Analytical and numerical studies of the boundary slip in the immersed boundary-thermal lattice Boltzmann method

Seta, Takashi

Hayashi, Kosuke

Tomiyama, Akio

(Citation)
International Journal for Numerical Methods in Fluids,86(7):454-490

(Issue Date)
2018-03-10

(Resource Type)
journal article

(Version)
Version of Record

Received: 3 August 2016 | Revised: 26 July 2017 | Accepted: 26 August 2017

DOI: 10.1002/fld.4462

RESEARCH ARTICLE

WILEY

Analytical and numerical studies of the boundary slip in the immersed boundary-thermal lattice Boltzmann method

Takashi Seta[1] | Kosuke Hayashi[2] | Akio Tomiyama[2]

[1]Graduate School of Science and Engineering for Research, University of Toyama, Toyama, 930-8555, Japan

[2]Graduate School of Engineering, Kobe University, Hyogo, 657-8501, Japan

Correspondence
Takeshi Seta, Graduate School of Science and Engineering for Research, University of Toyama, Toyama 930-8555, Japan.
Email: seta@eng.u-toyama.ac.jp

Funding information
JSPS KAKENHI, Grant/Award Number: JP16K06070

Summary

We analytically and numerically investigate the boundary slip, including the velocity slip and the temperature jump, in immersed boundary-thermal lattice Boltzmann methods (IB-TLBMs) with the two-relaxation-time collision operator. We derive the theoretical equation for the relaxation parameters considering the effect of the advection velocity on the temperature jump of the IB-TLBMs. The analytical and numerical solutions demonstrate that the proposed iterative correction methods without the computational cost of the sparse matrix solver reduce the boundary slip and boundary-value deviation as effectively as the implicit correction method for any relaxation time. Because the commonly used multi-direct forcing method does not consider the contributions of the body force to the momentum flux, it cannot completely eliminate the boundary slip because of the numerical instability for a long relaxation time. Both types of proposed iterative correction methods are more numerically stable than the implicit correction method. In simulations of flow past a circular cylinder and of natural convection, the present iterative correction methods yield adequate results without the errors of the velocity slip, the temperature jump, and the boundary-value deviation for any relaxation time parameters and for any number of Lagrangian points per length. The combination of the present methods and the two-relaxation-time collision operator is suitable for simulating fluid flow with thermal convection in the multiblock method in which the relaxation time increases in inverse proportion to the grid size.

KEYWORDS

double-population approach, immersed boundary method, lattice Boltzmann method, multiblock approach, temperature jump, velocity slip

1 | INTRODUCTION

Because the immersed boundary method (IBM)[1] and the lattice Boltzmann method (LBM)[2] use the uniform Cartesian mesh to calculate fluid flow, a variety of immersed boundary-lattice Boltzmann methods (IB-LBMs) have been proposed

to calculate the dynamics of solid particles in an incompressible viscous fluid and the related thermal phenomena.[3-11] Feng and Michaelides combined the LBM and a penalty method that used the restoring force from the deformable particle with an a priori–determined high-stiffness spring constant.[3] Feng and Michaelides proposed the IB-LBM based on the direct forcing method that obtained the force by comparing the fluid velocity to the desired velocity at the boundary without requiring a user-defined stiffness parameter.[4] Niu et al introduced the momentum exchange approach proposed by Ladd[12] into the IB-LBM, in which the force density at the boundary point is calculated by the momentum difference between the density distribution function and its counterpart with the opposite discrete velocity.[5] Although these explicit IB-LBMs succeeded in a simulation of the sedimentation of particles in an enclosure,[3-5] they provoked spurious fluid mass exchange between inside and outside the solid due to the inexactness of the nonslip boundary condition. To exactly satisfy the nonslip boundary condition, Wu and Shu proposed an implicit velocity correction-based IB-LBM that gave the force density and velocity correction by solving the system of equations using an inverse of matrix.[6] Because the nonslip boundary condition is accurately enforced, the implicit correction method successfully prevents the penetration of a fluid into a solid surface observed in conventional IB-LBMs. An iterative correction approach was also proposed to satisfy the nonslip boundary condition.[7-9] Using the multi-direct forcing scheme proposed by Wang et al,[13] Suzuki and Inamuro reduced the discrepancy between the computed velocity and the desired velocity in the iterative procedure.[7] Kang and Hassan improved the nonslip boundary conditions by iterative correction of both the velocity and the force density based on Guo's external forcing term.[8,14] Zhang et al proposed an iterative force-correction method based on Cheng's split forcing lattice Boltzmann equation (LBE), whose forcing term affected the distribution function.[9,15] By using the iterative procedure and the split forcing term, Cheng et al improved the accuracy and the numerical stability of the penalty method.[11]

Although some methods succeeded in accurately enforcing the nonslip boundary condition, the numerical errors in the velocity and the velocity gradient at the boundary, which are referenced collectively as boundary slip, increase with relaxation time in numerical simulation by the IB-LBM.[16] The use of the multirelaxation time (MRT) collision operator[17] solved the problem of boundary slip occurring at high relaxation times in the IB-LBM.[18] The two-relaxation-time (TRT) collision operator[19] reduced the boundary slip velocity as effectively as the MRT collision operator.[20] Another boundary error, ie, the disagreement between the boundary velocity on the Eulerian node and the desired velocity, also increases with increasing relaxation time in the IB-LBM calculation.[16,18] By solving systems of algebraic equations, the implicit velocity-correction method with the TRT collision operator succeeded in removing all boundary errors that depend on the relaxation time.[21] Although Zhang et al numerically demonstrated that the IB-LBM using the iterative procedure with the MRT collision operator also reduced the boundary errors, they did not completely eliminate the velocity slip in their numerical calculations.[9] For IB-LBMs using iterative schemes, it is necessary to obtain an optimal analytical solution of the relaxation parameters to reduce the velocity slip.

An internal energy distribution function was introduced to the LBM to compute the temperature evolution for incompressible flow.[22-24] The double-population thermal lattice Boltzmann method (TLBM) provided better numerical stability and successfully simulated natural convection inside a laterally heated square enclosure for a wide range of Rayleigh numbers.[25] Using the TLBM under the reflective bounce-back and diffuse-scattering boundary conditions, Karimipour et al calculated the slip velocity and temperature jump in the numerical simulation of mixed convection heat transfer in a microchannel at high Knudsen numbers.[26] The TLBM computed the nanofluid mixed convection in a lid-driven cavity with the same precision as conventional finite volume approaches.[27-29] Karimipour et al investigated the effects of nanoparticle volume fractions and the slip coefficient on the velocity slip, temperature jump, and Nusselt number in a numerical simulation of a nanofluid laminar forced convection inside a microchannel using the TLBM.[30,31] Jeong et al applied the IBM to the double-population TLBM to simulate natural convection in a square cavity with curve boundaries.[10] This combination of methods is known as the immersed boundary-thermal lattice Boltzmann method (IB-TLBM). A numerical error in temperature at the boundary was observed in the IB-TLBM based on the single relaxation time (SRT) collision operator.[32] The numerical error arising at long relaxation times remains to be resolved for the thermal models.

In this paper, we propose an IB-TLBM based on the iterative approach and theoretically and numerically investigate the effectiveness of combining the iterative correction method and the TRT collision operator on the reduction of the boundary slip. The paper is organized as follows. Section 2 briefly describes the IB-TLBMs, including the direct forcing method, the multi-direct forcing method, the implicit correction method, and 2 types of the iterative correction methods. In Section 3, we derive the theoretical relation between the temperature jump and the relaxation time, and we analytically and numerically examine the effect of the advection velocity on the boundary error in the IB-TLBM. We examine the accuracy and validity of the current methods through two-dimensional simulations of symmetric shear flows, the porous

plates problem, cylindrical Couette flows, flows past a circular cylinder at high viscosity, natural convection, and the multiblock approach. Final conclusions are given in Section 4.

