

Development of an efficient gas kinetic scheme for simulation of two-dimensional incompressible thermal flows

L. M. Yang,^{1,2,3} C. Shu,^{3,*} W. M. Yang,^{2,3} and J. Wu¹

¹*Department of Aerodynamics, College of Aerospace Engineering, Nanjing University of Aeronautics and Astronautics, Yuda Street, Nanjing 210016, Jiangsu, China*

²*Sembcorp-NUS Corporate Laboratory, 1 Engineering Drive 2, Singapore 117576*

³*Department of Mechanical Engineering, National University of Singapore, 10 Kent Ridge Crescent, Singapore 119260*



(Received 11 September 2017; revised manuscript received 17 November 2017; published 11 January 2018)

In this work, an efficient gas kinetic scheme is presented for simulation of two-dimensional incompressible thermal flows. In the scheme, the macroscopic governing equations for mass, momentum, and energy conservation are discretized by the finite volume method and the numerical fluxes at the cell interface are reconstructed by the local solution of the Boltzmann equation. To compute these fluxes, two distribution functions are involved. One is the circular function, which is used to calculate the numerical fluxes of mass and momentum equations. Due to the incompressible limit, the circle at the cell interface can be approximately considered to be symmetric so that the expressions for the conservative variables and numerical fluxes at the cell interface can be given explicitly and concisely. Another one is the D2Q4 model, which is utilized to compute the numerical flux of the energy equation. By following the process for derivation of numerical fluxes of mass and momentum equations, the numerical flux of the energy equation can also be given explicitly. The accuracy, efficiency, and stability of the present scheme are validated by simulating several thermal flow problems. Numerical results showed that the present scheme can provide accurate numerical results for incompressible thermal flows at a wide range of Rayleigh numbers with less computational cost than that needed by the thermal lattice Boltzmann flux solver (TLBFS), which has been proven to be more efficient than the thermal lattice Boltzmann method (TLBM).

DOI: [10.1103/PhysRevE.97.013305](https://doi.org/10.1103/PhysRevE.97.013305)

I. INTRODUCTION

The Boltzmann equation is obtained by kinetic theory. It provides the theoretical connection between hydrodynamics and the underlying microscopic physics. Within the limitation of local equilibrium for the distribution function, the Euler equations, Navier-Stokes (NS) equations, and Burnett and super-Burnett equations can be derived from the Boltzmann equation by using the truncated expansion to different orders. This means that we can solve fluid flow problems directly by the solution of Boltzmann equation. These kinds of methods are generally called kinetic methods. Two typical kinetic methods are the lattice Boltzmann method (LBM) [1–4] and gas kinetic scheme (GKS) [5–8]. Due to their kinetic nature and strong foundation in physics, these two methods have received much attention and have been widely used for modeling and simulation of complex fluids [9–10].

In LBM, the discrete particle distribution functions are updated by solving the algebraic lattice Boltzmann equation (LBE) and the macroscopic flow properties are then computed by the moments of particle distribution functions. The solution of the standard LBM is very simple and efficient, and only consists of a linear streaming process and a collision process [11]. Due to its distinct features, the application of LBM has been extended to solving incompressible thermal flow problems [12–19]. So far, a number of promising thermal LBE

models, such as the two-component distribution model [12], the thermal energy distribution model [13–14], the temperature distribution model [15], and the total energy distribution model [16], have been proposed to describe the evolution of the temperature/energy. The thermal lattice Boltzmann method (TLBM) inherits the merits of the standard isothermal LBM but, at the same time, also suffers from its drawbacks. First, due to the lattice uniformity, the applications of TLBM are usually restricted to uniform meshes. To apply it on nonuniform meshes, some extra computational efforts are required [20–22]. In addition, the TLBM requires large amounts of virtual memory, because both the density and the temperature/energy distribution functions need to be stored during the computation [23]. Moreover, in the implementation of TLBM, the physical boundary condition must be converted into the boundary condition for distribution functions. This process is quite challenging for cases with curved boundaries [24]. To overcome these drawbacks, Wang *et al.* [25–26] combined the LBM with the conventional Navier-Stokes solver and proposed the thermal lattice Boltzmann flux solver (TLBFS) for simulation of incompressible thermal flows. In their method, the incompressible Navier-Stokes equations are discretized by the finite volume method (FVM) and the numerical fluxes at the cell interface are reconstructed by the local solution of the LBE. The conservative variables at cell centers are then obtained by marching the macroscopic governing equations in time. This solver has been proven to be more efficient than the TLBM of Peng *et al.* [14].

*Corresponding author: mpeshuc@nus.edu.sg

Different from the LBM, the GKS solves the macroscopic governing equations directly so that the shortcomings of standard LBM, which are mentioned above, can be effectively overcome. In GKS, the finite volume method (FVM) or the finite difference method (FDM) is commonly adopted to discretize the macroscopic governing equations, and the local solution of the continuous Boltzmann equation is utilized to compute the numerical fluxes of conservative variables at the cell interface. The conventional GKS usually applies the Maxwellian function as the equilibrium distribution [27–31]. For simulation of compressible viscous flows, the formulations of conventional GKS are quite complicated due to the discontinuity of conservative variables and their derivatives at the cell interface [31]. By assuming that the flow variables and their derivatives at the cell interface change smoothly, several simple GKSs have been developed for simulation of both incompressible isothermal flows [32–34] and thermal flows [35]. Although the incompressible GKS is simpler than the compressible one, it is very inefficient compared to LBM. As indicated by Guo *et al.* [34], for two-dimensional (2D) problems, the LBM is about 10 and 3 times faster than the GKS for steady and unsteady flow calculations, respectively. In addition, Chen *et al.* [36] recently found that the above incompressible GKSs may encounter a stability problem when simulating incompressible flows at high Reynolds numbers. Hence, they commented that the discontinuous derivatives of flow variables at the cell interface should be retained in order to improve the stability of incompressible GKS. Nevertheless, this treatment leads the implementation of incompressible GKS [36–37] to be as complicated as the compressible one. Moreover, there are very few applications of GKS for simulation of incompressible thermal flows in the literature. Among the limited works, Xu and Lui [35] proposed an incompressible GKS to solve Rayleigh-Bénard flows by introducing an additional distribution function for the evolution of thermal energy. Given the above, it is imperative to develop an efficient and robust GKS for simulation of incompressible flows, especially for thermal flows.

To simplify the Maxwellian function-based GKS [5–7,27–31], Shu and his coworkers proposed the circular function-based GKS (CGKS) [38–40] for 2D cases. In CGKS, the Maxwellian distribution function is first simplified to a circular function. Then the integrals for conservation forms of moments in the infinite domain for the Maxwellian function-based GKS, which are needed to recover macroscopic governing equations, are reduced to those in the finite domain (integrals along the circle) for the CGKS. As a consequence, the expressions for numerical fluxes at the cell interface can be simplified correspondingly. In order to improve the computational efficiency of incompressible isothermal flow problems, a simplified CGKS is further developed based on the incompressible limit [41]. In the simplified CGKS, the integral domain along the circle is approximately considered to be symmetric at the cell interface due to the incompressible limit. Besides that, the energy equation is neglected since the incompressible isothermal flows are simulated. With all these simplifications, the formulations for the conservative variables and numerical fluxes at the cell interface can be given explicitly and concisely. At the same time, the discontinuity of conservative variables and their derivatives at the cell interface can still be retained for

the simplified CGKS in order to keep good numerical stability at high Reynolds numbers. This scheme has been proven to be as accurate and robust as the original CGKS [39] and the lattice Boltzmann flux solver (LBFS) [42] for simulation of incompressible isothermal flows. In terms of computational cost, the simplified CGKS is more efficient than the other two. Hence, it is worthwhile to extend the application of simplified CGKS to incompressible thermal flows.

It can be found from the above discussion that the distribution function of temperature/energy is usually introduced in TLBM [12–19], TLBFS [25–26], and GKS of Xu and Lui [35] in order to solve incompressible thermal flow problems. In TLBM and TLBFS, the discrete thermal LBE model is used, while in GKS of Xu and Lui [35] the continuous distribution function is utilized. Due to its simplicity, the discrete thermal LBE model is introduced into the simplified CGKS for simulation of incompressible thermal flows in this work. To date, a lot of thermal LBE models have been proposed for incompressible thermal flows [12–16]. Among these models, the D2Q4 model proposed by Guo *et al.* [15] is relatively simple, containing only four discrete velocity points for the 2D case. As commented by Huang *et al.* [43], the D2Q4 model is capable of obtaining results with accuracy equal to the D2Q5 or D2Q9 models, while the time of simulation with the D2Q4 model is less than that of the latter. In addition, the D2Q4 model has been validated to have excellent numerical stability and accuracy at high Rayleigh numbers up to 10^8 for simulation of natural convection in a square cavity [44]. Therefore, it was chosen as the equilibrium distribution function for the evolution of temperature in the present work. From Chapman-Enskog expansion analysis, the governing equation of temperature can be recovered from the LBE with the D2Q4 model, and the flux of governing equation at the cell interface can be computed by the moment of particle distribution functions. Like the derivation of the simplified CGKS [41], we can discretize the governing equation of temperature by FVM and evaluate the numerical flux by the local solution of LBE with the D2Q4 model. In addition, the effect of temperature field on the flow field can be taken into account by the buoyancy force represented in the momentum equation. Although the particle velocity space is continuous for the circular function and discrete for the D2Q4 model, it has no direct effect on the Galilean invariance of the present scheme. The reason is that the Navier-Stokes equations are directly solved in the present method and the local solution of the Boltzmann equation is only used to reconstruct the numerical fluxes at the cell interface. It is known that the Navier-Stokes equations are Galilean invariant. To validate the present method, several test examples of incompressible thermal flows with various Rayleigh numbers are simulated. Since it has been proven to be accurate, robust and efficient for simulation of incompressible thermal flows, the TLBFS of Wang *et al.* [25–26] will be chosen as the basis to assess the capability of the present scheme.

II. MACROSCOPIC GOVERNING EQUATIONS, BOLTZMANN EQUATION, AND DISTRIBUTION FUNCTIONS

A. Macroscopic governing equations and FVM discretization

In this work, we confine the study to incompressible thermal flows. The macroscopic governing equations usually consist

of incompressible Navier-Stokes equations and the energy equation, which can be written as [35]

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0, \quad (1)$$

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u} \mathbf{u} + p \mathbf{I}) = \nabla \cdot \{\mu [\nabla \mathbf{u} + (\nabla \mathbf{u})^T]\} + \mathbf{f}_E, \quad (2)$$

$$\frac{\partial T}{\partial t} + \nabla \cdot (T \mathbf{u}) = \kappa \nabla^2 T, \quad (3)$$

where ρ , \mathbf{u} , p , and T are respectively the density, velocity, pressure, and temperature of fluid flow. μ and κ represent the dynamic viscosity and the thermal diffusivity. \mathbf{I} is the unit tensor. \mathbf{f}_E is the buoyancy force resulting from the uneven distribution of temperature field. According to the Boussinesq approximation, \mathbf{f}_E can be computed by

$$\mathbf{f}_E = -\rho\beta(T - T_m)g\mathbf{j}, \quad (4)$$

where g represents the acceleration due to gravity, \mathbf{j} is the unit vector in the y direction, β is the thermal expansion coefficient, and T_m is the average temperature of the flow field. As can be seen from Eq. (3), the temperature field is affected by the flow field. In contrast, the buoyancy force resulting from temperature field also plays an essential role as an external force for the momentum equation, as shown in Eq. (2).

To solve thermal flow problems with curved boundaries, the governing equations (1)–(3) are discretized by the finite volume method (FVM) in this work, where the conservative variables are defined at cell centers. For the 2D case, Eqs. (1)–(3) given by FVM can be written as

$$\frac{d\mathbf{W}_I}{dt} = -\frac{1}{\Omega_I} \sum_{i=1}^{N_f} \mathbf{F}_{ni} S_i + Q_I, \quad (5)$$

where I is the index of a control volume, and Ω_I and N_f represent the volume and the number of the faces of the control volume I , respectively. S_i denotes the area of the i th interface of the control volume. The conservative variable vector \mathbf{W} , flux \mathbf{F}_n , and source term Q are given by

$$\mathbf{W} = (\rho, \rho u, \rho v, T)^T, \quad (6)$$

$$\mathbf{F}_n = (F_\rho, F_{\rho u}, F_{\rho v}, F_T)^T, \quad (7)$$

$$Q = (0, 0, -\rho\beta(T - T_m)g, 0)^T. \quad (8)$$

$\mathbf{u} = (u, v)$ is the velocity vector expressed in the global Cartesian coordinate system. The calculation of flux \mathbf{F}_n and source term Q are two keys for solving Eq. (5). The source term Q can be computed straightforwardly from the conservative variables \mathbf{W} at cell centers. The calculation of \mathbf{F}_n will be discussed in the following sections.

For the convenience of derivation, a local-coordinate system is introduced with the x_1 axis pointing in the normal direction and the x_2 axis pointing in the tangential direction with respect to the cell interface. In the local-coordinate system, the conservative variables and fluxes can be expressed as

$$\bar{\mathbf{W}} = (\rho, \rho u_1, \rho u_2, T)^T, \quad (9)$$

$$\bar{\mathbf{F}}_n = (F_\rho, F_{\rho u_1}, F_{\rho u_2}, F_T)^T, \quad (10)$$

where $\mathbf{u} = (u_1, u_2)$ is the velocity vector expressed in the local-coordinate system. By applying the coordinate transformation from the local-coordinate system to the global Cartesian coordinate system, we can get the relationship between Eqs. (7) and (10) as follows:

$$\mathbf{F}_n = (F_\rho, n_x F_{\rho u_1} - n_y F_{\rho u_2}, n_x F_{\rho u_2} + n_y F_{\rho u_1}, F_T)^T. \quad (11)$$

$\mathbf{n} = (n_x, n_y)$ denotes the unit normal vector of the cell interface in the global Cartesian coordinate system. It can be seen from Eq. (11) that the calculation of \mathbf{F}_n is equivalent to evaluating $\bar{\mathbf{F}}_n$ in the local-coordinate system.

