H. Lei, C. Shu, and C. Dai, The more actual macroscopic equations recovered from lattice Boltzmann equation and their applications, *J. Comput. Phys.* **415**, 109546 (2020).
- [36] U. Ghia, K. Ghia, and C. Shin, High-Re solutions for incompressible flow using the Navier-Stokes equations and a multigrid method, *J. Comput. Phys.* **48**, 387 (1982).
- [37] Z.-L. Guo, C.-G. Zheng, and B.-C. Shi, Non-equilibrium extrapolation method for velocity and pressure boundary conditions in the lattice Boltzmann method, *Chin. Phys. (Overseas Edition)* **11**, 366 (2002).
- [38] S. Qin, G. Hou, Y. Yang, Liumingand Gao, W. Guo, and J. Shi, A novel one-step simplified lattice Boltzmann method and its application to multiphase flows with large density ratio, *Phys. Fluids* **35**, 053318 (2023).
- [39] Y. Wang, C. Shu, and C. Teo, Thermal lattice Boltzmann flux solver and its application for simulation of incompressible thermal flows, *Comput. Fluids* **94**, 98 (2014).
- [40] C. Shu and H. Xue, Comparison of two approaches for implementing stream function boundary conditions in DQ simulation of natural convection in a square cavity, *Int. J. Heat Fluid Flow* **19**, 59 (1998).
- [41] G. De Vahl Davis, Natural convection of air in a square cavity: A bench mark numerical solution, *Int. J. Numer. Methods Fluids* **3**, 249 (1983).
- [42] Y. Peng, C. Shu, and Y. T. Chew, Simplified thermal lattice Boltzmann model for incompressible thermal flows, *Phys. Rev. E* **68**, 026701 (2003).
- [43] A. D’Orazio, M. Corcione, and G. P. Celata, Application to natural convection enclosed flows of a lattice Boltzmann BGK model coupled with a general purpose thermal boundary condition, *Int. J. Therm. Sci.* **43**, 575 (2004).
- [44] X. He, L. Luo, M. Dembo, and Q. Zou, Analytic solutions of simple flows and analysis of nonslip boundary conditions for the lattice Boltzmann BGK model, *J. Stat. Phys.* **87**, 115 (1997).
- [45] C. S. Zhen Chen and D. S. Tan, The simplified lattice Boltzmann method on non-uniform meshes, *Commun. Comput. Phys.* **23**, 1131 (2018).
- [46] C. Shu, Application of differential quadrature method to simulate natural convection in a concentric annulus, *Int. J. Numer. Methods Fluids* **30**, 977 (1999).