2 | IMMERSED BOUNDARY-THERMAL LATTICE BOLTZMANN METHOD

As shown in Figure 1A, the IBM sets the Lagrangian points $\boldsymbol{x}_b$ in uniform Cartesian coordinates to represent the boundary of an arbitrarily shaped body in a viscous incompressible fluid. The force density $\boldsymbol{G}$ and the heat source density Q are used to calculate the interaction between the boundary node $\boldsymbol{x}_b$ and the Eulerian grid $\boldsymbol{x}_f$. Among the numerous effective methods for computing the force $\boldsymbol{G}$ and the heat source Q in the IB-TLBM, we herein consider 3 typical methods: the direct forcing method,[4] the multi-direct forcing method,[7] and the implicit correction method.[32] From the perspectives of simplicity and computational efficiency, we propose 2 types of iterative correction methods to exploit the concept of the implicit diffuse direct forcing approach[8] and the iterative force-correction method,[9] which set the boundary values equal to the desired ones. We call the proposed IB-TLBM an iterative velocity and temperature-correction method, as well as an iterative force and heat-correction method. We apply the TRT collision operator to these 5 methods. In this Section, we explain these 5 methods in simple terms.

2.1 | Direct forcing method with 2 relaxation times

The direct forcing method with the TRT collision operator for the incompressible Navier-Stokes equations uses the following kinetic equations for the distribution function f_k:

$$\tilde{f}_k(\boldsymbol{x},t) = f_k(\boldsymbol{x},t) + \Omega_k^v(\boldsymbol{x},t) + \frac{3\delta_t\rho\omega_k\boldsymbol{e}_k\cdot\boldsymbol{G}(\boldsymbol{x},t)}{c^2}, \tag{1a}$$

$$f_k(\boldsymbol{x}+\boldsymbol{e}_k\delta_t, t+\delta_t) = \tilde{f}_k(\boldsymbol{x},t), \tag{1b}$$

where c is the lattice velocity magnitude, $\boldsymbol{G}$ is a body force, δ_t is the time step, and f_k^{eq} is the equilibrium distribution function. Equations 1a and 1b represent the collision and streaming processes, respectively. The TRT collision operator is applied as follows:

$$\Omega_k^v(\boldsymbol{x},t) = -\frac{f_k^+(\boldsymbol{x},t) - f_k^{+eq}(\boldsymbol{x},t)}{\tau_+^v} - \frac{f_k^-(\boldsymbol{x},t) - f_k^{-eq}(\boldsymbol{x},t)}{\tau_-^v}, \tag{2}$$

where τ_+^v and τ_-^v are the relaxation times for the symmetric (even) and antisymmetric (odd) distribution functions, f_k^+ and f_k^-, respectively.

The TLBM for the temperature equation uses the following kinetic equations for the distribution function g_k:

$$\tilde{g}_k(\boldsymbol{x},t) = g_k(\boldsymbol{x},t) + \Omega_k^c(\boldsymbol{x},t) + \delta_t\omega_k\rho Q(\boldsymbol{x},t), \tag{3a}$$

$$g_k(\boldsymbol{x}+\boldsymbol{e}_k\delta_t, t+\delta_t) = \tilde{g}_k(\boldsymbol{x},t), \tag{3b}$$

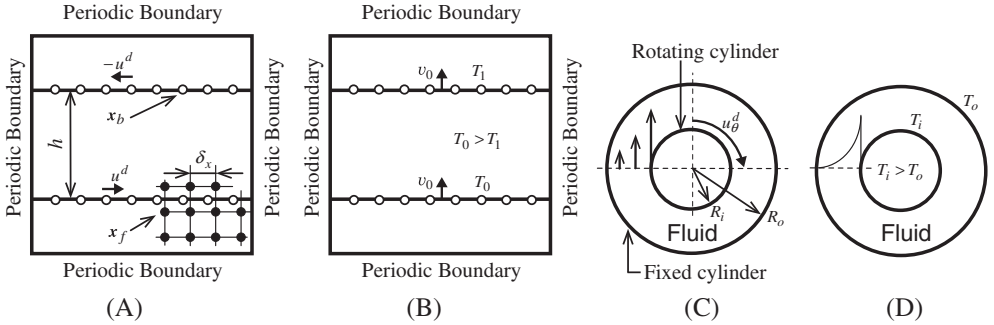


FIGURE 1 Schematic diagrams of 4 cases: A, symmetric shear flows; B, heat transfer between 2 porous plates; C, hydrodynamic boundary condition in cylindrical Couette flow; D, thermal boundary condition in cylindrical Couette flow

where

$$\Omega_k^c(\boldsymbol{x},t) = -\frac{g_k^+(\boldsymbol{x},t) - g_k^{+eq}(\boldsymbol{x},t)}{\tau_+^c} - \frac{g_k^-(\boldsymbol{x},t) - g_k^{-eq}(\boldsymbol{x},t)}{\tau_-^c}. \tag{4}$$

Here, g_k^+ and g_k^- are the even and odd components of g_k, respectively, and g_k^{eq} is the equilibrium distribution function. τ_+^c and τ_-^c are the relaxation times for g_k^+ and g_k^-, respectively, and Q is a heat source.

The fluid density ρ, velocity $\boldsymbol{u}$, and temperature T are conventionally written in terms of the distribution function as follows:

$$\rho = \sum_k f_k, \boldsymbol{u} = \frac{1}{\rho}\sum_k f_k \boldsymbol{e}_k, T = \frac{1}{\rho}\sum_k g_k. \tag{5}$$

The pressure is calculated by $p = c_s^2\rho$. The speed of sound c_s is $c/\sqrt{3}$. The D2Q9 model is used in this study. The discrete velocities are defined as

$$\boldsymbol{e}_k = \begin{cases} (0,0), & k = 0, \\ c(\cos(\frac{\pi(k-1)}{2}), \sin(\frac{\pi(k-1)}{2})), & k = 1-4, \\ \sqrt{2}c(\cos(\frac{\pi(2k-9)}{4}), \sin(\frac{\pi(2k-9)}{4})), & k = 5-8. \end{cases} \tag{6}$$

In the simplified thermal energy distribution model for the D2Q9 model, the equilibrium distribution function is given by Shi et al[23]:

$$f_k^{eq} = \omega_k\rho\left(1 + \frac{3\boldsymbol{e}_k\cdot\boldsymbol{u}}{c^2} + \frac{9(\boldsymbol{e}_k\cdot\boldsymbol{u})^2}{2c^4} - \frac{3\boldsymbol{u}\cdot\boldsymbol{u}}{2c^2}\right), \tag{7}$$

$$g_k^{eq} = \omega_k\rho T\left(1 + \frac{3\boldsymbol{e}_k\cdot\boldsymbol{u}}{c^2} + \frac{9(\boldsymbol{e}_k\cdot\boldsymbol{u})^2}{2c^4} - \frac{3\boldsymbol{u}\cdot\boldsymbol{u}}{2c^2}\right), \tag{8}$$

where ω_k are the weight coefficients, ie, $\omega_0 = 4/9$, $\omega_{1-4} = 1/9$, and $\omega_{5-8} = 1/36$.

In Seta,[32] we used the equilibrium distribution function proposed by Peng et al.[24] They defined the equilibrium distribution function for the resting particle as $g_0^{eq} = -\frac{2\rho T}{3}\frac{\boldsymbol{u}\cdot\boldsymbol{u}}{c^2}$, which is negative for a nonzero absolute value of the fluid velocity. Because the nonnegativity of all equilibrium distribution functions gives a sufficient condition for numerical stability,[19] we use the equilibrium distribution function given by Equation 8 in this study. The velocity-squared terms in Equation 8 are necessary to remove the full second-order tensor of the numerical diffusion.[19]

In the TRT collision model, the distribution functions f_k and g_k and their counterparts with the opposite velocity $f_{\bar{k}}$ and $g_{\bar{k}}$ generate the symmetric (even) f_k^+, g_k^+ and antisymmetric (odd) components f_k^- and g_k^-[19]:

$$f_k^{\pm} = \frac{f_k \pm f_{\bar{k}}}{2}, g_k^{\pm} = \frac{g_k \pm g_{\bar{k}}}{2}, f_0^+ = f_0, g_0^+ = g_0, f_0^- = 0, g_0^- = 0. \tag{9}$$

The equilibrium distribution functions f_k^{eq} and g_k^{eq} also take the symmetric and antisymmetric components as follows:

$$f_k^{\pm eq} = \frac{f_k^{eq} \pm f_{\bar{k}}^{eq}}{2}, g_k^{\pm eq} = \frac{g_k^{eq} \pm g_{\bar{k}}^{eq}}{2}, f_0^{+eq} = f_0^{eq}, g_0^{+eq} = g_0^{eq}, f_0^{-eq} = 0, g_0^{-eq} = 0. \tag{10}$$