B. Boltzmann equation, circular function, and D2Q4 model

In order to recover the macroscopic governing equations (1)–(3) without the external forcing term \mathbf{f}_E , two Boltzmann equations with different distribution functions can be constructed in the following forms [15,44]:

$$\frac{\partial f}{\partial t} + \boldsymbol{\xi} \cdot \nabla f = \frac{f^{\text{eq}} - f}{\tau_v}, \quad (12)$$

$$\frac{\partial h}{\partial t} + \boldsymbol{\xi} \cdot \nabla h = \frac{h^{\text{eq}} - h}{\tau_\kappa}, \quad (13)$$

where f and h are respectively the density distribution function and the temperature distribution function. f^{eq} and h^{eq} are the equilibrium states approached by f and h through particle collisions within the collision time scales τ_v and τ_κ , respectively. $\boldsymbol{\xi} = (\xi_1, \xi_2)$ is the particle velocity in the particle velocity space. Equation (12) is used to recover the mass and momentum equations without the external forcing term \mathbf{f}_E ; Eq. (13) is utilized to recover the energy equation.

The equilibrium state f^{eq} in Eq. (12) is originally the Maxwellian distribution function. As reported in [38–39], for the 2D case, the Maxwellian distribution function can be simplified to the circular function g_C given by,

$$f^{\text{eq}} = g_C = \begin{cases} \frac{\rho}{2\pi} & \text{if } (\xi_1 - u_1)^2 + (\xi_2 - u_2)^2 = c^2, \\ 0 & \text{otherwise,} \end{cases} \quad (14)$$

where the square of radius c actually represents the mean kinetic energy of the particles. For incompressible flows, c^2 can be linked to the reference velocity u_0 and Mach number Ma by the following form [41]:

$$c^2 = D c_s^2 = \frac{D u_0^2}{\text{Ma}^2} \quad (15)$$

Here, D is the abbreviation of the dimension ($D = 2$ for two dimensions) and $c_s = u_0/\text{Ma}$ is the sound speed. For simulation of incompressible flows, u_0 and c_s can be taken as 0.1 and $1/\sqrt{3}$ respectively to satisfy the requirement of the incompressible limit (i.e., $\text{Ma} < 0.3$). This setup is also widely used in the lattice Boltzmann method (LBM) [1–4]. In addition, from Eq. (14), the particle velocity components in the local coordinate system can be expressed as

$$\xi_1 = u_1 + c \cos(\theta), \quad (16a)$$

$$\xi_2 = u_2 + c \sin(\theta), \quad (16b)$$

The definition of the angle θ is shown in Fig. 1.

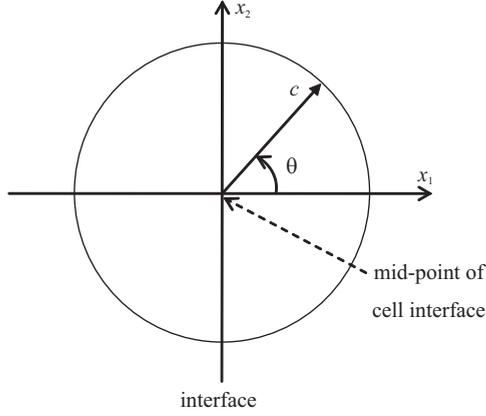


FIG. 1. Distribution of the circular function in the particle velocity space at a cell interface for incompressible flows.

With notations of $c_1 = c \cos(\theta)$, $c_2 = c \sin(\theta)$, and $\xi_\alpha = u_\alpha + c_\alpha$, the conservation forms of moments for the circular function, which are used to recover incompressible Navier-Stokes equations, can be expressed as [41]

$$\int_0^{2\pi} g_C d\theta = \rho, \quad (17a)$$

$$\int_0^{2\pi} g_C \xi_\alpha d\theta = \rho u_\alpha, \quad (17b)$$

$$\int_0^{2\pi} g_C \xi_\alpha \xi_\beta d\theta = \rho u_\alpha u_\beta + p \delta_{\alpha\beta}, \quad (17c)$$

$$\int_0^{2\pi} g_C \xi_\alpha \xi_\beta \xi_\chi d\theta = p(u_\alpha \delta_{\beta\chi} + u_\beta \delta_{\chi\alpha} + u_\chi \delta_{\alpha\beta}) + \rho u_\alpha u_\beta u_\chi, \quad (17d)$$

where ξ_α , ξ_β , ξ_χ and u_α , u_β , u_χ are the particle velocities and macroscopic flow velocities in the α , β , and χ directions, respectively. As shown in Eq. (17), the connection between the distribution function and the conservative variables and fluxes can be written as

$$\overline{\mathbf{W}}(1:3) = \int_0^{2\pi} \boldsymbol{\varphi}_\alpha f d\theta, \quad (18)$$

$$\overline{\mathbf{F}}_n(1:3) = \int_0^{2\pi} \xi_1 \boldsymbol{\varphi}_\alpha f d\theta. \quad (19)$$

Here, the notation (1 : 3) represents the first three components of the vector. $\boldsymbol{\varphi}_\alpha$ is the moment vector given by

$$\boldsymbol{\varphi}_\alpha = (1, \xi_1, \xi_2)^T \quad (20)$$

In addition, from the Chapman-Enskog expansion analysis, the relationship between the kinematic viscosity μ and collision time scale τ_v can be expressed as [33]

$$\tau_v = \frac{\mu}{\rho c^2 / D} = \frac{\mu}{\rho c_s^2}. \quad (21)$$

Equation (19) shows that the key issue to calculate the fluxes $\overline{\mathbf{F}}_n(1:3)$ is to obtain the distribution function f and the moments $\boldsymbol{\varphi}_\alpha$ at the cell interface.

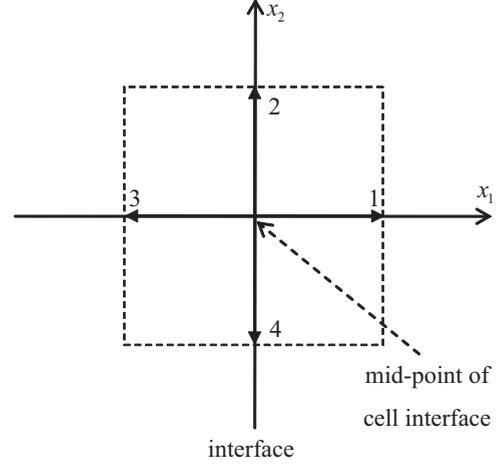


FIG. 2. Distribution of D2Q4 model in the particle velocity space at a cell interface.

To solve the energy equation, since the discrete LBE model is used, we should reformulate Eq. (13) into the discrete form

$$\frac{\partial h_\alpha}{\partial t} + \mathbf{e}_\alpha \cdot \nabla h_\alpha = \frac{h_\alpha^{\text{eq}} - h_\alpha}{\tau_\kappa}. \quad (22)$$

Note that in Eq. (22), the particle velocity $\boldsymbol{\xi}$ has been written as \mathbf{e}_α . In this work, the D2Q4 model is applied in the discrete lattice velocity space. The equilibrium state h_α^{eq} and the discrete particle velocity \mathbf{e}_α of the D2Q4 model are given by [15]

$$h_\alpha^{\text{eq}} = \frac{T}{4} [1 + 2\mathbf{e}_\alpha \cdot \mathbf{u}], \quad \alpha = 1, 2, 3, 4, \quad (23)$$

$$\mathbf{e}_\alpha = (\cos[\pi(\alpha - 1)/2], \sin[\pi(\alpha - 1)/2]), \quad \alpha = 1, 2, 3, 4. \quad (24)$$

The distribution of the D2Q4 model in the particle velocity space at a cell interface in the local coordinate system is shown in Fig. 2.

From the Chapman-Enskog expansion analysis, we can get the connection between the distribution function and temperature and flux of the energy equation as follows:

$$\overline{\mathbf{W}}(4) = \sum_\alpha h_\alpha, \quad (25)$$

$$\overline{\mathbf{F}}_n(4) = \sum_\alpha e_{\alpha,1} h_\alpha, \quad (26)$$

where the notation (4) denotes that the fourth component of the vector is computed by Eqs. (25) and (26). $e_{\alpha,1}$ is the first component of particle velocity in the local-coordinate system. In addition, the relationship between the thermal diffusivity κ and collision time scale τ_κ can be written as

$$\tau_\kappa = 2\kappa. \quad (27)$$

The detailed Chapman-Enskog expansion analysis for recovering the energy equation from Eq. (22) with the D2Q4 model can be found in Appendix A. Equation (26) shows that the key issue to calculate the flux of the energy equation, $\overline{\mathbf{F}}_n(4)$, is to compute the distribution function h_α at the cell interface.

III. AN EFFICIENT GAS KINETIC SCHEME FOR INCOMPRESSIBLE THERMAL FLOWS

A. Evolution for fluxes of mass and momentum equations

In our early work [41], a simplified CGKS was proposed for simulation of incompressible isothermal flows, and the expressions of numerical fluxes for mass and momentum equations were derived in detail. These formulations can be directly used to calculate the fluxes $\bar{\mathbf{F}}_n(1:3)$ in this work. For convenient application, a brief review and the final expressions of the simplified CGKS are given in this subsection. Suppose that the cell interface is located at $\mathbf{r} = 0$; the distribution function at the cell interface can be written as

$$f(0,t) = f^{\text{eq}}(0,t) + f^{\text{neq}}(0,t) \approx g_C(0,t) + \tau_v^*[g_C(-\xi\delta t, t - \delta t) - g_C(0,t)], \quad (28)$$

where $g_C(0,t) = f^{\text{eq}}(0,t)$ is the equilibrium distribution function at the cell interface, and $g_C(-\xi\delta t, t - \delta t)$ is the equilibrium distribution function on the circle. $\tau_v^* = \tau_v/\delta t$ is the dimensionless collision time and δt is the streaming time step in the solution reconstruction at the cell interface. The determination of δt can be found in Appendix B. To simplify the notation, we denote $(0,t)$ by superscript *face* and $(-\xi\delta t, t - \delta t)$ by superscript *cir* in the following text. Substituting Eq. (28) into Eq. (19), we can get

$$\begin{aligned} \bar{\mathbf{F}}_n(1:3) &= \int \xi_1^{\text{face}} \boldsymbol{\varphi}_\alpha^{\text{face}} g_C^{\text{face}} d\theta + \tau_v^* \left[\int \xi_1^{\text{cir}} \boldsymbol{\varphi}_\alpha^{\text{cir}} g_C^{\text{cir}} d\theta \right. \\ &\quad \left. - \int \xi_1^{\text{face}} \boldsymbol{\varphi}_\alpha^{\text{face}} g_C^{\text{face}} d\theta \right] \\ &= \mathbf{F}^I(1:3) + \tau_v^*[\mathbf{F}^{II}(1:3) - \mathbf{F}^I(1:3)]. \end{aligned} \quad (29)$$

In Eq. (29), $\mathbf{F}^I(1:3)$ denotes the flux attributed to the equilibrium distribution function and moments at the cell interface, and $\mathbf{F}^{II}(1:3)$ represents the flux attributed to the equilibrium distribution function and moments on the circle.

In the simplified CGKS, the equilibrium distribution function and moments are functions of the conservative variables. Furthermore, the conservative variables on the circle can be obtained by interpolation from those at cell centers, and the conservative variables at the cell interface can be computed by Eq. (18). According to the compatibility condition and considering the incompressible limit, the conservative variables at the cell interface can be expressed as [41]

$$\bar{\mathbf{W}}^{\text{face}}(1:3) = \int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} \boldsymbol{\varphi}_\alpha^{\text{cir},L} g_C^{\text{cir},L} d\theta + \int_{\frac{\pi}{2}}^{\frac{3\pi}{2}} \boldsymbol{\varphi}_\alpha^{\text{cir},R} g_C^{\text{cir},R} d\theta. \quad (30)$$

The superscripts *L* and *R* represent the variables defined at the left and right cells, respectively. Once the conservative variables at the cell interface are obtained, the flux $\mathbf{F}^I(1:3)$ can be calculated by directly substituting $\bar{\mathbf{W}}^{\text{face}}(1:3)$ into the expression of inviscid flux, i.e.,

$$\mathbf{F}^I(1:3) = \left[\begin{array}{c} \rho u_1 \\ \rho u_1 u_1 + \rho c^+ c^+ / D \\ \rho u_1 u_2 \end{array} \right]^{\text{face}}, \quad (31)$$

where $c^+ = c$ is a constant determined from Eq. (15). Similarly to the derivation of Eq. (30), the flux $\mathbf{F}^{II}(1:3)$ can be computed by

$$\begin{aligned} \mathbf{F}^{II}(1:3) &= \int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} \xi_1^{\text{cir},L} \boldsymbol{\varphi}_\alpha^{\text{cir},L} g_C^{\text{cir},L} d\theta \\ &\quad + \int_{\frac{\pi}{2}}^{\frac{3\pi}{2}} \xi_1^{\text{cir},R} \boldsymbol{\varphi}_\alpha^{\text{cir},R} g_C^{\text{cir},R} d\theta. \end{aligned} \quad (32)$$

The expressions of $\bar{\mathbf{W}}^{\text{face}}(1:3)$ and $\mathbf{F}^{II}(1:3)$ can be found in Ref. [41] or Appendix B. Note that both $\bar{\mathbf{W}}^{\text{face}}(1:3)$ and $\mathbf{F}^{II}(1:3)$ can be expressed concisely and explicitly for the simplified CGKS, which are the functions of the conservative variables and their derivatives at the left and right cells. In the meantime, the discontinuities of conservative variables and their derivatives at the cell interface are still kept in the present scheme for enhancing the numerical stability, as shown in Eqs. (30) and (32).

B. Evolution for flux of energy equation

Like the derivation of the numerical fluxes for mass and momentum equations, the temperature distribution function at the cell interface can also be written as

$$\begin{aligned} h_\alpha(0,t) &= h_\alpha^{\text{eq}}(0,t) + h_\alpha^{\text{neq}}(0,t) \\ &\approx h_\alpha^{\text{eq}}(0,t) + \tau_\kappa^*[h_\alpha^{\text{eq}}(-\mathbf{e}_\alpha \delta t, t - \delta t) - h_\alpha^{\text{eq}}(0,t)]. \end{aligned} \quad (33)$$

where $\tau_\kappa^* = \tau_\kappa/\delta t$ is the dimensionless collision time. By substituting Eq. (33) into Eq. (26), we have

$$\begin{aligned} \bar{\mathbf{F}}_n(4) &= \sum_\alpha e_{\alpha,1} h_\alpha^{\text{eq}}(0,t) + \tau_\kappa^* \left[\sum_\alpha e_{\alpha,1} h_\alpha^{\text{eq}}(-\mathbf{e}_\alpha \delta t, t - \delta t) \right. \\ &\quad \left. - \sum_\alpha e_{\alpha,1} h_\alpha^{\text{eq}}(0,t) \right] \\ &= \mathbf{F}^I(4) + \tau_\kappa^*[\mathbf{F}^{II}(4) - \mathbf{F}^I(4)]. \end{aligned} \quad (34)$$

To compute $\mathbf{F}^I(4)$ and $\mathbf{F}^{II}(4)$, the flow velocity and temperature at the surrounding points of the cell interface have to be determined in advance. For any variable ϕ , its value at the surrounding points of the cell interface can be computed by

$$\phi(-\mathbf{e}_\alpha \delta t, t - \delta t) = \begin{cases} \phi^L - \nabla \phi^L \cdot \mathbf{e}_\alpha \delta t & \text{if } e_{\alpha,1} > 0, \\ \phi^R - \nabla \phi^R \cdot \mathbf{e}_\alpha \delta t & \text{if } e_{\alpha,1} < 0, \\ \phi^M - \nabla \phi^M \cdot \mathbf{e}_\alpha \delta t & \text{if } e_{\alpha,1} = 0, \end{cases} \quad (35)$$

where ϕ represents the variables u_1, u_2 , and T . ϕ^L and ϕ^R are the values of ϕ at the left and right sides of the cell interface. $\nabla \phi^L$ and $\nabla \phi^R$ are the first-order derivatives of ϕ at the left and right cells around the cell interface, respectively. The superscript *M* denotes the average value at the cell interface, i.e., $\phi^M = (\phi^L + \phi^R)/2$ and $\nabla \phi^M = (\nabla \phi^L + \nabla \phi^R)/2$. Once the velocity and temperature at the surrounding points are obtained, the distribution function $h_\alpha^{\text{eq}}(-\mathbf{e}_\alpha \delta t, t - \delta t)$ can be determined by substituting Eq. (35) into Eq. (23).

According to the compatibility condition and Eq. (25), the temperature at the cell interface can be computed by

$$\overline{\mathbf{W}}^{\text{face}}(4) = \sum_{\alpha} h_{\alpha}^{\text{eq}}(-\mathbf{e}_{\alpha}\delta t, t - \delta t). \quad (36)$$

By substituting Eq. (35) into Eq. (36), we have

$$\begin{aligned} \overline{\mathbf{W}}^{\text{face}}(4) = & \left[\frac{1}{4} \left(T - \frac{\partial T}{\partial x_1} \delta t \right) \left(1 + 2u_1 - 2 \frac{\partial u_1}{\partial x_1} \delta t \right) \right]^L \\ & + \left[\frac{1}{4} \left(T + \frac{\partial T}{\partial x_1} \delta t \right) \left(1 - 2u_1 - 2 \frac{\partial u_1}{\partial x_1} \delta t \right) \right]^R \\ & + \left[\frac{1}{2} T - T \frac{\partial u_2}{\partial x_2} \delta t - u_2 \frac{\partial T}{\partial x_2} \delta t \right]^M. \end{aligned} \quad (37)$$

By combining Eqs. (B1)–(B3) and Eq. (37), we can obtain all the conservative variables at the cell interface. Then, the flux of the energy equation attributed to the equilibrium distribution function at the cell interface can be calculated by

$$\mathbf{F}^I(4) = \overline{\mathbf{W}}^{\text{face}}(2) \overline{\mathbf{W}}^{\text{face}}(4) \overline{\mathbf{W}}^{\text{face}}(1). \quad (38)$$

The flux attributed to the equilibrium distribution at surrounding points of the cell interface can be computed straightforwardly by substituting Eq. (35) into the definition of $\mathbf{F}^{II}(4)$. After some algebraic manipulations, we have

$$\begin{aligned} \mathbf{F}^{II}(4) = & \left[\frac{1}{4} \left(T - \frac{\partial T}{\partial x_1} \delta t \right) \left(1 + 2u_1 - 2 \frac{\partial u_1}{\partial x_1} \delta t \right) \right]^L \\ & - \left[\frac{1}{4} \left(T + \frac{\partial T}{\partial x_1} \delta t \right) \left(1 - 2u_1 - 2 \frac{\partial u_1}{\partial x_1} \delta t \right) \right]^R. \end{aligned} \quad (39)$$

Furthermore, by substituting Eqs. (38) and (39) into Eq. (34), we can obtain the whole expression for the numerical flux of the energy equation.

IV. NUMERICAL EXAMPLES

In this section, the performances of the developed solver are validated by simulating several incompressible thermal flow problems. In the simulations, the conservative variables at two sides of cell interface are obtained by linear interpolation from those at cell centers. This implementation has been validated to have second-order accuracy in space for isothermal incompressible flows [41]. In fact, the calculation of distribution function at the cell interface [Eqs. (28) and (33)] was proved to be of second-order accuracy in both space and time in our previous work [45]. Thus, the accuracy of the present scheme could be affected by the interpolation technique and temporal discretization. For the temporal discretization to solve Eq. (5), the explicit Euler method is applied, which is inherently first-order accurate in time.

Since the macroscopic governing equations are actually solved in the present method, the boundary condition can be implemented based on the flow variables by creating two ghost cells. For the adiabatic wall condition, the flow variables in the ghost cells can be taken as

$$\begin{aligned} \rho_{-i} &= \rho_i, \quad T_{-i} = T_i, \quad i = 1, 2, \\ u_{-i} &= -u_i, \quad v_{-i} = -v_i, \quad i = 1, 2, \end{aligned} \quad (40)$$

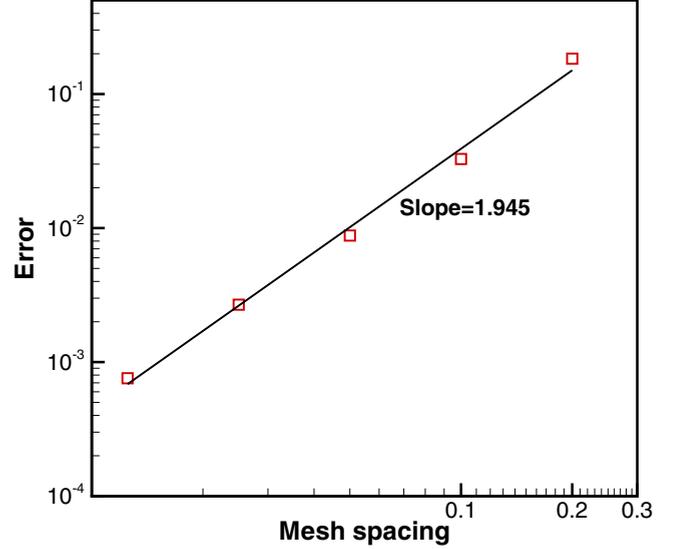


FIG. 3. Error of Nusselt number versus mesh spacing for Rayleigh-Bénard convection at $Ra = 5 \times 10^3$.

where the subscripts -1 and -2 represent the first and second ghost cells and the subscripts 1 and 2 denote the first and second cells adjacent to the boundary. For the isothermal wall condition, the density and velocity in the ghost cells can be calculated the same way as the adiabatic wall condition, while the temperature should be determined by

$$T_{-i} = 2T_b - T_i, \quad i = 1, 2, \quad (41)$$

where T_b is the temperature at the boundary. For the far-field boundary, the characteristic-based boundary condition is utilized. Specifically, for the inflow boundary condition, the velocity and temperature in the ghost cells are determined from the free-stream state and the density is calculated by interpolation from those at the interior cells. For the outflow

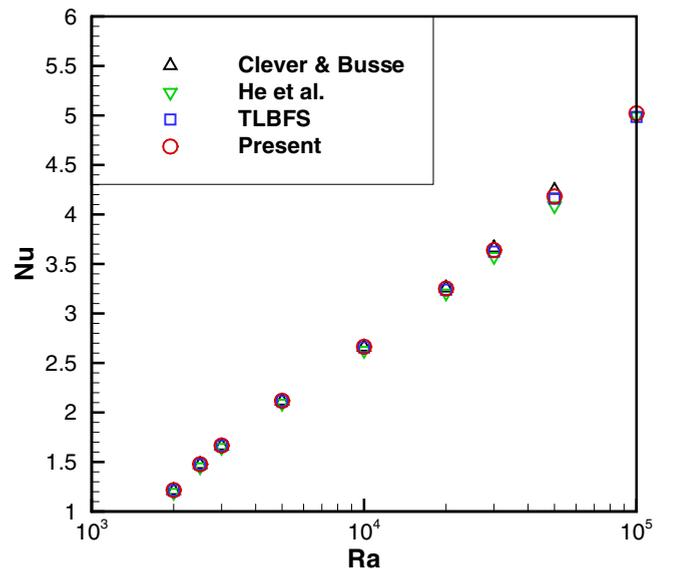


FIG. 4. The dependence of Nusselt number on Rayleigh number for Rayleigh-Bénard convection.

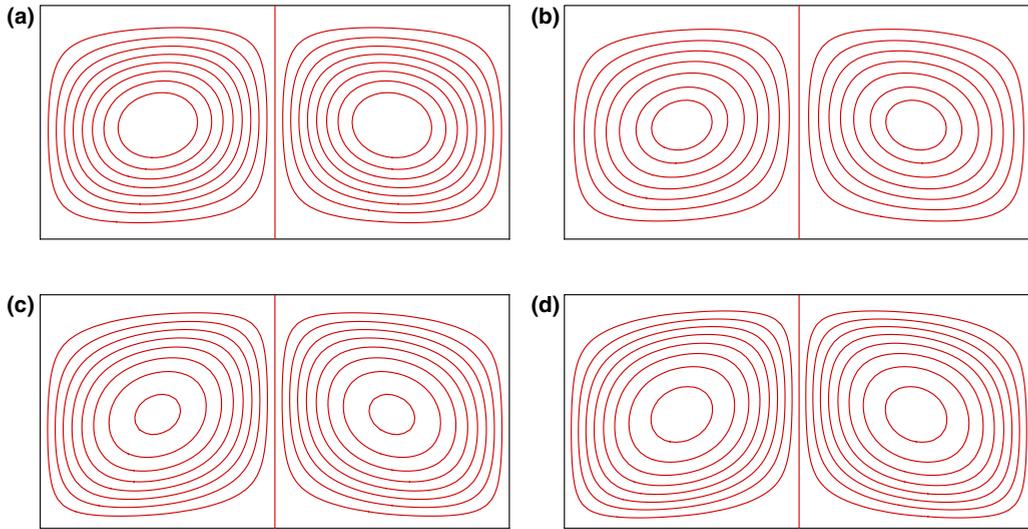


FIG. 5. Streamlines for Rayleigh-Bénard convection at different Rayleigh numbers. (a) $Ra = 5 \times 10^3$, (b) $Ra = 10^4$, (c) $Ra = 5 \times 10^4$, (d) $Ra = 10^5$.

boundary condition, the velocity and temperature in the ghost cells are interpolated from the interior cells and the density is set as the free-stream state.

In all simulations, the convergence criterion is set to be

$$\text{Error} = \max(V_{\text{error}}, T_{\text{error}}) < 1 \times 10^{-8}, \quad (42)$$

where V_{error} and T_{error} are respectively the relative errors of velocity field and temperature field, which are defined as

$$V_{\text{error}} = \frac{\sum_{N_x \times N_y} |(\sqrt{u^2 + v^2})^{n+1} - (\sqrt{u^2 + v^2})^n|}{\sum_{N_x \times N_y} (\sqrt{u^2 + v^2})^{n+1}}, \quad (43a)$$

$$T_{\text{error}} = \frac{\sum_{N_x \times N_y} |T^{n+1} - T^n|}{\sum_{N_x \times N_y} T^{n+1}}. \quad (43b)$$

In order to compare the solution accuracy and computational efficiency of the present scheme with those of TLBFS

[25–26], all the configurations of the two schemes are taken to be the same except for the calculation of numerical fluxes at the cell interface. In addition, all the computations were done on a PC with an Intel® Xeon® processor E5-2643 CPU at 3.3 GHz, and no parallel computation is adopted here.