Applying the Chapman-Enskog expansion to Equations 1a, 1b, 3a, 3b, 7, and 8 yields the following macroscopic equations:

$$\nabla\cdot\boldsymbol{u} = 0, \tag{11}$$

$$\frac{\partial\boldsymbol{u}}{\partial t} + \nabla\cdot(\boldsymbol{u}\boldsymbol{u}) = -\frac{1}{\rho}\nabla p + \nu\nabla\cdot[\nabla\boldsymbol{u} + (\nabla\boldsymbol{u})^T] + \boldsymbol{G}, \tag{12}$$

$$\frac{\partial T}{\partial t} + \boldsymbol{u}\cdot\nabla T = \chi\nabla^2 T + Q. \tag{13}$$

The kinematic viscosity ν and the thermal diffusivity χ are related with τ_+^v and τ_-^c, respectively, as follows:

$$\nu = \frac{(2\tau_+^v - 1)\delta_x^2}{6\delta_t}, \chi = \frac{(2\tau_-^c - 1)\delta_x^2}{6\delta_t}, \tag{14}$$

where $\delta_i (i = x \text{ or } y)$ is the Eulerian mesh size, which satisfies $\delta_i = c\delta_t$, and the subscript denotes the direction, and where δ_x is the Eulerian mesh size, which satisfies $\delta_x = c\delta_t$.

The τ_-^v and τ_+^c are independent of the physical properties and can be used to reduce the numerical errors. To reduce the boundary slip for the direct forcing method, we use the following equations[21]:

$$\tau_-^v = \frac{3\tau_+^v + 3}{4\tau_+^v - 2}, \tau_+^c = \frac{3\tau_-^c + 3}{4\tau_-^c - 2}. \tag{15}$$

The direct forcing method evaluates $\boldsymbol{G}(\boldsymbol{x}_b, t)$ and $Q(\boldsymbol{x}_b, t)$ at the Lagrangian nodes $\boldsymbol{x}_b$ by

$$\boldsymbol{G}(\boldsymbol{x}_b, t) = \frac{\boldsymbol{u}^d(\boldsymbol{x}_b, t) - \boldsymbol{u}(\boldsymbol{x}_b, t)}{\delta_t}, Q(\boldsymbol{x}_b, t) = \frac{T^d(\boldsymbol{x}_b, t) - T(\boldsymbol{x}_b, t)}{\delta_t}. \tag{16}$$

where $\boldsymbol{u}^d$ is the desired wall velocity at the boundary nodes, and T^d is the desired temperature at the boundary nodes. The velocity $\boldsymbol{u}(\boldsymbol{x}_b)$ and temperature $T(\boldsymbol{x}_b)$ are evaluated through the interpolation of the velocities and temperatures at the neighboring fluid nodes $\boldsymbol{x}_f$ using a discrete delta function $D(\boldsymbol{x})$ as follows:

$$\boldsymbol{u}(\boldsymbol{x}_b, t) = \sum_f \boldsymbol{u}(\boldsymbol{x}_f, t) D(\boldsymbol{x}_f - \boldsymbol{x}_b)\delta_x^2, T(\boldsymbol{x}_b, t) = \sum_f T(\boldsymbol{x}_f, t) D(\boldsymbol{x}_f - \boldsymbol{x}_b)\delta_x^2. \tag{17}$$

In a two-dimensional coordinate system $\boldsymbol{x} = (x, y)$, $D(\boldsymbol{x})$ is expressed as

$$D(\boldsymbol{x}_f - \boldsymbol{x}_b) = \delta(x_f - x_b)\delta(y_f - y_b), \tag{18}$$

where

$$\delta(r) = \begin{cases} \frac{1}{8\delta_x}\left(3 - 2\frac{|r|}{\delta_x} + \sqrt{1 + 4\frac{|r|}{\delta_x} - 4\frac{|r|^2}{\delta_x^2}}\right), & |r| \le \delta_x, \\ \frac{1}{8\delta_x}\left(5 - 2\frac{|r|}{\delta_x} - \sqrt{-7 + 12\frac{|r|}{\delta_x} - 4\frac{|r|^2}{\delta_x^2}}\right), & \delta_x < |r| \le 2\delta_x, \\ 0, & 2\delta_x \le |r|. \end{cases} \tag{19}$$

To calculate the force and heat from a solid structure to a fluid, we must extend $\boldsymbol{G}(\boldsymbol{x}_b)$ and $Q(\boldsymbol{x}_b)$ at the solid nodes to the fluid nodes. In the Eulerian coordinate system, $\boldsymbol{G}(\boldsymbol{x}_f)$ and $Q(\boldsymbol{x}_f)$ are evaluated by interpolating $\boldsymbol{G}(\boldsymbol{x}_b)$ and $Q(\boldsymbol{x}_b)$ as

$$\boldsymbol{G}(\boldsymbol{x}_f, t) = \sum_b^N \boldsymbol{G}(\boldsymbol{x}_b, t) D(\boldsymbol{x}_f - \boldsymbol{x}_b)\Delta s, Q(\boldsymbol{x}_f, t) = \sum_b^N Q(\boldsymbol{x}_b, t) D(\boldsymbol{x}_f - \boldsymbol{x}_b)\Delta s, \tag{20}$$

where N is the total number of Lagrangian points, and Δs is the discrete length for each Lagrangian point.

The solution procedure of the TRT direct forcing method is as follows:

1. Give the initial density $\rho(\boldsymbol{x}_f)$, velocity $\boldsymbol{u}(\boldsymbol{x}_f)$, temperature $T(\boldsymbol{x}_f)$, and the desired values $\boldsymbol{u}^d(\boldsymbol{x}_b)$ and $T^d(\boldsymbol{x}_b)$. Initially, $\boldsymbol{G}(\boldsymbol{x}_f)$ and $Q(\boldsymbol{x}_f)$ are zero.
2. Compute $f_k^{eq}(\boldsymbol{x}_f, t)$ and $g_k^{eq}(\boldsymbol{x}_f, t)$ corresponding to the initial values using Equations 7 and 8, respectively. Give the initial distribution functions as $f_k(\boldsymbol{x}_f, t) = f_k^{eq}(\boldsymbol{x}_f, t)$ and $g_k(\boldsymbol{x}_f, t) = g_k^{eq}(\boldsymbol{x}_f, t)$.
3. Obtain $\boldsymbol{u}(\boldsymbol{x}_b, t)$ and $T(\boldsymbol{x}_b, t)$ from $\boldsymbol{u}(\boldsymbol{x}_f, t)$ and $T(\boldsymbol{x}_f, t)$ using Equation 17.
4. Compute $\boldsymbol{G}(\boldsymbol{x}_b, t)$ and $Q(\boldsymbol{x}_b, t)$ using Equation 16, and distribute $\boldsymbol{G}(\boldsymbol{x}_b, t)$ and $Q(\boldsymbol{x}_b, t)$ to the Eulerian nodes using Equation 20.
5. Compute $f_k^{eq}(\boldsymbol{x}_f, t)$ and $g_k^{eq}(\boldsymbol{x}_f, t)$ using Equations 7 and 8.
6. Compute $f_k^{\pm}(\boldsymbol{x}_f, t)$, $g_k^{\pm}(\boldsymbol{x}_f, t)$, $f_k^{\pm eq}(\boldsymbol{x}_f, t)$, and $g_k^{\pm eq}(\boldsymbol{x}_f, t)$ using Equations 9 and 10.
7. Obtain $\tilde{f}_k(\boldsymbol{x}_f, t)$ and $\tilde{g}_k(\boldsymbol{x}_f, t)$ using Equations 1a and 3a.
8. $\tilde{f}_k(\boldsymbol{x}_f, t)$ and $\tilde{g}_k(\boldsymbol{x}_f, t)$ stream following Equations 1b and 3b.
9. Compute $\rho(\boldsymbol{x}_f, t + \delta_t)$, $\boldsymbol{u}(\boldsymbol{x}_f, t + \delta_t)$, and $T(\boldsymbol{x}_f, t + \delta_t)$ using Equation 5, and return to step 3.