A. Case 1: Rayleigh-Bénard convection

A good benchmark test for the developed solver is Rayleigh-Bénard convection, in which a viscous fluid between two horizontal walls is heated from the bottom while the top is maintained at a lower temperature. The temperatures at the bottom and top walls are set as $T_1 = 1$ and $T_0 = 0$, with the difference $\Delta T = 1$. The dynamic similarity of this test example depends on two dimensionless parameters: the Prandtl number Pr and the Rayleigh number Ra . They are respectively defined as

$$Pr = \frac{\nu}{\kappa}, \quad Ra = \frac{g\beta\Delta TH^3}{\nu\kappa} = \frac{V_c^2 H^2}{\nu\kappa}, \quad (44)$$

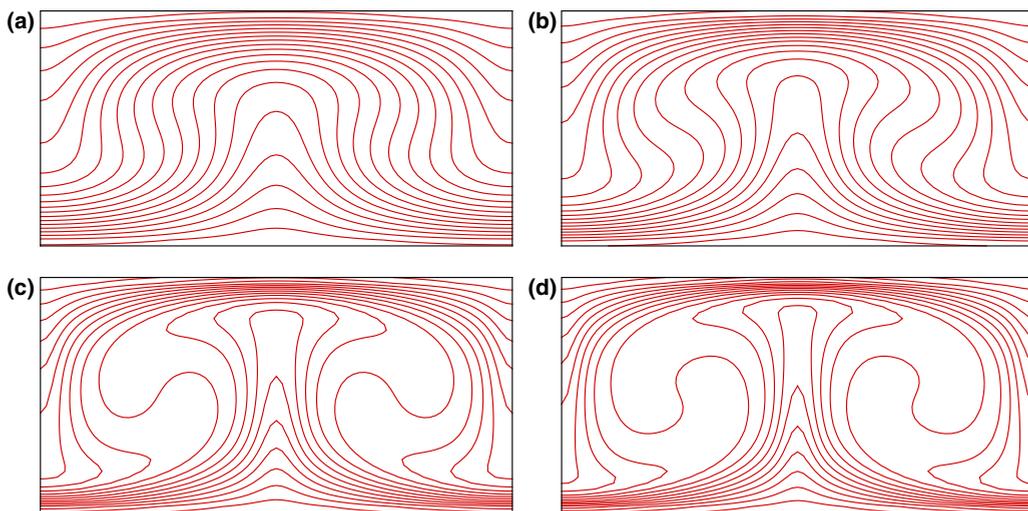


FIG. 6. Isotherms for Rayleigh-Bénard convection at different Rayleigh numbers. (a) $Ra = 5 \times 10^3$, (b) $Ra = 10^4$, (c) $Ra = 5 \times 10^4$, (d) $Ra = 10^5$.

TABLE I. Comparison of computational time (minutes) of TLBFS and the present scheme for Rayleigh-Bénard convection at different Rayleigh numbers.

Schemes	$Ra = 5 \times 10^3$	$Ra = 10^4$	$Ra = 5 \times 10^4$	$Ra = 10^5$
TLBFS	3.9234	3.8511	6.8529	11.2126
Present	2.8538	2.8434	5.2520	8.1157
Present/TLBFS	0.7274	0.7383	0.7664	0.7238

where ν is the kinematic viscosity. H is the vertical length scale of the test case, which is chosen as the characteristic length; $V_c = \sqrt{g\beta H\Delta T}$ is the characteristic thermal velocity. We chose $Pr = 0.71$ and $V_c = 0.1$ in the present calculations. In addition, the horizontal and vertical length scales are taken as $L = 2$ and $H = 1$, respectively. Moreover, no-slip boundary conditions are implemented at the bottom and top walls, and periodic boundary conditions are applied on the left and right boundaries.

For this test case, there exists a critical Rayleigh number $Ra_c = 1707.76$, which can be obtained by linear stability theory [46]. At Rayleigh numbers below Ra_c , a static solution with zero velocity everywhere and a linear function of the vertical coordinate for temperature exists for this problem, while at

Rayleigh numbers above Ra_c , the static conduction becomes unstable to any small disturbance and the system becomes convective. In order to invoke the convective phenomenon, the Rayleigh numbers $Ra = 2 \times 10^3$, 2.5×10^3 , 3×10^3 , 5×10^3 , 10^4 , 2×10^4 , 3×10^4 , 5×10^4 , and 10^5 are considered in our work. The heat transfer between the top and bottom walls can be described by the Nusselt number, which is defined as

$$Nu = 1 + \frac{\langle vT \rangle}{k\Delta T/H}, \quad (45)$$

where $\langle vT \rangle$ represents the average of product of the vertical velocity and temperature over the whole flow domain. First, the test case of $Ra = 5 \times 10^3$ is simulated to validate the overall accuracy of the present method in space. In the simulation, the computational domain is divided by six different uniform grids, $N_x \times N_y = 10 \times 5$, 20×10 , 40×20 , 80×40 , 160×80 , and 320×160 . The numerical error is defined as the absolute value of the difference between the final steady value of Nu for the result of $N_x \times N_y = 320 \times 160$ (this result is considered as the benchmark datum) and that of each resolution. Figure 3 shows the numerical error versus mesh spacing (Δx) in logarithmic scale. It can be seen that the overall accuracy of the present scheme is about second order in space.

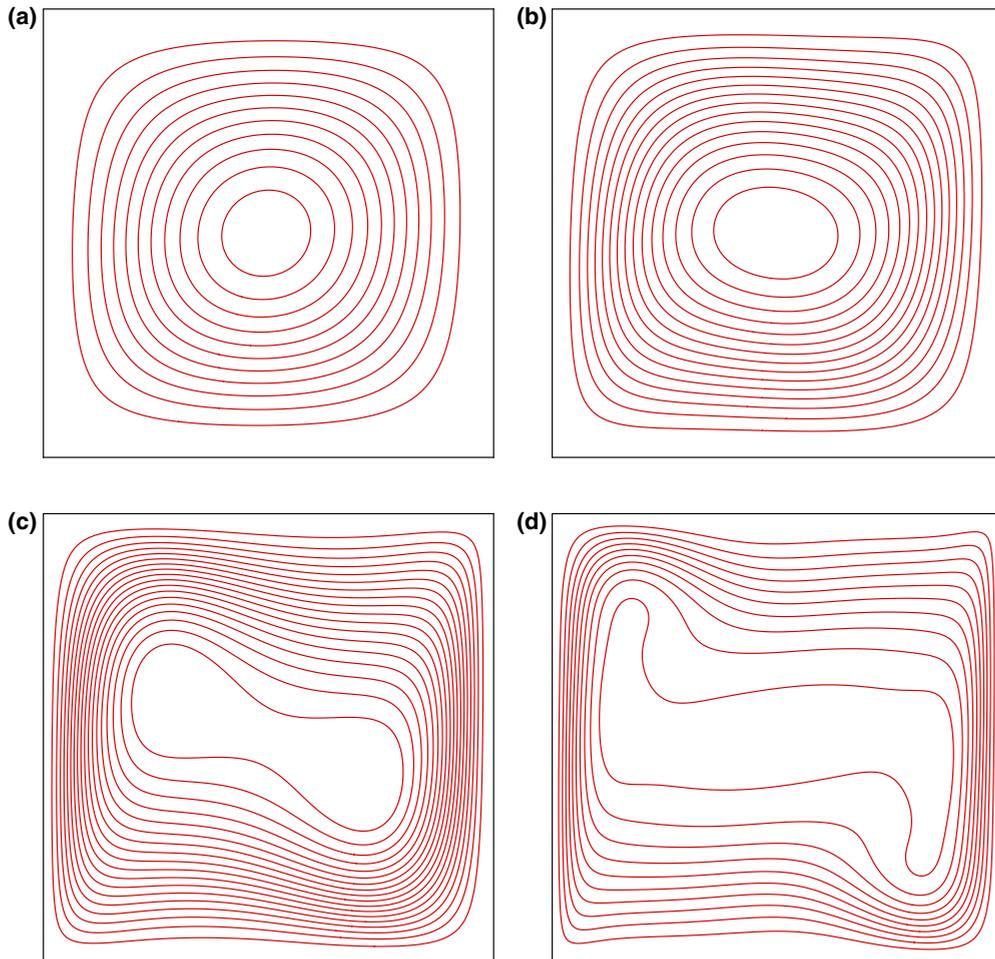


FIG. 7. Streamlines for natural convection in a square cavity at four different Rayleigh numbers. (a) $Ra = 10^3$, (b) $Ra = 10^4$, (c) $Ra = 10^5$, (d) $Ra = 10^6$.

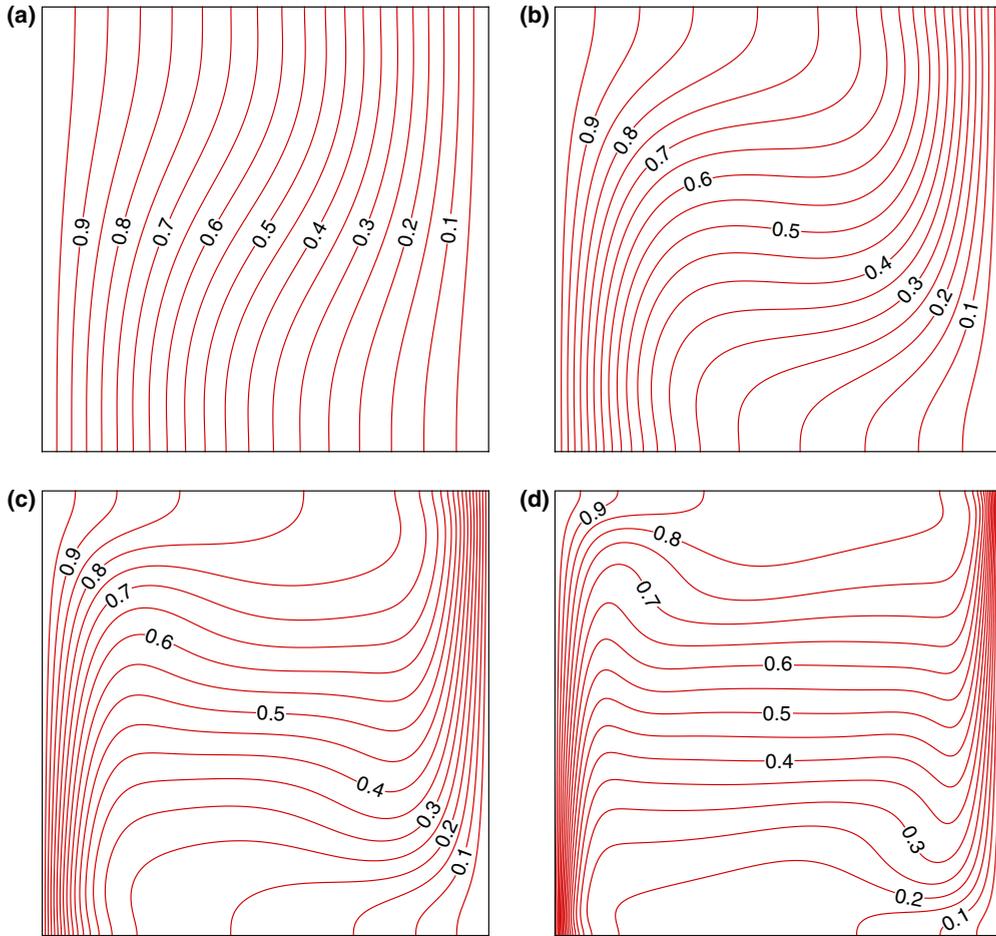


FIG. 8. Isotherms for natural convection in a square cavity at four different Rayleigh numbers. (a) $Ra = 10^3$, (b) $Ra = 10^4$, (c) $Ra = 10^5$, (d) $Ra = 10^6$.

Second, the uniform mesh with $N_x \times N_y = 80 \times 40$ cells is used for simulation of this test case at different Rayleigh numbers. Figure 4 shows the relationship between the Nusselt number and the Rayleigh number obtained by the present scheme and TLBFS. Also displayed in this figure are the simulation results of Clever and Busse [47] and He *et al.* [13]. It can be clearly seen that good agreements are achieved. The computed streamlines and isotherms of the Rayleigh-Bénard convection at four different Rayleigh numbers are shown in Figs. 5 and 6, respectively. It can be observed that the heat transfer in the box is enhanced as the Rayleigh number increases. In addition, in terms of computational efficiency, Table I shows that the present method takes at most 77% of the computational time needed by TLBFS. This implies that the present scheme is more efficient than the TLBFS. The reason for this may be that a large number of interpolations are required for the TLBFS to calculate the nonequilibrium term of the distribution function at the cell interface since the D2Q9 lattice velocity model is utilized, while the numerical fluxes of the present scheme are given explicitly as the function of flow variables and their derivatives.

B. Case 2: Natural convection in a square cavity

The second test case is natural convection in a square cavity, which is utilized to validate the present solver for simulation

of incompressible thermal flows at a wide range of Rayleigh numbers. In this test example, the no-slip boundary condition is applied on all walls. The adiabatic condition is set on the top and bottom walls, while isothermal conditions with fixed temperatures of $T_1 = 1$ and $T_0 = 0$ are respectively applied to

TABLE II. Results of natural convection in a square cavity at four different Rayleigh numbers: $Ra = 10^3, 10^4, 10^5$, and 10^6 .

Schemes		$Ra = 10^3$	$Ra = 10^4$	$Ra = 10^5$	$Ra = 10^6$
u_{max}	DQ [48]	3.649	16.190	34.736	64.775
	TLBFS [25]	3.640	16.14	34.87	64.838
	Present	3.647	16.183	34.775	64.938
y	DQ [48]	0.815	0.825	0.855	0.850
	TLBFS [25]	0.815	0.825	0.855	0.850
	Present	0.815	0.823	0.853	0.850
v_{max}	DQ [48]	3.698	19.638	68.640	220.64
	TLBFS [25]	3.708	19.67	68.85	220.92
	Present	3.696	19.627	68.634	220.67
x	DQ [48]	0.180	0.120	0.065	0.035
	TLBFS [25]	0.180	0.118	0.065	0.038
	Present	0.175	0.117	0.067	0.038
\overline{Nu}	DQ [48]	1.118	2.245	4.523	8.762
	TLBFS [25]	1.115	2.232	4.491	8.711
	Present	1.118	2.245	4.524	8.835

TABLE III. Comparison of computational time (hours) of TLBFS and the present scheme for natural convection in a square cavity at different Rayleigh numbers.

Schemes	Ra = 10 ³	Ra = 10 ⁴	Ra = 10 ⁵	Ra = 10 ⁶
TLBFS	0.2652	1.8068	7.7501	31.4062
Present	0.1682	1.3596	5.9587	23.9196
Present/TLBFS	0.6342	0.7525	0.7689	0.7616

the left and right walls. In this test case, the vertical length is equal to the horizontal length, i.e., $H = L = 1$. The dynamic similarity of this test example depends on two dimensionless parameters: the Prandtl number Pr and the Rayleigh number Ra , which are defined the same as in Eq. (44). In the simulation, the Nusselt number Nu is used to evaluate the heat transfer rate. The averaged Nusselt number over the whole computational domain and that along the line of $x = L/2$ are respectively defined as

$$\overline{Nu} = \frac{L}{\kappa \Delta T} \frac{1}{L^2} \iint_{\Omega} \left(uT - \kappa \frac{\partial T}{\partial x} \right) d\Omega, \quad (46)$$

$$Nu_{1/2} = \frac{L}{\kappa \Delta T} \frac{1}{L} \int_{x=L/2} \left(uT - \kappa \frac{\partial T}{\partial x} \right) dl. \quad (47)$$

In the present study, we set $Pr = 0.71$ and $V_c = 0.1$. The simulated Rayleigh number is changed from 10^3 to 10^8 . It should be noted that the natural convection at high Rayleigh numbers ($Ra = 10^7$ and 10^8) is a challenging test case. It provides a good chance to examine the performance of the present solver.

First, the test case of $Ra = 10^3, 10^4, 10^5$, and 10^6 is simulated on uniform grids. In the present simulation, four different uniform grids with $100 \times 100, 150 \times 150, 200 \times 200$, and 250×250 cells are respectively used for $Ra = 10^3, 10^4, 10^5$ and 10^6 , as suggested by Peng *et al.* [14]. Figures 7 and 8 respectively show the streamlines and isotherms computed by the present scheme. These plots agree well with those obtained by Peng *et al.* [14] using the TLBM, Wang *et al.* [25] adopting the TLBFS, and Shu and Xue [48] applying the high order differential quadrature (DQ) method. In addition, Table II quantitatively compares the computed representative properties with those of Wang *et al.* [25] and Shu and Xue [48]. In this table, u_{\max} is the maximum u velocity at the vertical centerline and y is the y coordinate of the corresponding location, v_{\max} is the maximum v velocity at the horizontal centerline, and x is the x coordinate of the corresponding location. It can be seen that the present results show good agreement with those of the previous studies [25,48]. Moreover, Table III compares the computational cost of TLBFS and the present scheme at four

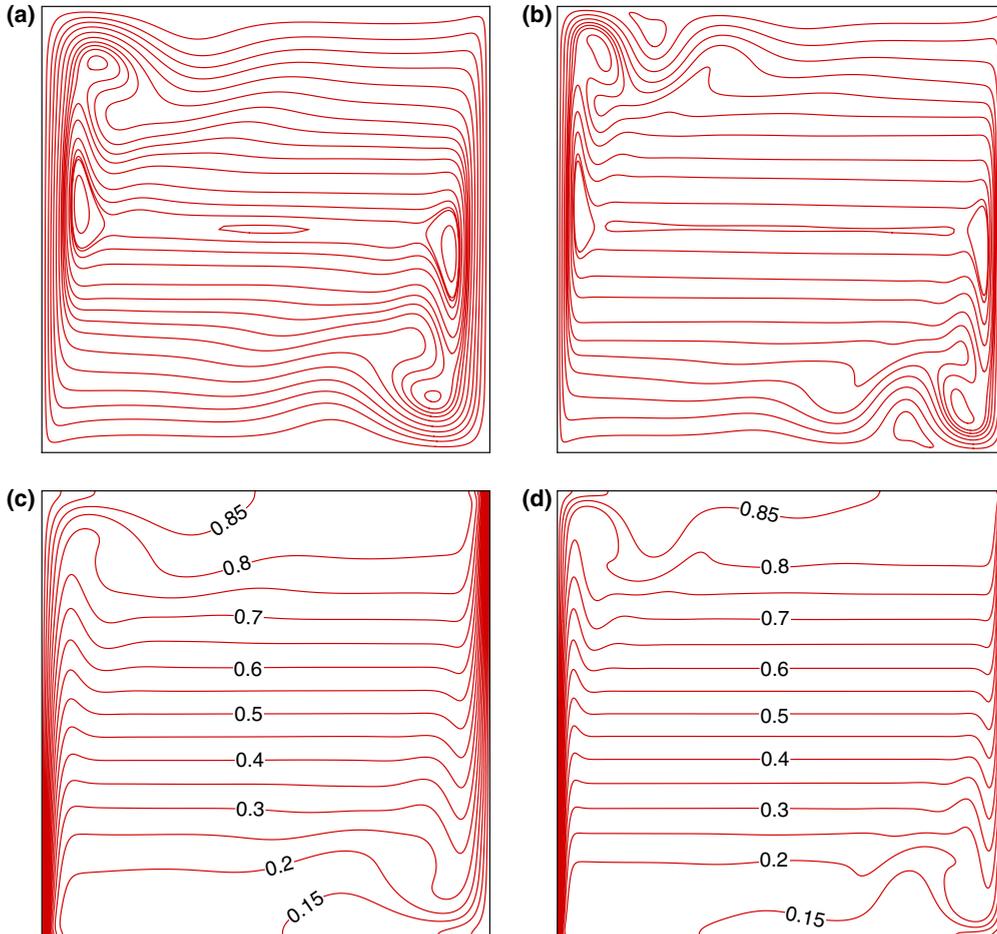


FIG. 9. Simulation results for natural convection in a square cavity. (a) Streamlines at $Ra = 10^7$, (b) streamlines at $Ra = 10^8$, (c) isotherms at $Ra = 10^7$, (d) isotherms at $Ra = 10^8$.

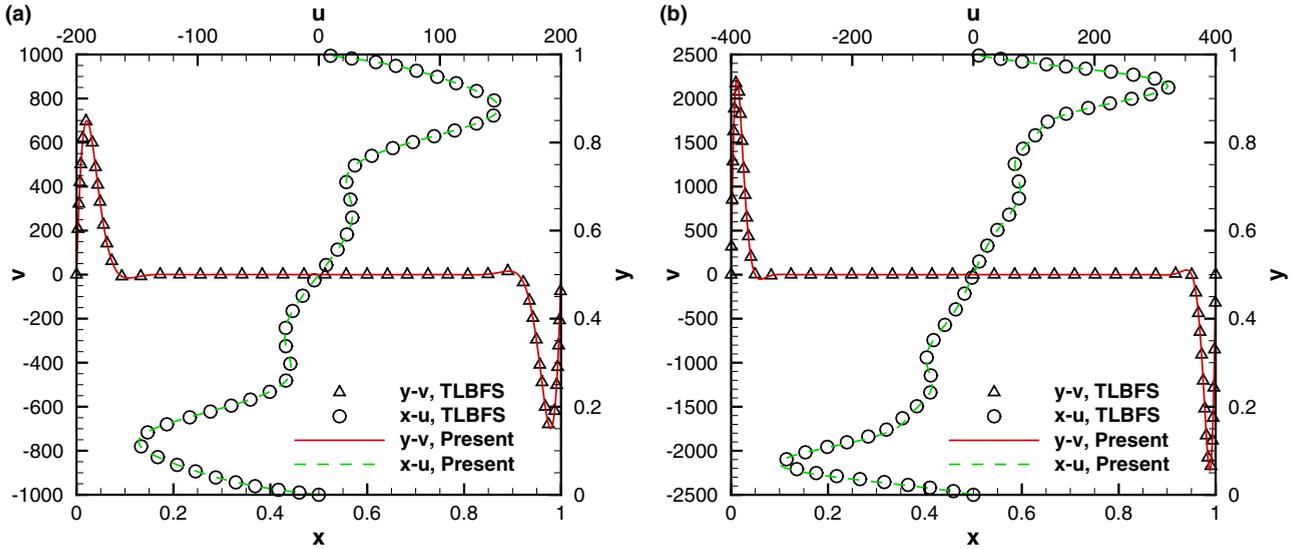


FIG. 10. Comparison of velocity profiles at the centerlines for natural convection in a square cavity. (a) $Ra = 10^7$, (b) $Ra = 10^8$.

different Rayleigh numbers. The high efficiency of the present method is clearly demonstrated.

Second, the test case of $Ra = 10^7$ and 10^8 is solved on nonuniform grids. For $Ra = 10^7$, a nonuniform grid with 301×301 points is used, and the distance from the wall to the nearest grid point in the computational domain is taken as 0.0004, while for $Ra = 10^8$, the computational grid is set as 401×401 , and the distance from the wall to the nearest grid point in the computational domain is chosen as 0.0001. The mesh stretching transformation near the wall is controlled by a hyperbolic tangent function. The computed streamlines and isotherms at $Ra = 10^7$ and 10^8 are depicted in Fig. 9. As can be seen from the figure, both the flow and temperature boundary layers close to the hot and cold walls are very thin compared with the test case at moderate Rayleigh numbers ($Ra = 10^3, 10^4, 10^5$, and 10^6). Furthermore, vertical convection in the central area becomes very weak and heat conduction dominates this region. This observation can be further validated through Fig. 10, which shows the u velocity along the vertical centerline and v velocity along the horizontal centerline. Also displayed in this figure are the results of Wang *et al.* [44] obtained by the TLBFS. Clearly, good agreements are achieved. Detailed comparisons of representative proper-

ties at $Ra = 10^7$ and 10^8 are respectively listed in Tables IV and V. In these tables, $|\varphi|_{\max}$ is the maximum stream function in the whole computational domain, and x and y below $|\varphi|_{\max}$ are the coordinates of the corresponding location. Also listed in these tables are the simulation results of Contrino *et al.* [49] obtained by the thermal lattice Boltzmann equation with the multiple-relaxation-times collision model (MRT-TLBE) and Qu er e [50] calculated by the high order pseudospectral method. Once again, the present results compare well with the published data [44,49–50]. Note that, since a nonuniform grid is utilized, relatively fewer grid points are required in our simulation as compared with the MRT-TLBE.

C. Case 3: Natural convection in a concentric annulus

To illustrate the capability of the present scheme for solving thermal flow problems with curved boundaries, we present a numerical test of the natural convection in a concentric annulus [25,51–53] in this subsection. The configuration of the problem is illustrated in Fig. 11. Basically, it is an internal natural convection problem bounded by a hot outer cylinder with radius R_o and temperature $T_o = 1$ and a cold inner cylinder with radius R_i and temperature $T_i = 0$. The dynamic similarity of this

TABLE IV. Results of natural convection in a square cavity at $Ra = 10^7$.

Parameters	Contrino <i>et al.</i> [49]	Qu�er�e [50]	TLBFS [44]	Present
Mesh	1531^2		301^2	301^2
$ \varphi _{\max}$	30.185	30.165	30.164	30.192
x	0.0857	0.086	0.0857	0.0848
y	0.5559	0.556	0.5559	0.5548
$Nu_{1/2}$	16.523	16.52	16.543	16.518
u_{\max}	148.58	148.59	148.84	148.86
y	0.8793	0.879	0.8789	0.8800
v_{\max}	699.31	699.18	699.91	699.20
x	0.0213	0.021	0.0216	0.0217

TABLE V. Results of natural convection in a square cavity at $Ra = 10^8$.