2.2 | Multi-direct forcing method with 2 relaxation times

The velocity and temperature at the Lagrangian nodes may not agree with the desired values during the processes of interpolation and extrapolation given by Equations 16 through 20. Wang et al proposed the multi-direct forcing method, which causes the velocity and temperature at the Lagrangian nodes to satisfy the Dirichlet boundary condition using

iterative calculations. For the multi-direct forcing method in the framework of the LBM,[7] the distribution functions f_k and g_k are updated using the forcing and heat source terms, respectively, after the TRT collision and streaming steps.

$$\tilde{f}_k(\boldsymbol{x}+\boldsymbol{e}_k\delta_t, t+\delta_t) = f_k(\boldsymbol{x}, t) + \Omega_k^v(\boldsymbol{x}, t), \tag{21a}$$

$$f_k(\boldsymbol{x}, t+\delta_t) = \tilde{f}_k(\boldsymbol{x}, t+\delta_t) + \frac{3\delta_t\rho\omega_k\boldsymbol{e}_k\cdot\boldsymbol{G}(\boldsymbol{x}, t+\delta_t)}{c^2}, \tag{21b}$$

$$\tilde{g}_k(\boldsymbol{x}+\boldsymbol{e}_k\delta_t, t+\delta_t) = g_k(\boldsymbol{x}, t) + \Omega_k^c(\boldsymbol{x}, t), \tag{21c}$$

$$g_k(\boldsymbol{x}, t+\delta_t) = \tilde{g}_k(\boldsymbol{x}, t+\delta_t) + \delta_t\omega_k\rho Q(\boldsymbol{x}, t+\delta_t). \tag{21d}$$

For the multi-direct forcing method, we use the following relations between τ_- and τ_+:

$$\tau_-^v = \frac{\tau_+^v + 4}{4\tau_+^v - 2}, \tau_+^c = \frac{\tau_-^c + 4}{4\tau_-^c - 2}. \tag{22}$$

Equation 22 reduces the boundary slip (see Appendix A). The von Neumann linearized stability analysis of the LB scheme shows that numerical stability is guaranteed if $\tau_\pm^v > 0.5$ and $\tau_\pm^c > 0.5$.[33] Because τ_-^v is less than 0.5 for $\tau_+^v > 5$, Equation 22 is used under the condition $\tau_+^v < 5$. For a similar reason, in Equation 22, τ_-^c should be less than 5.

The multi-direct forcing method iteratively calculates the force density as follows[13]:

$$\tilde{\boldsymbol{u}}^{n-1}(\boldsymbol{x}_b, t) = \sum_f \tilde{\boldsymbol{u}}^{n-1}(\boldsymbol{x}_f, t)D(\boldsymbol{x}_f - \boldsymbol{x}_b)\delta_x^2, \tag{23a}$$

$$\tilde{\boldsymbol{G}}^{n-1}(\boldsymbol{x}_b, t) = \frac{\boldsymbol{u}^d(\boldsymbol{x}_b, t) - \tilde{\boldsymbol{u}}^{n-1}(\boldsymbol{x}_b, t)}{\delta_t}, \tag{23b}$$

$$\tilde{\boldsymbol{G}}^{n-1}(\boldsymbol{x}_f, t) = \sum_b^N \tilde{\boldsymbol{G}}^{n-1}(\boldsymbol{x}_b, t)D(\boldsymbol{x}_f - \boldsymbol{x}_b)\Delta s, \tag{23c}$$

$$\tilde{\boldsymbol{u}}^{n}(\boldsymbol{x}_f, t) = \tilde{\boldsymbol{u}}^{n-1}(\boldsymbol{x}_f, t) + \delta_t\tilde{\boldsymbol{G}}^{n-1}(\boldsymbol{x}_f, t), \tag{23d}$$

$$\boldsymbol{G}(\boldsymbol{x}_b, t) = \sum_n^{N_m} \tilde{\boldsymbol{G}}^n(\boldsymbol{x}_b, t), \quad \boldsymbol{G}(\boldsymbol{x}_f, t) = \sum_n^{N_m} \tilde{\boldsymbol{G}}^n(\boldsymbol{x}_f, t), \tag{23e}$$

where $\tilde{\boldsymbol{u}}$ is a temporal velocity, $\tilde{\boldsymbol{G}}$ is a temporal force, and the subscript n is the number of iterations in the multi-direct forcing method. Based on the calculation given by Equations 23a through 23d, $\|\tilde{\boldsymbol{u}}^n - \boldsymbol{u}^d\|$ is expected to be less than $\|\tilde{\boldsymbol{u}}^{n-1} - \boldsymbol{u}^d\|$. Through N_m iterations of the procedure, the body force is calculated by Equation 23e.

We execute a similar procedure N_m times per time step δ_t to bring the temperature at the Lagrangian node $\tilde{T}^n$ closer to the desired temperature T^d.

$$\tilde{T}^{n-1}(\boldsymbol{x}_b, t) = \sum_f \tilde{T}^{n-1}(\boldsymbol{x}_f, t)D(\boldsymbol{x}_f - \boldsymbol{x}_b)\delta_x^2, \tag{24a}$$

$$\tilde{Q}^{n-1}(\boldsymbol{x}_b, t) = \frac{T^d(\boldsymbol{x}_b, t) - \tilde{T}^{n-1}(\boldsymbol{x}_b, t)}{\delta_t}, \tag{24b}$$

$$\tilde{Q}^{n-1}(\boldsymbol{x}_f, t) = \sum_b^N \tilde{Q}^{n-1}(\boldsymbol{x}_b, t)D(\boldsymbol{x}_f - \boldsymbol{x}_b)\Delta s, \tag{24c}$$

$$\tilde{T}^{n}(\boldsymbol{x}_f, t) = \tilde{T}^{n-1}(\boldsymbol{x}_f, t) + \delta_t\tilde{Q}^{n-1}(\boldsymbol{x}_f, t), \tag{24d}$$

$$Q(\boldsymbol{x}_b, t) = \sum_n^{N_m} \tilde{Q}^n(\boldsymbol{x}_b, t), Q(\boldsymbol{x}_f, t) = \sum_n^{N_m} \tilde{Q}^n(\boldsymbol{x}_f, t), \tag{24e}$$

where $\tilde{T}$ is a temporal temperature, and $\tilde{Q}$ is a temporal heat source.

The solution procedure of the TRT multi-direct forcing method is as follows:

1. Give the initial density $\rho(\boldsymbol{x}_f)$, velocity $\boldsymbol{u}(\boldsymbol{x}_f)$, and temperature $T(\boldsymbol{x}_f)$.
2. Compute $f_k^{\pm}(\boldsymbol{x}_f, t)$, $g_k^{\pm}(\boldsymbol{x}_f, t)$, $f_k^{\pm eq}(\boldsymbol{x}_f, t)$, and $g_k^{\pm eq}(\boldsymbol{x}_f, t)$ using Equations 7 through 10.
3. $f_k(\boldsymbol{x}_f, t)$ and $g_k(\boldsymbol{x}_f, t)$ stream following Equations 21a and 21c.
4. Compute $\tilde{\rho}(\boldsymbol{x}_f, t)$, $\tilde{\boldsymbol{u}}(\boldsymbol{x}_f, t)$, and $\tilde{T}(\boldsymbol{x}_f, t)$ using Equation 5.
5. Obtain $\tilde{\boldsymbol{u}}^{n-1}(\boldsymbol{x}_b, t)$ and $\tilde{T}^{n-1}(\boldsymbol{x}_b, t)$ from $\boldsymbol{u}^{n-1}(\boldsymbol{x}_f, t)$ and $\tilde{T}^{n-1}(\boldsymbol{x}_f, t)$ using Equations 23a and 24a, respectively.
6. Compute $\tilde{\boldsymbol{G}}^{n-1}(\boldsymbol{x}_b, t)$ and $\tilde{Q}^{n-1}(\boldsymbol{x}_b, t)$ using Equations 23b and 24b.
7. Distribute $\tilde{\boldsymbol{G}}^{n-1}(\boldsymbol{x}_b, t)$ and $\tilde{Q}^{n-1}(\boldsymbol{x}_b, t)$ to Eulerian nodes using Equations 23c and 24c.
8. Compute $\tilde{\boldsymbol{u}}^{n}(\boldsymbol{x}_f, t)$ and $\tilde{T}^{n}(\boldsymbol{x}_f, t)$ using Equations 23d and 24d.
9. If $n \leq N_m$, return to step 5.
10. Obtain the body force $\boldsymbol{G}(\boldsymbol{x}_f, t)$ and heat source $Q(\boldsymbol{x}_f, t)$ using Equations 23e and 24e.
11. Add the effects of the force and the heat source to $\tilde{f}_k(\boldsymbol{x}_f, t + \delta_t)$ and $\tilde{g}_k(\boldsymbol{x}_f, t + \delta_t)$ using Equations 21b and 21d, respectively.
12. Compute $\rho(\boldsymbol{x}_f, t + \delta_t)$, $\boldsymbol{u}(\boldsymbol{x}_f, t + \delta_t)$, and $T(\boldsymbol{x}_f, t + \delta_t)$ using Equation 5, and return to step 2.