Parameters	Contrino <i>et al.</i> [49]	Qu�er�e [50]	TLBFS [44]	Present
Mesh	1531^2		401^2	401^2
$ \varphi _{\max}$	53.953	53.85	53.893	53.885
x	0.0480	0.048	0.0476	0.0476
y	0.5533	0.553	0.5528	0.5532
$Nu_{1/2}$	30.227	30.225	30.301	30.227
u_{\max}	321.37	321.9	323.65	321.49
y	0.9276	0.928	0.9288	0.9284
v_{\max}	2222.3	2222	2222.9	2221.7
x	0.0120	0.012	0.0119	0.0122

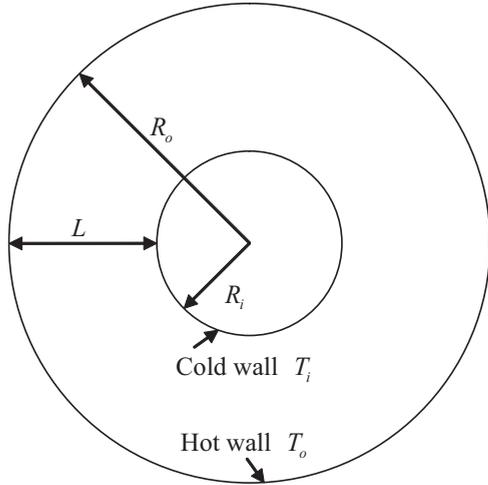


FIG. 11. Illustration of the setup for natural convection in a concentric annulus.

problem is characterized by three dimensionless parameters, i.e., the aspect ratio Ar , the Prandtl number Pr , and the Rayleigh number Ra . Pr and Ra were previously defined in Eq. (44), while the characteristic length H is changed to the distance between the two cylinders L in this test case. The aspect ratio Ar is defined as

$$Ar = \frac{R_o}{R_i}. \quad (48)$$

In addition, to quantify the heat transfer efficiency of this problem, the average equivalent heat conductivities on two cylinders are defined as follows [25,51]:

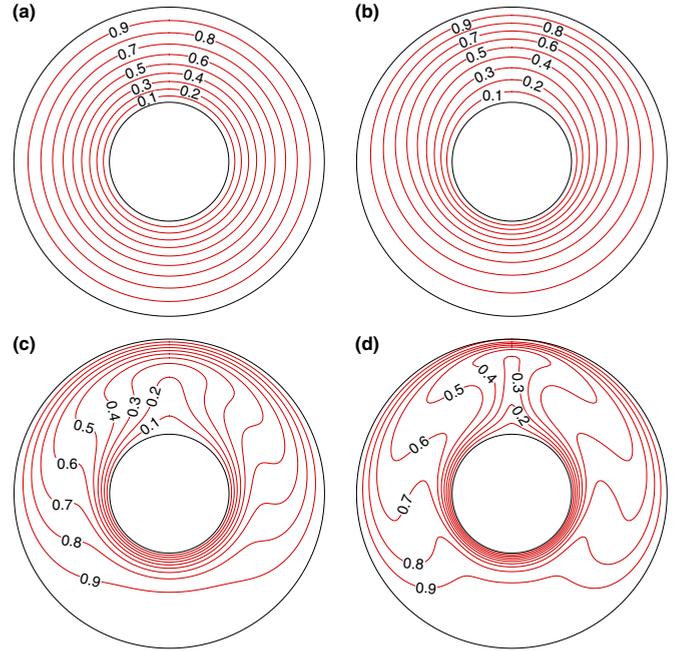


FIG. 13. Isotherms for natural convection in a concentric annulus at different Rayleigh numbers. (a) $Ra = 10^2$, (b) $Ra = 10^3$, (c) $Ra = 10^4$, (d) $Ra = 5 \times 10^4$.

$$\overline{k_{eqi}} = \frac{\ln(Ar)}{2\pi(Ar - 1)} \int_0^{2\pi} \frac{\partial T}{\partial r} d\theta \text{ for the inner cylinder,} \quad (49)$$

$$\overline{k_{eqo}} = \frac{Ar \ln(Ar)}{2\pi(Ar - 1)} \int_0^{2\pi} \frac{\partial T}{\partial r} d\theta \text{ for the outer cylinder.} \quad (50)$$

In the present study, we choose $Ar = 2.6$ and $Pr = 0.71$. Six cases with various Rayleigh numbers of $Ra = 10^2, 10^3, 3 \times 10^3, 6 \times 10^3, 10^4$, and 5×10^4 are carried out on a body-fitted O-type mesh with size 251×61 , and the distance from the inner wall to the nearest grid point in the computational domain is taken as 0.018.

Figures 12 and 13 respectively show the streamlines and isotherms at different Rayleigh numbers in our simulations. As can be seen from these figures, at lower Rayleigh number

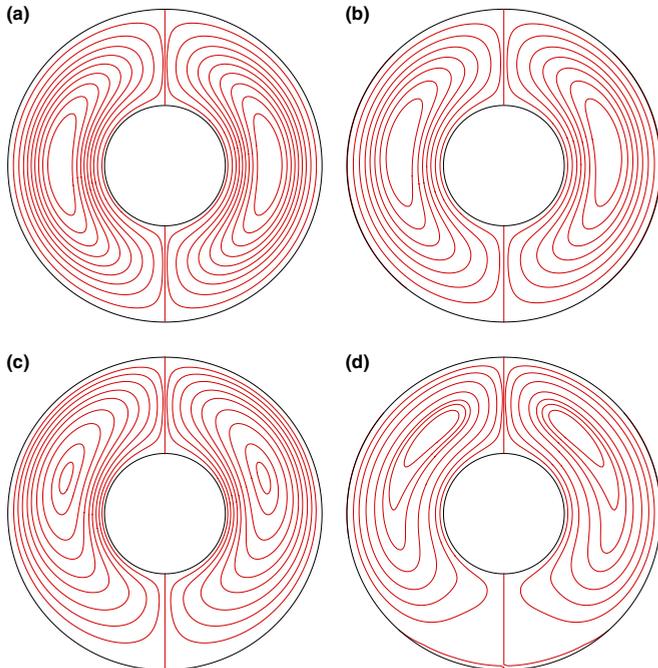


FIG. 12. Streamlines for natural convection in a concentric annulus at different Rayleigh numbers. (a) $Ra = 10^2$, (b) $Ra = 10^3$, (c) $Ra = 10^4$, (d) $Ra = 5 \times 10^4$.

TABLE VI. Comparison of average equivalent heat conductivity for natural convection in a concentric annulus at different Rayleigh numbers.

Ra	Inner cylinder, $\overline{k_{eqi}}$				Outer cylinder, $\overline{k_{eqo}}$			
	Kuehn		Shu TLBFS		Kuehn		Shu TLBFS	
	[52]	[51]	[25]	Present	[52]	[51]	[25]	Present
10^2	1.000	1.001	1.002	1.001	1.002	1.001	1.002	1.001
10^3	1.081	1.082	1.076	1.079	1.084	1.082	1.078	1.080
3×10^3	1.404	1.397	1.381	1.390	1.402	1.397	1.384	1.392
6×10^3	1.736	1.715	1.695	1.707	1.735	1.715	1.701	1.709
10^4	2.010	1.979	1.960	1.970	2.005	1.979	1.960	1.972
5×10^4	3.024	2.958	2.941	2.949	2.973	2.958	2.941	2.953

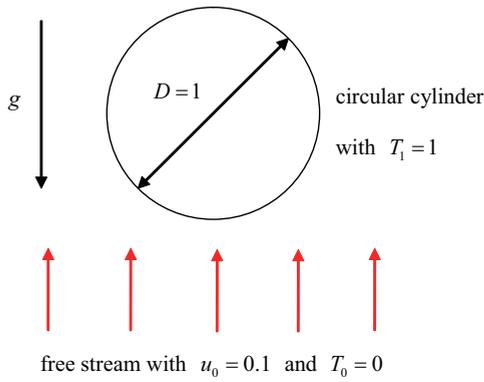


FIG. 14. Illustration of the setup for mixed heat transfer from a heated circular cylinder.

($Ra = 10^2$), both the streamlines and isotherms are almost symmetric about the x and y axes. This means that the thermal conduction dominates the heat transfer and the heat transfer efficiency is relatively low in this case. As the Rayleigh

TABLE VII. Comparison of computational time (hours) of TLBFS and the present scheme for natural convection in a concentric annulus at different Rayleigh numbers.

Schemes	$Ra = 10^2$	$Ra = 10^3$	$Ra = 10^4$	$Ra = 5 \times 10^4$
TLBFS	0.5631	1.0768	2.6920	4.0346
Present	0.4218	0.8027	2.0240	3.1278
Present/TLBFS	0.7491	0.7454	0.7519	0.7752

number is increased, the vortex centers of the flow field move upwards gradually and the isotherms are squeezed closer to the boundaries. This indicates that thermal convection dominates the heat transfer and the heat transfer efficiency is relatively high at large Rayleigh number. These observations can also be quantitatively verified via Table VI, in which the converged results of the average equivalent heat conductivities are compared with the data given by Shu [51], Kuehn and Goldstein [52], and Wang *et al.* [25]. It can be observed that the average equivalent conductivities on both inner and

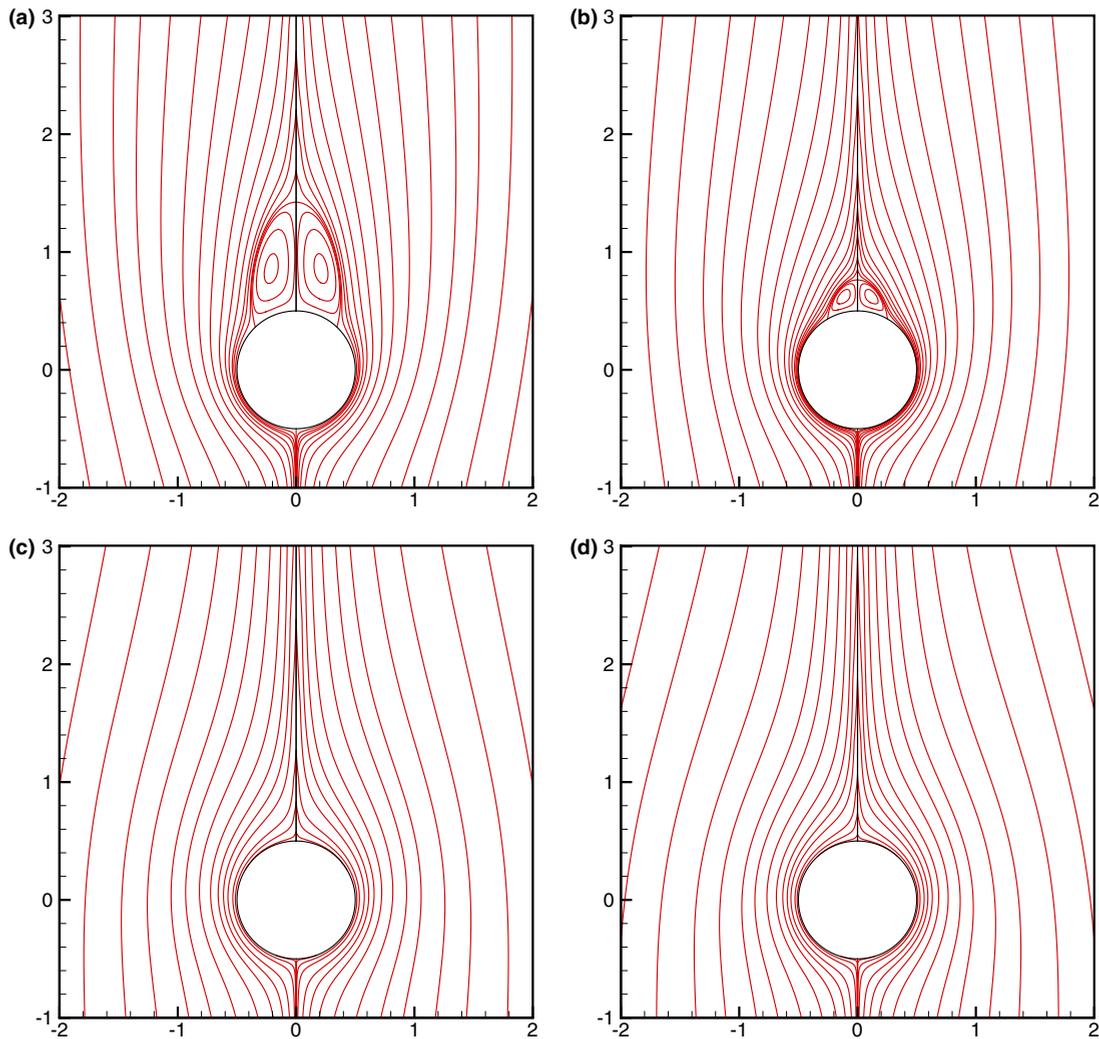


FIG. 15. Streamlines for mixed convection at $Re = 20$ and various Gr . (a) $Gr = 0$, (b) $Gr = 100$, (c) $Gr = 800$, (d) $Gr = 1600$.

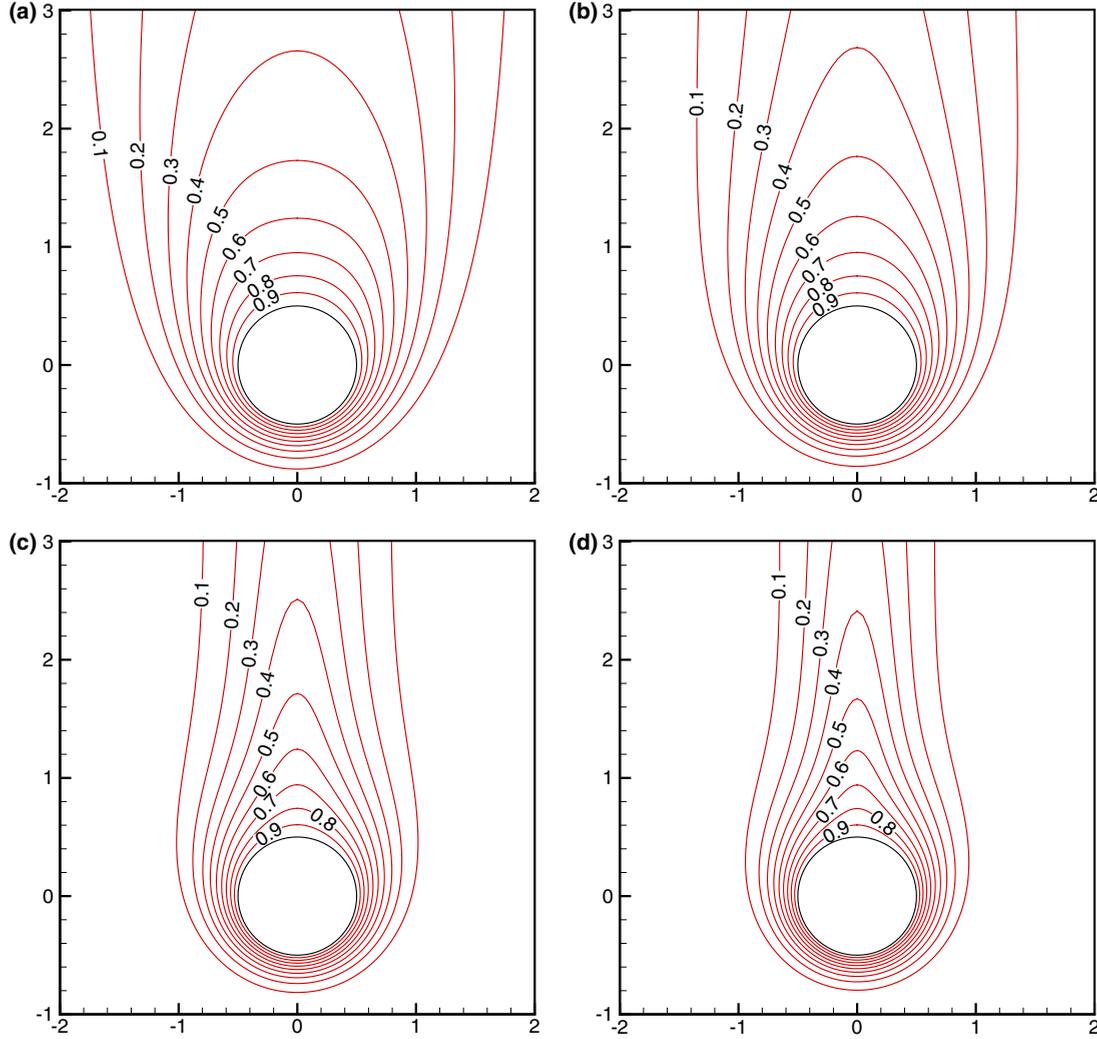


FIG. 16. Isotherms for mixed convection at $Re = 20$ and various Gr . (a) $Gr = 0$, (b) $Gr = 100$, (c) $Gr = 800$, (d) $Gr = 1600$.

outer cylinders increase gradually as the Rayleigh number is increased. Besides that, we also observe that the computed parameters agree well with the reference data. Finally, we compare the computational cost of TLBFS and the present scheme in Table VII. It clearly shows that the present scheme is more efficient than the TLBFS.

D. Case 4: Mixed heat transfer from a heated circular cylinder

The last test case is the mixed heat transfer from a heated circular cylinder, which is used to examine the capability of present solver in modeling external thermal flows. Since both natural heat convection and forced heat convection are involved, the physical mechanism of this problem is more complicated than the previous three examples. As shown in Fig. 14, a stationary heated circular cylinder with the diameter $D = 1$ and temperature $T_1 = 1$ is placed in an incoming fluid flow with free-stream velocity $u_0 = 0.1$ and free-stream temperature $T_0 = 0$. Apart from the Prandtl number defined in Eq. (44) with the characteristic length D , the Reynolds number Re , Grashof number Gr , and average Nusselt number \overline{Nu} are usually introduced in order to better describe the

mixed convection problem. These dimensionless parameters are defined as follows:

$$Re = \frac{u_0 D}{\nu}, \quad (51)$$

$$Gr = \frac{g\beta(T_1 - T_0)D^3}{\nu^2}, \quad (52)$$

$$\overline{Nu} = \frac{D}{2\pi(T_1 - T_0)} \int_0^{2\pi} \left. \frac{\partial T}{\partial n} \right|_w d\theta. \quad (53)$$

In the present simulation, we choose $Re = 20$ and $Pr = 0.7$ to keep in accordance with the initial setup in Refs. [25,54]. Four cases with various Grashof numbers of $Gr = 0, 100, 800,$ and 1600 are simulated on a body-fitted O-type mesh. The computational grid is taken as 301×201 , and the far-field boundary is put at 25.5 diameters away from the center of the cylinder. The distance from the wall to the nearest grid point in the computational domain is taken as 0.005.

The computed streamlines and isotherms at four different Grashof numbers are plotted in Figs. 15 and 16, respectively. Since the temperature field has no effect on the flow field at

TABLE VIII. Comparison of average Nusselt number and separation angle on a circular cylinder for mixed convection at $Re = 20$ and various Gr .

Gr	\overline{Nu}			$\theta_s(\text{deg})$		
	Badr [54]	TLBFS [25]	Present	Badr [54]	TLBFS [25]	Present
0	2.540	2.523	2.454	43.13	43.19	43.59
100	2.654	2.640	2.655	29.51	29.73	30.01
800	3.227	3.208	3.201	0.0	0.0	0.0
1600	3.564	3.554	3.508	0.0	0.0	0.0

zero Grashof number, clear recirculation zones are captured and the streamlines obtained are the same as those in isothermal simulations at the same Reynolds number. As the Grashof number is increased, the separation angle and the length of the vortex behind the circular cylinder decreases gradually until the vortex disappears at $Gr \geq 800$. This phenomenon means that the viscous effect is weakened by the heat convection as the Grashof number increases. In the isotherm figure, it can be seen that the isotherms are squeezed closer to the cylinder surface at high Grashof number. Hence, high heat transfer rates could be expected in such cases. In addition, Table VIII quantitatively compares the computed average Nusselt number and separation angle on the circular cylinder at different Grashof numbers with the results of Badr [54] and Wang *et al.* [25]. As expected, at higher Grashof number, a larger average Nusselt number is obtained in our simulations. In the meantime, the results of the present solver match very well with the published data [25,54].

V. CONCLUSIONS

This work presents an efficient gas kinetic scheme for simulation of incompressible thermal flows. In the scheme, two distribution functions are introduced to evaluate the numerical fluxes at the cell interface. One is the circular function in the gas kinetic scheme, and another one is the distribution function in LBE with the use of the D2Q4 lattice velocity model. The circular function is used to calculate the numerical fluxes of mass and momentum equations, and the D2Q4 model is utilized to compute the numerical flux of the energy equation. By using the difference of equilibrium distribution functions at the cell interface and its surrounding points to approximate the nonequilibrium distribution function, and at the same time considering the incompressible limit, the formulations for the conservative variables and numerical fluxes at the cell interface can be given concisely and explicitly. The effect of temperature field on the flow field is taken into account by the buoyancy force exerted on the momentum equation. It can be treated as a source term and calculated straightforwardly from the conservative variables at cell centers.

To validate the accuracy, efficiency, and stability of the present scheme, several numerical examples, including Rayleigh-Bénard convection, natural convection in a square cavity, natural convection in a concentric annulus, and mixed heat transfer from a heated circular cylinder, are simulated. Numerical results showed that the incompressible thermal flows can be well simulated by the developed solver. The

numerical stability of present scheme is validated by the test case of natural convection in a square cavity at two high Rayleigh numbers of $Ra = 10^7$ and 10^8 . In terms of the computational efficiency, the developed solver is more efficient than the TLBFS of Wang *et al.* [25]. As shown in this work, the present method takes about 63%–78% of the computational time needed by TLBFS.

ACKNOWLEDGMENTS

The research is supported by the National Research Foundation Singapore, Sembcorp Industries Ltd., and the National University of Singapore under the Sembcorp-NUS Corporate Laboratory.

APPENDIX A: CHAPMAN-ENSKOG ANALYSIS FOR RECOVERING ENERGY EQUATION

By introducing multiscale expansion, the temperature distribution function, temporal derivative, and spatial derivative can be respectively expanded as

$$h_\alpha = h_\alpha^{(0)} + \varepsilon h_\alpha^{(1)} + \varepsilon^2 h_\alpha^{(2)}, \quad (\text{A1})$$

$$\partial_t = \varepsilon \partial_{t1} + \varepsilon^2 \partial_{t2}, \quad (\text{A2})$$

$$\nabla = \varepsilon \nabla_1, \quad (\text{A3})$$

where $h_\alpha^{(0)} = h_\alpha^{\text{eq}}$ is the equilibrium state. ε is a small parameter proportional to the Knudsen number. By substituting Eqs. (A1)–(A3) into Eq. (22), we have the following three equations in terms of ε order:

$$O(\varepsilon^0) : h_\alpha^{\text{eq}} - h_\alpha^{(0)} = 0, \quad (\text{A4})$$

$$O(\varepsilon^1) : (\partial_{t1} + \mathbf{e}_\alpha \cdot \nabla_1) h_\alpha^{(0)} + h_\alpha^{(1)} / \tau_\kappa = 0, \quad (\text{A5})$$

$$O(\varepsilon^2) : \partial_{t2} h_\alpha^{(0)} + (\partial_{t1} + \mathbf{e}_\alpha \cdot \nabla_1) h_\alpha^{(1)} + h_\alpha^{(2)} / \tau_\kappa = 0. \quad (\text{A6})$$

By using the D2Q4 model [Eqs. (23) and (24)] and taking summation of Eqs. (A5) and (A6) about index α , we can get

$$\partial_{t1} T + \nabla_1 \cdot (\mathbf{u} T) = 0, \quad (\text{A7})$$

$$\partial_{t2} T + \nabla_1 \cdot \prod^{(1)} = 0. \quad (\text{A8})$$

According to the Boussinesq approximation [15], $\prod^{(1)}$ can be expressed as

$$\begin{aligned} \prod^{(1)} &= \sum_\alpha \mathbf{e}_\alpha h_\alpha^{(1)} \\ &= -\tau_\kappa \sum_\alpha \mathbf{e}_\alpha (\partial_{t1} + \mathbf{e}_\alpha \cdot \nabla_1) h_\alpha^{(0)} \\ &= -\tau_\kappa \nabla_1 T / 2 + O(\tau_\kappa \text{Ma}^2 \nabla_1 T). \end{aligned} \quad (\text{A9})$$

Here, Ma is the Mach number. Substituting Eq. (A9) into Eq. (A8) and combining Eqs. (A7) and (A8), we have

$$\partial_t T + \nabla \cdot (T \mathbf{u}) = \frac{\tau_\kappa}{2} \nabla^2 T + O(\tau_\kappa Ma^2 \nabla^2 T). \quad (\text{A10})$$

By comparing Eq. (A10) with Eq. (3), it is found that the collision time scale and the thermal diffusivity have the following relationship:

$$\tau_\kappa = 2\kappa. \quad (\text{A11})$$

In addition, the error term for recovering energy equation is $O(\tau_\kappa Ma^2 \nabla^2 T)$. For simulation of incompressible flows, this error term can be neglected.

APPENDIX B: CALCULATION OF $\overline{\mathbf{W}}^{\text{face}}(1:3)$ AND $\mathbf{F}^{II}(1:3)$

As shown in Eq. (29), the numerical fluxes of mass and momentum equations at the cell interface consist of two parts: the flux attributed to the equilibrium distribution function and moments at the cell interface $\mathbf{F}^I(1:3)$ and the flux attributed to the equilibrium distribution function and moments on the circle $\mathbf{F}^{II}(1:3)$. $\mathbf{F}^I(1:3)$ can be directly computed by the conservative variables at the cell interface $\overline{\mathbf{W}}^{\text{face}}(1:3)$, as shown in Eq. (31). Hence, the expression of $\overline{\mathbf{W}}^{\text{face}}(1:3)$ has to be given first. By using Eq. (30), we have

$$\overline{\mathbf{W}}^{\text{face}}(1) = [\pi g_0 + 2g_1]^L + [\pi g_0 - 2g_1]^R, \quad (\text{B1})$$

$$\begin{aligned} \overline{\mathbf{W}}^{\text{face}}(2) &= \left[\pi a_0 g_0 + 2(a_1 g_0 + a_0 g_1) + \frac{\pi}{2}(a_1 g_1 + a_2 g_2) \right]^L \\ &+ \left[\pi a_0 g_0 - 2(a_1 g_0 + a_0 g_1) + \frac{\pi}{2}(a_1 g_1 + a_2 g_2) \right]^R \end{aligned} \quad (\text{B2})$$

$$\begin{aligned} \overline{\mathbf{W}}^{\text{face}}(3) &= \left[\pi b_0 g_0 + 2(b_1 g_0 + b_0 g_1) + \frac{\pi}{2}(b_1 g_1 + b_2 g_2) \right]^L \\ &+ \left[\pi b_0 g_0 - 2(b_1 g_0 + b_0 g_1) + \frac{\pi}{2}(b_1 g_1 + b_2 g_2) \right]^R \end{aligned} \quad (\text{B3})$$

where

$$\begin{aligned} a_0^L &= u_1^L - \frac{\partial u_1^L}{\partial x_1} u_1^+ \delta t - \frac{\partial u_1^L}{\partial x_2} u_2^+ \delta t, \quad a_1^L = c^+ - \frac{\partial u_1^L}{\partial x_1} c^+ \delta t, \\ a_2^L &= -\frac{\partial u_1^L}{\partial x_2} c^+ \delta t, \\ b_0^L &= u_2^L - \frac{\partial u_2^L}{\partial x_1} u_1^+ \delta t - \frac{\partial u_2^L}{\partial x_2} u_2^+ \delta t, \quad b_1^L = -\frac{\partial u_2^L}{\partial x_1} c^+ \delta t, \\ b_2^L &= c^+ - \frac{\partial u_2^L}{\partial x_2} c^+ \delta t, \\ g_0^L &= g_C^L - \frac{\partial g_C^L}{\partial x_1} u_1^+ \delta t - \frac{\partial g_C^L}{\partial x_2} u_2^+ \delta t, \quad g_1^L = -\frac{\partial g_C^L}{\partial x_1} c^+ \delta t, \\ g_2^L &= -\frac{\partial g_C^L}{\partial x_2} c^+ \delta t, \end{aligned}$$

and the coefficients a_i^R , b_i^R , and g_i^R , $i = 0, 1, 2$, can be easily obtained by replacing the superscript L of the above equations with R . Note that these coefficients are only related to the conservative variables and their derivatives at the left and right cells around the cell interface. In addition, the predicted mean flow velocities u_1^+ and u_2^+ can be calculated by Roe average or simple average from those at the left and right sides of cell interface [38–40].