As shown in Equations 23a through 23e, the velocity field is updated in an iterative process with the original multi-direct forcing method using the fourth-order compact finite difference scheme.[13] In the IB-LBM proposed by Suzuki and Inamuro, the distribution function f_k and velocity field $\boldsymbol{u}$ are updated only once in steps 11 and 12 after the iterative process between steps 5 and 9.[7] There is a difference in the number of iterations for the correction of the velocity $\boldsymbol{u}$ between the original multi-direct forcing method and the IB-TLBM version. By using the sum of the force $\tilde{\boldsymbol{G}}^n$ for all the iteration times in Equation 23e, the IB-TLBM successfully satisfies the nonslip boundary condition. The maximum number of iterations $N_m = 20$, following Wang et al.[13]

2.3 | Implicit velocity and temperature-correction method with 2 relaxation times

The kinetic equation of f_k in the implicit velocity-correction method with the TRT collision operator uses the extra forcing term proposed by Guo et al[14] as follows[6,32]:

$$\tilde{f}_k(\boldsymbol{x}, t) = f_k(\boldsymbol{x}, t) + \Omega_k^v(\boldsymbol{x}, t) + \delta_t\left(1 - \frac{1}{2\tau_+^v}\right)F_k^+(\boldsymbol{x}, t) + \delta_t\left(1 - \frac{1}{2\tau_-^v}\right)F_k^-(\boldsymbol{x}, t), \tag{25a}$$

$$f_k(\boldsymbol{x} + \boldsymbol{e}_k\delta_t, t + \delta_t) = \tilde{f}_k(\boldsymbol{x}, t). \tag{25b}$$

Guo redefined the fluid velocity and the forcing term in Equation 25a as

$$\rho = \sum_k f_k, \boldsymbol{u} = \frac{1}{\rho}\sum_k f_k\boldsymbol{e}_k + \delta_t\frac{\boldsymbol{G}}{2}, \tag{26}$$

$$F_k = \rho\omega_k\left[\frac{3\boldsymbol{e}_k \cdot \boldsymbol{G}}{c^2} + \frac{9(\boldsymbol{u}\boldsymbol{G} : \boldsymbol{e}_k\boldsymbol{e}_k)}{c^4} - \frac{3\boldsymbol{u} \cdot \boldsymbol{G}}{c^2}\right]. \tag{27}$$

The forcing term is separated into its symmetric and asymmetric parts in the TRT collision operator.

$$F_k^{\pm} = \frac{F_k \pm F_{\bar{k}}}{2}, F_0^+ = F_0, F_0^- = 0. \tag{28}$$

To apply the TRT collision operator to the implicit temperature-correction method,[32] the kinetic equation of g_k uses the heat source term as follows:

$$\tilde{g}_k(\boldsymbol{x}, t) = g_k(\boldsymbol{x}, t) + \Omega_k^c(\boldsymbol{x}, t) + \delta_t\left(1 - \frac{1}{2\tau_+^c}\right)\omega_k\rho Q(\boldsymbol{x}, t), \tag{29a}$$

$$g_k(\boldsymbol{x} + \boldsymbol{e}_k\delta_t, t + \delta_t) = \tilde{g}_k(\boldsymbol{x}, t). \tag{29b}$$

In the implicit temperature-correction method, we obtain the temperature correction at all boundary nodes using the following temperature definition and source term:

$$\rho T = \sum_k g_k + \rho\delta_t\frac{Q}{2}. \tag{30}$$

Ginzburg defines the magic parameter Λ as follows[34]:

$$\Lambda = \left(\tau_+ - \frac{1}{2}\right)\left(\tau_- - \frac{1}{2}\right). \tag{31}$$

The truncation errors of the LBMs with the TRT collision operator depend on Λ. For instance, choosing $\Lambda = \frac{1}{12}$ gives the exact solution for an arbitrarily rotated Poiseuille flow.[34]

The relaxation time τ_-^v is given so as to reduce the velocity slip.[21] The velocity slip and the temperature jump are zero, if the relaxation times τ_-^v and τ_+^c are given by

$$\tau_-^v = \frac{4\tau_+^v + 7}{8\tau_+^v - 4}, \tau_+^c = \frac{4\tau_-^c + 7}{8\tau_-^c - 4}, \left(\Lambda = \frac{9}{8}\right). \tag{32}$$

In Equation 32, τ_-^v and τ_+^c are always larger than 0.5 for all values of τ_+^v and τ_-^c. Equations 26 and 30 can be rewritten as

$$\boldsymbol{u} = \boldsymbol{u}^* + \delta\boldsymbol{u}, T = T^* + \delta T. \tag{33}$$

Here, the intermediate velocity $\boldsymbol{u}^*$ and intermediate temperature T^* are defined by

$$\boldsymbol{u}^* = \frac{1}{\rho}\sum_k f_k \mathbf{e}_k, T^* = \frac{1}{\rho}\sum_k g_k. \tag{34}$$

The implicit velocity and temperature-correction method solves the following simultaneous equations to compute the force and heat at the Lagrangian nodes:

$$A\boldsymbol{X} = \boldsymbol{B}, \tag{35a}$$

$$A_{b'b} = \sum_f D(\boldsymbol{x}_f - \boldsymbol{x}_{b'})\Delta s D(\boldsymbol{x}_f - \boldsymbol{x}_b)\delta_x^2. \tag{35b}$$

To obtain the unknown velocity correction $\delta\boldsymbol{u}$, vectors $\boldsymbol{X}$ and $\boldsymbol{B}$ are given by the following equations:

$$\boldsymbol{X} = (\delta\boldsymbol{u}^1(\boldsymbol{x}_b), \delta\boldsymbol{u}^2(\boldsymbol{x}_b), \cdots \delta\boldsymbol{u}^N(\boldsymbol{x}_b))^T, \tag{36a}$$

$$\boldsymbol{B} = (\Delta\boldsymbol{u}^1(\boldsymbol{x}_{b'}), \Delta\boldsymbol{u}^2(\boldsymbol{x}_{b'}), \cdots \Delta\boldsymbol{u}^N(\boldsymbol{x}_{b'}))^T, \tag{36b}$$

$$\Delta\boldsymbol{u}(\boldsymbol{x}_{b'}) = \boldsymbol{u}^d(\boldsymbol{x}_{b'}) - \sum_f \boldsymbol{u}^*(\boldsymbol{x}_f) D(\boldsymbol{x}_f - \boldsymbol{x}_{b'})\delta_x^2. \tag{36c}$$

The temperature-correction method gives $\boldsymbol{X}$ and $\boldsymbol{B}$ as follows:

$$\boldsymbol{X} = (\delta T^1(\boldsymbol{x}_b), \delta T^2(\boldsymbol{x}_b), \cdots \delta T^N(\boldsymbol{x}_b))^T, \tag{37a}$$

$$\boldsymbol{B} = (\Delta T^1(\boldsymbol{x}_{b'}), \Delta T^2(\boldsymbol{x}_{b'}), \cdots \Delta T^N(\boldsymbol{x}_{b'}))^T, \tag{37b}$$

$$\Delta T(\boldsymbol{x}_{b'}) = T^d(\boldsymbol{x}_{b'}) - \sum_f T^*(\boldsymbol{x}_f) D(\boldsymbol{x}_f - \boldsymbol{x}_{b'})\delta_x^2. \tag{37c}$$

We obtain the force and heat source from the unknown velocity correction and temperature correction at all Lagrangian nodes as follows:

$$\boldsymbol{G}(\boldsymbol{x}_b, t) = \frac{2\delta\boldsymbol{u}(\boldsymbol{x}_b, t)}{\delta_t}, \quad Q(\boldsymbol{x}_b, t) = \frac{2\delta T(\boldsymbol{x}_b, t)}{\delta_t}. \tag{38}$$

This scheme can enforce the Dirichlet boundary condition for both the velocity and the temperature. The solution procedure of the TRT implicit velocity and temperature-correction method can be summarized as follows:

1. Set the initial values, and compute the elements of matrix A and inverse matrix A^{-1} for Equations 35a and 35b.
2. Give the initial intermediate velocity $\boldsymbol{u}^*(\boldsymbol{x}_f)$ and intermediate temperature $T^*(\boldsymbol{x}_f)$.
3. Compute $\boldsymbol{B}$ using Equations 36b, 36c, 37b, and 37c.

4. Solve $A\boldsymbol{X} = \boldsymbol{B}$ using LU decomposition to obtain $\delta\boldsymbol{u}(\boldsymbol{x}_b, t)$ and $\delta T(\boldsymbol{x}_b, t)$.
5. Obtain $\boldsymbol{G}(\boldsymbol{x}_b, t)$ and $Q(\boldsymbol{x}_b, t)$ at all boundary nodes using Equation 38.
6. Use Equation 20 to spread $\boldsymbol{G}(\boldsymbol{x}_b, t)$ and $Q(\boldsymbol{x}_b, t)$ to the Eulerian nodes.
7. Correct the fluid velocity $\boldsymbol{u}(\boldsymbol{x}_f, t)$ and the temperature $T(\boldsymbol{x}_f, t)$ at the Eulerian nodes using Equation 33.
8. Compute the equilibrium distribution functions $f_k^{eq}(\boldsymbol{x}_f, t)$ and $g_k^{eq}(\boldsymbol{x}_f, t)$ using Equations 7 and 8.
9. Compute $f_k^{\pm}(\boldsymbol{x}_f, t)$, $f_k^{\pm eq}(\boldsymbol{x}_f, t)$, $g_k^{\pm}(\boldsymbol{x}_f, t)$, $g_k^{\pm eq}(\boldsymbol{x}_f, t)$, and $F_k^{\pm}(\boldsymbol{x}_f, t)$ using Equations 9, 10, 27, and 28.
10. Solve Equations 25a, 25b, 29a, and 29b to obtain the distribution functions $f_k(\boldsymbol{x}_f, t + \delta_t)$ and $g_k(\boldsymbol{x}_f, t + \delta_t)$.
11. Use Equations 26 and 34 to compute the macroscopic variables $\rho(\boldsymbol{x}_f, t + \delta_t)$, $\boldsymbol{u}^*(\boldsymbol{x}_f, t + \delta_t)$, and $T^*(\boldsymbol{x}_f, t + \delta_t)$.
12. Increase the time from t to $t + \delta_t$, and return to step 3.

2.4 | Iterative velocity and temperature-correction method with 2 relaxation times

Although the implicit velocity-correction method succeeds in preventing fluid penetration into the solid body, the size of matrix A in Equation 35a is proportional to the square of the number of Lagrangian points, N^2, and the memory consumption and computing time both increase quadratically with increasing N. Kang and Hassan proposed the implicit diffuse direct forcing method with the SRT collision operator based on the lattice Boltzmann equation as shown in Equations 25a through 27, which considers the discrete lattice effect and the contributions of the body force to the momentum flux.[8,14] The multi-direct forcing method uses the conventional definition of momentum shown in Equation 5, which does not adequately introduce a body force in the Navier-Stokes equations. We propose an iterative velocity and temperature-correction method that uses the implicit diffuse direct forcing approach in the thermal lattice Boltzmann model. While the multi-direct forcing method uses the temporal values $\tilde{\boldsymbol{u}}^n$ and $\tilde{T}^n$ in Equations 23d and 24d, the proposed method directly corrects the velocity $\boldsymbol{u}^n$ and temperature T^n, ie

$$\boldsymbol{u}^n(\boldsymbol{x}_f, t) = \boldsymbol{u}^{n-1}(\boldsymbol{x}_f, t) + \frac{\delta_t}{2}\tilde{\boldsymbol{G}}^{n-1}(\boldsymbol{x}_f, t), \tag{39a}$$

$$T^n(\boldsymbol{x}_f, t) = T^{n-1}(\boldsymbol{x}_f, t) + \frac{\delta_t}{2}\tilde{Q}^{n-1}(\boldsymbol{x}_f, t). \tag{39b}$$

The iterative calculation of Equations 39a and 39b is equivalent to the definitions of velocity and temperature shown in Equations 26 and 30. After N_m iterations of the procedure, we obtain the velocity and temperature by $\boldsymbol{u}(\boldsymbol{x}_f, t) = \boldsymbol{u}^{N_m}(\boldsymbol{x}_f, t)$ and $T(\boldsymbol{x}_f, t) = T^{N_m}(\boldsymbol{x}_f, t)$. The force and the heat source are given as follows:

$$\tilde{\boldsymbol{G}}^{n-1}(\boldsymbol{x}_b, t) = \frac{2(\boldsymbol{u}^d(\boldsymbol{x}_b, t) - \boldsymbol{u}^{n-1}(\boldsymbol{x}_b, t))}{\delta_t}, \tag{40a}$$

$$\tilde{Q}^{n-1}(\boldsymbol{x}_b, t) = \frac{2(T^d(\boldsymbol{x}_b, t) - T^{n-1}(\boldsymbol{x}_b, t))}{\delta_t}. \tag{40b}$$

The velocity and temperature at $n-1$ iterations are interpolated by the following equations:

$$\boldsymbol{u}^{n-1}(\boldsymbol{x}_b, t) = \sum_f \boldsymbol{u}^{n-1}(\boldsymbol{x}_f, t) D(\boldsymbol{x}_f - \boldsymbol{x}_b)\delta_x^2, \tag{41a}$$

$$T^{n-1}(\boldsymbol{x}_b, t) = \sum_f T^{n-1}(\boldsymbol{x}_f, t) D(\boldsymbol{x}_f - \boldsymbol{x}_b)\delta_x^2. \tag{41b}$$

Because the proposed iterative method solves the kinetic Equations 25a, 25b, 29a, and 29b, we use the TRT collision operator with $\Lambda = \frac{9}{8}$ given by Equation 32 to reduce the velocity slip and the temperature jump. By iteratively correcting the velocity and temperature, we succeed in setting the numerical solution equal to the desired value at the boundary. Because the relation $\tau_+ = \tau_-$ gives the SRT collision operator, the proposed TRT method includes Kang's original SRT method.

The solution procedure of the iterative velocity and temperature-correction method with 2 relaxation times can be summarized as follows:

1. Give the initial density $\rho(\boldsymbol{x}_f)$, velocity $\boldsymbol{u}(\boldsymbol{x}_f)$, temperature $T(\boldsymbol{x}_f)$, desired velocity $\boldsymbol{u}^d(\boldsymbol{x}_b)$, and desired temperature $T^d(\boldsymbol{x}_b)$.
2. Obtain $\boldsymbol{u}^{n-1}(\boldsymbol{x}_b, t)$ and $T^{n-1}(\boldsymbol{x}_b, t)$ from $\boldsymbol{u}^{n-1}(\boldsymbol{x}_f, t)$ and $T^{n-1}(\boldsymbol{x}_f, t)$ using Equations 41a and 41b.

3. Compute $\tilde{\boldsymbol{G}}^{n-1}(\boldsymbol{x}_b, t)$ and $\tilde{Q}^{n-1}(\boldsymbol{x}_b, t)$ using Equations 40a and 40b.
4. Distribute $\tilde{\boldsymbol{G}}^{n-1}(\boldsymbol{x}_b, t)$ and $\tilde{Q}^{n-1}(\boldsymbol{x}_b, t)$ to the Eulerian nodes using Equations 23c and 24c.
5. Compute $\boldsymbol{u}^n(\boldsymbol{x}_f, t)$ and $T^n(\boldsymbol{x}_f, t)$ using Equations 39a and 39b.
6. Increase the number of iterations from $n-1$ to n. If $n \leq N_m$, return to step 2.
7. Obtain the velocity and temperature by $\boldsymbol{u}(\boldsymbol{x}_f, t) = \boldsymbol{u}^{N_m}(\boldsymbol{x}_f, t)$ and $T(\boldsymbol{x}_f, t) = T^{N_m}(\boldsymbol{x}_f, t)$.
8. Compute the force $\boldsymbol{G}(\boldsymbol{x}_f, t)$ and the heat source $Q(\boldsymbol{x}_f, t)$ using Equations 23e and 24e.
9. Compute $f_k^{\pm}(\boldsymbol{x}_f, t), f_k^{\pm eq}(\boldsymbol{x}_f, t), g_k^{\pm}(\boldsymbol{x}_f, t), g_k^{\pm eq}(\boldsymbol{x}_f, t)$, and $F_k^{\pm}(\boldsymbol{x}_f, t)$ using Equations 7 through 10, 27, and 28.
10. Solve Equations 25a, 25b, 29a, and 29b to obtain the distribution functions $f_k(\boldsymbol{x}_f, t+\delta_t)$ and $g_k(\boldsymbol{x}_f, t+\delta_t)$, and use Equations 26 and 30 to compute the macroscopic variables $\rho(\boldsymbol{x}_f, t+\delta_t)$, $\boldsymbol{u}(\boldsymbol{x}_f, t+\delta_t)$, and $T(\boldsymbol{x}_f, t+\delta_t)$.
11. Increase the time from t to $t+\delta_t$, and return to step 2.