The flux attributed to the equilibrium distribution function and moments on the circle can be computed by Eq. (32). After some algebraic manipulations, we have

$$\mathbf{F}^{II}(1) = \left[\pi a_0 g_0 + 2(a_1 g_0 + a_0 g_1) + \frac{\pi}{2}(a_1 g_1 + a_2 g_2) \right]^L + \left[\pi a_0 g_0 - 2(a_1 g_0 + a_0 g_1) + \frac{\pi}{2}(a_1 g_1 + a_2 g_2) \right]^R, \quad (\text{B4})$$

$$\begin{aligned} \mathbf{F}^{II}(2) &= \left[a_0 g_0 (\pi a_0 + 2a_1) + \left(2a_0 + \frac{\pi a_1}{2} \right) (a_0 g_1 + a_1 g_0) + \frac{\pi a_2}{2} (a_0 g_2 + a_2 g_0) + \left(\frac{\pi a_0}{2} + \frac{4a_1}{3} \right) (a_1 g_1 + a_2 g_2) + \frac{2}{3} a_2 a_2 g_1 \right]^L \\ &+ \left[a_0 g_0 (\pi a_0 - 2a_1) - \left(2a_0 - \frac{\pi a_1}{2} \right) (a_0 g_1 + a_1 g_0) + \frac{\pi a_2}{2} (a_0 g_2 + a_2 g_0) + \left(\frac{\pi a_0}{2} - \frac{4a_1}{3} \right) (a_1 g_1 + a_2 g_2) - \frac{2}{3} a_2 a_2 g_1 \right]^R \end{aligned} \quad (\text{B5})$$

$$\begin{aligned} \mathbf{F}^{II}(3) &= \left[b_0 g_0 (\pi a_0 + 2a_1) + \left(2a_0 + \frac{\pi a_1}{2} \right) (b_0 g_1 + b_1 g_0) + \frac{\pi a_2}{2} (b_0 g_2 + b_2 g_0) + \left(\frac{\pi a_0}{2} + \frac{2a_1}{3} \right) (b_1 g_1 + b_2 g_2) \right. \\ &+ \left. \frac{2}{3} a_2 (b_1 g_2 + b_2 g_1) + \frac{2}{3} a_1 b_1 g_1 \right]^L + \left[b_0 g_0 (\pi a_0 - 2a_1) - \left(2a_0 - \frac{\pi a_1}{2} \right) (b_0 g_1 + b_1 g_0) + \frac{\pi a_2}{2} (b_0 g_2 + b_2 g_0) \right. \\ &+ \left. \left(\frac{\pi a_0}{2} - \frac{2a_1}{3} \right) (b_1 g_1 + b_2 g_2) - \frac{2}{3} a_2 (b_1 g_2 + b_2 g_1) - \frac{2}{3} a_1 b_1 g_1 \right]^R \end{aligned} \quad (\text{B6})$$

Finally, by substituting Eq. (31) and Eqs. (B4)–(B6) into Eq. (29), we can obtain the whole expression of numerical fluxes at the cell interface for the mass and momentum equations.

The last undetermined variable in Eqs. (29) and (34) is the streaming time step δt . The principle for the choice of δt is that the circle in the physical space and the position of discrete velocity points of the D2Q4 model must be within the cell of the interface in order to avoid extrapolation. They are respectively shown in Figs. 1 and 2. A feasible way to determine δt can be expressed as

$$\delta t = \frac{0.4 \times \min\{\Delta l, \Delta r\}}{\max\{u_{\max}^{\text{circle}}, u_{\max}^{\text{D2Q4}}\}}, \quad (\text{B7})$$

where

$$u_{\max}^{\text{circle}} = \max\{u_1^+, u_2^+\} + c^+ \quad \text{and} \quad u_{\max}^{\text{D2Q4}} = |\mathbf{e}_\alpha| = 1.$$

Δl and Δr are the shortest edge lengths of the left and right cells around the cell interface, respectively.

-
- [1] S. Chen and G. D. Doolen, Lattice Boltzmann method for fluid flows, *Annu. Rev. Fluid Mech.* **30**, 329 (1998).
- [2] S. Succi, *The Lattice Boltzmann Equation: For Fluid Dynamics and Beyond* (Oxford University Press, Oxford, 2001).
- [3] M. Karttunen, I. Vattulainen, and A. Lukkarinen, *Novel Methods in Soft Matter Simulations* (Springer, New York, 2004).
- [4] Z. Guo and C. Shu, *Lattice Boltzmann Method and Its Applications in Engineering* (World Scientific, Singapore, 2013).
- [5] K. Xu, Gas-kinetic schemes for unsteady compressible flow simulations, *Lect. Ser. -van Kareman Inst. Fluid Dyn.* **3**, 1 (1998).
- [6] D. Chae, C. Kim, and O. H. Rho, Development of an improved gas-kinetic BGK scheme for inviscid and viscous flows, *J. Comput. Phys.* **158**, 1 (2000).
- [7] G. May, B. Srinivasan, and A. Jameson, An improved gas-kinetic BGK finite-volume method for three-dimensional transonic flow, *J. Comput. Phys.* **220**, 856 (2007).
- [8] L. M. Yang, C. Shu, and J. Wu, A three-dimensional explicit sphere function-based gas-kinetic flux solver for simulation of inviscid compressible flows, *J. Comput. Phys.* **295**, 322 (2015).
- [9] K. Xu, *Direct Modeling for Computational Fluid Dynamics: Construction and Application of Unified Gas-Kinetic Schemes* (World Scientific, Singapore, 2014).
- [10] C. K. Aidun and J. R. Clausen, Lattice-Boltzmann method for complex flows, *Annu. Rev. Fluid Mech.* **42**, 439 (2010).
- [11] Y. H. Qian, D. d'Humières, and P. Lallemand, Lattice BGK models for Navier-Stokes equation, *Europhys. Lett.* **17**, 479 (1992).
- [12] X. Shan, Simulation of Rayleigh-Bénard convection using a lattice Boltzmann method, *Phys. Rev. E* **55**, 2780 (1997).
- [13] 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).
- [14] Y. Peng, C. Shu, and Y. T. Chew, Simplified thermal lattice Boltzmann model for incompressible thermal flows, *Phys. Rev. E* **68**, 026701 (2003).
- [15] Z. Guo, B. Shi, and C. Zheng, A coupled lattice BGK model for the Boussinesq equations, *Int. J. Numer. Methods Fluids* **39**, 325 (2002).
- [16] Z. Guo, C. Zheng, B. Shi, and T. S. Zhao, Thermal lattice Boltzmann equation for low Mach number flows: Decoupling model, *Phys. Rev. E* **75**, 036704 (2007).
- [17] H. N. Dixit and V. Babu, Simulation of high Rayleigh number natural convection in a square cavity using the lattice Boltzmann method, *Int. J. Heat Mass Transfer* **49**, 727 (2006).
- [18] N. I. Prasianakis and I. V. Karlin, Lattice Boltzmann method for thermal flow simulation on standard lattices, *Phys. Rev. E* **76**, 016702 (2007).
- [19] A. E. F. Monfared, A. Sarrafi, S. Jafari, and M. Schaffie, Thermal flux simulations by lattice Boltzmann method; investigation of high Richardson number cross flows over tandem square cylinders, *Int. J. Heat Mass Transfer* **86**, 563 (2015).
- [20] X. He, L. S. Luo, and M. Dembo, Some progress in lattice Boltzmann method. Part I. Nonuniform mesh grids, *J. Comput. Phys.* **129**, 357 (1996).
- [21] C. Shu, X. D. Niu, and Y. T. Chew, Taylor-series expansion and least-squares-based lattice Boltzmann method: Two-dimensional formulation and its applications, *Phys. Rev. E* **65**, 036708 (2002).
- [22] A. Bardow, I. V. Karlin, and A. A. Gusev, Multispeed models in off-lattice Boltzmann simulations, *Phys. Rev. E* **77**, 025701 (2008).
- [23] 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).
- [24] Z. Guo, C. Zheng, and B. Shi, An extrapolation method for boundary conditions in lattice Boltzmann method, *Phys. Fluids* **14**, 2007 (2002).
- [25] Y. Wang, C. Shu, and C. J. Teo, Thermal lattice Boltzmann flux solver and its application for simulation of incompressible thermal flows, *Comput. Fluids* **94**, 98 (2014).
- [26] Y. Wang, C. Shu, C. J. Teo, J. Wu, and L. M. Yang, Three-dimensional lattice Boltzmann flux solver and its applications to incompressible isothermal and thermal flows, *Commun. Comput. Phys.* **18**, 593 (2015).
- [27] S. Y. Chou and D. Baganoff, Kinetic flux-vector splitting for the Navier-Stokes equations, *J. Comput. Phys.* **130**, 217 (1997).
- [28] K. Xu, A gas-kinetic BGK scheme for the Navier-Stokes equations and its connection with artificial dissipation and Godunov method, *J. Comput. Phys.* **171**, 289 (2001).
- [29] H. Z. Tang, Gas-kinetic schemes for compressible flow of real gases, *Comput. Math. Appl.* **41**, 723 (2001).
- [30] G. Kumar, S. S. Girimaji, and J. Kerimo, WENO-enhanced gas-kinetic scheme for direct simulations of compressible transition and turbulence, *J. Comput. Phys.* **234**, 499 (2013).
- [31] L. Tang, Progress in gas-kinetic upwind schemes for the solution of Euler/Navier-Stokes equations - I: Overview, *Comput. Fluids* **56**, 39 (2012).

- [32] M. Su, K. Xu, and M. S. Ghidaoui, Low-speed flow simulation by the gas-kinetic scheme, *J. Comput. Phys.* **150**, 17 (1999)
- [33] K. Xu and X. He, Lattice Boltzmann method and gas-kinetic BGK scheme in the low-Mach number viscous flow simulations, *J. Comput. Phys.* **190**, 100 (2003).
- [34] Z. Guo, H. Liu, L. S. Luo, and K. Xu, A comparative study of the LBE and GKS methods for 2D near incompressible laminar flows, *J. Comput. Phys.* **227**, 4955 (2008).
- [35] K. Xu and S. H. Lui, Rayleigh-Bénard simulation using the gas-kinetic Bhatnagar-Gross-Krook scheme in the incompressible limit, *Phys. Rev. E* **60**, 464 (1999).
- [36] S. Chen, C. Jin, C. B. Lee, and Q. Cai, Gas-kinetic scheme with discontinuous derivative for low speed flow computation, *J. Comput. Phys.* **230**, 2045 (2011).
- [37] R. Yuan, C. Zhong, and H. Zhang, An immersed-boundary method based on the gas kinetic BGK scheme for incompressible viscous flow, *J. Comput. Phys.* **296**, 184 (2015).
- [38] L. M. Yang, C. Shu, J. Wu, N. Zhao, and Z. L. Lu, Circular function-based gas-kinetic scheme for simulation of inviscid compressible flows, *J. Comput. Phys.* **255**, 540 (2013).
- [39] L. M. Yang, C. Shu, and J. Wu, A simple distribution function-based gas-kinetic scheme for simulation of viscous incompressible and compressible flows, *J. Comput. Phys.* **274**, 611 (2014).
- [40] L. M. Yang, C. Shu, and Y. Wang, Development of a discrete gas-kinetic scheme for simulation of two-dimensional viscous incompressible and compressible flows, *Phys. Rev. E* **93**, 033311 (2016).
- [41] L. M. Yang, C. Shu, W. M. Yang, and Y. Wang, A simplified circular function-based gas kinetic scheme for simulation of incompressible flows, *Int. J. Numer. Methods Fluids* **85**, 583 (2017).
- [42] C. Shu, Y. Wang, C. J. Teo, and J. Wu, Development of lattice Boltzmann flux solver for simulation of incompressible flows, *Adv. Appl. Math. Mech.* **6**, 436 (2014).
- [43] H. B. Huang, X. Y. Lu, and M. C. Sukop, Numerical study of lattice Boltzmann methods for a convection-diffusion equation coupled with Navier-Stokes equations, *J. Phys. A: Math. Theor.* **44**, 055001 (2011).
- [44] Y. Wang, L. M. Yang, and C. Shu, From lattice Boltzmann method to lattice Boltzmann flux solver, *Entropy* **17**, 7713 (2015).
- [45] L. M. Yang, C. Shu, W. M. Yang, Y. Wang, and C. B. Lee, An implicit simplified sphere function-based gas kinetic scheme for simulation of 3D incompressible isothermal flows, *Comput. Fluids* **160**, 204 (2018).
- [46] W. H. Reid and D. L. Harris, Some further results on the Bénard problem, *Phys. Fluids* **1**, 102 (1958).
- [47] R. M. Clever and F. H. Busse, Transition to time-dependent convection, *J. Fluid Mech.* **65**, 625 (1974).
- [48] C. Shu and H. Xue, Comparison of two approaches for implementing stream function boundary condition in DQ simulation of natural in a square cavity, *Int. J. Heat Fluid Flow* **19**, 59 (1998).
- [49] D. Contrino, P. Lallemand, P. Asinari, and L. S. Luo, Lattice-Boltzmann simulations of the thermally driven 2D square cavity at high Rayleigh numbers, *J. Comput. Phys.* **275**, 257 (2014).
- [50] P. Le Quéré, Accurate solutions to the square thermally driven cavity at high Rayleigh number, *Comput. Fluids* **20**, 29 (1991).
- [51] C. Shu, Application of differential quadrature method to simulate natural convection in a concentric annulus, *Int. J. Numer. Methods Fluids* **30**, 977 (1999).
- [52] T. H. Kuehn and R. J. Goldstein, An experimental and theoretical study of natural convection in the annulus between horizontal concentric cylinders, *J. Fluid Mech.* **74**, 695 (1976).
- [53] C. B. Lee, H. W. Peng, H. J. Yuan, J. Z. Wu, M. D. Zhou, and F. Hussain, Experimental studies of surface waves inside a cylindrical container, *J. Fluid Mech.* **677**, 39 (2011).
- [54] H. M. Badr, Laminar combined convection from a horizontal cylinder-parallel and contra flow regimes, *Int. J. Heat Mass Transfer* **27**, 15 (1984).