2.5 | Iterative force and heat-correction method with 2 relaxation times

Zhang et al proposed an iterative force-correction method based on the split forcing scheme and the SRT collision operator for isothermal systems.[9,15] The thermal LBM based on the split forcing approach gives the kinetic equation with the TRT collision operator as follows:

$$f_k(\boldsymbol{x}+\boldsymbol{e}_k\delta_t, t+\delta_t) = f_k(\boldsymbol{x}, t) + \Omega_k^v(\boldsymbol{x}, t) + \frac{\delta_t}{2}\Big(F_k(\boldsymbol{x}, t) + F_k(\boldsymbol{x}+\boldsymbol{e}_k\delta_t, t+\delta_t)\Big), \tag{42a}$$

$$g_k(\boldsymbol{x}+\boldsymbol{e}_k\delta_t, t+\delta_t) = g_k(\boldsymbol{x}, t) + \Omega_k^c(\boldsymbol{x}, t) + \frac{\delta_t\omega_k\rho}{2}\Big(Q(\boldsymbol{x}, t) + Q(\boldsymbol{x}+\boldsymbol{e}_k\delta_t, t+\delta_t)\Big). \tag{42b}$$

The macroscopic values are defined by Equation 5. Because Equation 42a is equivalent to Equations 25a and 25b with Guo's external forcing term,[8] we can use $\Lambda = \frac{9}{8}$ given by Equation 32 to reduce the boundary slip. To implicitly compute $F_k(\boldsymbol{x}+\boldsymbol{e}_k\delta_t, t+\delta_t)$ and $Q(\boldsymbol{x}+\boldsymbol{e}_k\delta_t, t+\delta_t)$ in Equations 42a and 42b, we define the intermediate values f_k^{n-1}, g_k^{n-1}, F_k^{n-1}, and Q^{n-1} in the iterative procedure. Hence,

$$f_k^{n-1}(\boldsymbol{x}+\boldsymbol{e}_k\delta_t, t+\delta_t) = f_k(\boldsymbol{x}, t) + \Omega_k^v(\boldsymbol{x}, t) + \frac{\delta_t}{2}\Big(F_k(\boldsymbol{x}, t) + F_k^{n-1}(\boldsymbol{x}+\boldsymbol{e}_k\delta_t, t+\delta_t)\Big), \tag{43a}$$

$$g_k^{n-1}(\boldsymbol{x}+\boldsymbol{e}_k\delta_t, t+\delta_t) = g_k(\boldsymbol{x}, t) + \Omega_k^c(\boldsymbol{x}, t) + \frac{\delta_t\omega_k\rho}{2}\Big(Q(\boldsymbol{x}, t) + Q^{n-1}(\boldsymbol{x}+\boldsymbol{e}_k\delta_t, t+\delta_t)\Big). \tag{43b}$$

Considering the definition given by Equation 5, the difference between Equation 42a and Equation 43a, and the difference between Equation 42b and Equation 43b, we obtain the following equations:

$$\boldsymbol{u}(\boldsymbol{x}_f, t+\delta_t) - \boldsymbol{u}^{n-1}(\boldsymbol{x}_f, t+\delta_t) = \frac{\delta_t}{2}\Big(\boldsymbol{G}(\boldsymbol{x}_f, t+\delta_t) - \boldsymbol{G}^{n-1}(\boldsymbol{x}_f, t+\delta_t)\Big), \tag{44a}$$

$$T(\boldsymbol{x}_f, t+\delta_t) - T^{n-1}(\boldsymbol{x}_f, t+\delta_t) = \frac{\delta_t}{2}\Big(Q(\boldsymbol{x}_f, t+\delta_t) - Q^{n-1}(\boldsymbol{x}_f, t+\delta_t)\Big). \tag{44b}$$

Substituting $\boldsymbol{u}(t+\delta_t) = \boldsymbol{u}^d(t+\delta_t)$ and $T(t+\delta_t) = T^d(t+\delta_t)$ into Equations 44a and 44b at the boundary points $\boldsymbol{x}_b$ gives

$$\boldsymbol{G}^n(\boldsymbol{x}_b, t+\delta_t) = \boldsymbol{G}^{n-1}(\boldsymbol{x}_b, t+\delta_t) + \frac{2}{\delta_t}\Big(\boldsymbol{u}^d(\boldsymbol{x}_b, t+\delta_t) - \boldsymbol{u}^{n-1}(\boldsymbol{x}_b, t+\delta_t)\Big), \tag{45a}$$

$$Q^n(\boldsymbol{x}_b, t+\delta_t) = Q^{n-1}(\boldsymbol{x}_b, t+\delta_t) + \frac{2}{\delta_t}\Big(T^d(\boldsymbol{x}_b, t+\delta_t) - T^{n-1}(\boldsymbol{x}_b, t+\delta_t)\Big). \tag{45b}$$

Equations 45a and 45b are used for an iterative procedure that sets the boundary velocity and temperature equal to the desired values. The force and heat at the nth iteration are interpolated by the following equations:

$$\boldsymbol{G}^n(\boldsymbol{x}_f, t+\delta_t) = \sum_b^N \boldsymbol{G}^n(\boldsymbol{x}_b, t+\delta_t) D(\boldsymbol{x}_f - \boldsymbol{x}_b)\Delta s, \tag{46a}$$

$$Q^n(\boldsymbol{x}_f, t+\delta_t) = \sum_b^N Q^n(\boldsymbol{x}_b, t+\delta_t) D(\boldsymbol{x}_f - \boldsymbol{x}_b)\Delta s. \tag{46b}$$

The solution procedure of the iterative force and heat-correction method with 2 relaxation times can be summarized as follows:

1. Give the initial density $\rho(\boldsymbol{x}_f)$, velocity $\boldsymbol{u}(\boldsymbol{x}_f)$, temperature $T(\boldsymbol{x}_f)$, desired velocity $\boldsymbol{u}^d(\boldsymbol{x}_b)$, and desired temperature $T^d(\boldsymbol{x}_b)$. Initially, $\boldsymbol{G}(\boldsymbol{x}_b)$ and $Q(\boldsymbol{x}_b)$ are zero.
2. Compute $f_k^{\pm}(\boldsymbol{x}_f, t)$, $f_k^{\pm eq}(\boldsymbol{x}_f, t)$, $g_k^{\pm}(\boldsymbol{x}_f, t)$, and $g_k^{\pm eq}(\boldsymbol{x}_f, t)$ using Equations 7 through 10. Obtain $F_k(\boldsymbol{x}_f, t)$ using Equation 27.
3. Compute $\boldsymbol{u}^{n-1}(\boldsymbol{x}_b, t+\delta_t)$ and $T^{n-1}(\boldsymbol{x}_b, t+\delta_t)$ using Equations 41a and 41b.
4. Correct the boundary force $\boldsymbol{G}^{n-1}(\boldsymbol{x}_b, t+\delta_t)$ and the boundary heat $Q^{n-1}(\boldsymbol{x}_b, t+\delta_t)$ using Equations 45a and 45b.
5. Distribute $\boldsymbol{G}^n(\boldsymbol{x}_b, t+\delta_t)$ and $Q^n(\boldsymbol{x}_b, t+\delta_t)$ using Equations 46a and 46b.
6. Solve Equations 43a and 43b to obtain the distribution functions $f_k^n(\boldsymbol{x}_f, t+\delta_t)$ and $g_k^n(\boldsymbol{x}_f, t+\delta_t)$
7. Compute $\rho^n(\boldsymbol{x}_f, t+\delta_t)$, $\boldsymbol{u}^n(\boldsymbol{x}_f, t+\delta_t)$, and $T^n(\boldsymbol{x}_f, t+\delta_t)$ using Equation 5.
8. If $n \leq N_m$, return to step 3.
9. Obtain the macroscopic values by $\boldsymbol{u}(\boldsymbol{x}_f, t+\delta_t) = \boldsymbol{u}^{N_m}(\boldsymbol{x}_f, t+\delta_t)$, $T(\boldsymbol{x}_f, t+\delta_t) = T^{N_m}(\boldsymbol{x}_f, t+\delta_t)$, $\boldsymbol{G}(\boldsymbol{x}_f, t+\delta_t) = \boldsymbol{G}^{N_m}(\boldsymbol{x}_f, t+\delta_t)$, and $Q(\boldsymbol{x}_f, t+\delta_t) = Q^{N_m}(\boldsymbol{x}_f, t+\delta_t)$.
10. The distribution functions $f_k(\boldsymbol{x}_f, t+\delta_t)$ and $g_k(\boldsymbol{x}_f, t+\delta_t)$ are given by $f_k^{N_m}(\boldsymbol{x}_f, t+\delta_t)$ and $g_k^{N_m}(\boldsymbol{x}_f, t+\delta_t)$, respectively.
11. Increase the time from t to $t+\delta_t$, and return to step 2.

The novelty of this study is to analytically provide the relationship between τ_+ and τ_- that eliminates the velocity slip and the temperature jump for the 5 types of IB-TLBMs. Although the nonslip boundary condition is exactly satisfied by the multi-direct forcing method, the multi-direct forcing method cannot completely reduce the boundary slip for any relaxation time (see Appendix A). By applying an iterative correction technique based on the external forcing approach proposed by Guo or Cheng,[14,15] we allow the IB-TLBMs to eliminate the boundary slip for any relaxation time. The next section validates these 5 IB-TLBMs against benchmark tests involving symmetric shear flows, heat transfer between 2 porous plates, flow past a circular cylinder, and natural convection.

3 | RESULTS AND DISCUSSION

3.1 | Theoretical analysis

We theoretically verify the effect of the TRT collision model on the reduction of the boundary slip, which is described as a velocity slip or a temperature jump in the IB-TLBM.

3.1.1 | Analysis of the IB-LBMs for symmetric shear flows

In the calculation of the symmetric shear flows shown in Figure 1A, we have

$$\rho = \text{const}, \quad v = 0, \frac{du}{dx} = 0, \quad \text{and } \frac{du}{dt} = 0, \tag{47}$$

where $\boldsymbol{u} = (u, v)$ and $\boldsymbol{G} = (G, 0)$ in 2 dimensions.

The computational domain with the periodic boundary is covered by 200×200 grid points. In the simulation of the symmetric shear flows, 2 plates moving at equal and opposite horizontal velocities u^d and $-u^d$ are placed at $y = 50\delta_x$ and $y = 150\delta_x$. The velocity $u^d = 0.01c$ is set at all Lagrangian points of the boundary, and the distance between the plates is $h = 100$. $\delta_x = \delta_t = c = 1$ are used. The convergence criterion for all calculations is

$$\max \frac{||\boldsymbol{u}(t+\delta_t)| - |\boldsymbol{u}(t)||}{u_0} \leq 10^{-8}, \tag{48}$$

where u_0 denotes the characteristic velocity. When the numerical solution does not satisfy Equation 48 because of numerical oscillation, we use the results at time $t = 100000\delta_t$ instead of the converged solution.

Figure 2 shows the profiles of the velocity calculated using the 5 kinds of IB-LBM, where the velocities are normalized with respect to the characteristic velocity. Although the TRT collision operator succeeds in removing the boundary slip, the boundary-deviation remains in the direct forcing method in Figure 2A. The multi-direct forcing method, the implicit

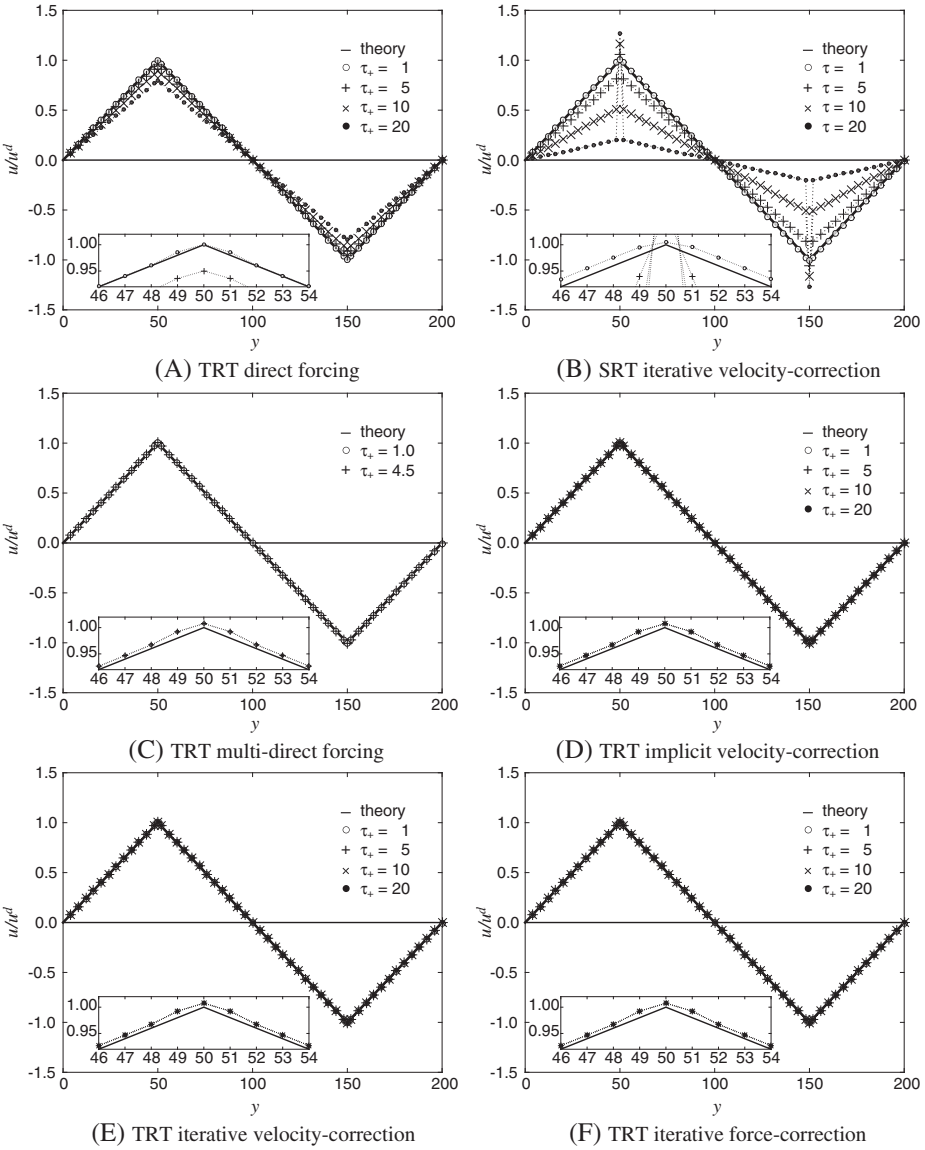


FIGURE 2 Velocity profiles obtained through calculation of the symmetric shear flows. The thick line indicates the exact velocity profile

velocity-correction method, the iterative velocity-correction method (implicit diffuse direct forcing method), and the iterative force-correction method ensure that the boundary velocity agrees with the desired velocity and prevent streamline penetration through the solid body. These advantages over the SRT collision operator, however, do not contribute to reducing the velocity slip at high relaxation time as shown in Figure 2B. In our previous study,[21] we proved both numerically and theoretically that the TRT collision model is as effective in reducing the boundary slip as the MRT collision model. Based on the relation shown in Equation 22, the TRT multi-direct forcing method eliminates the velocity slip in Figure 2C. When τ_+^v is larger than 5, τ_-^v in Equation 22 is less than 0.5. The multi-direct forcing method uses Equation 22 only when $\tau_+^v < 5$ to guarantee numerical stability. By satisfying Equation 32, the TRT implicit velocity-correction method, the TRT iterative velocity-correction method, and the TRT iterative force-correction method successfully reduce the velocity slip for high relaxation time in Figure 2D, 2E, and 2F. Deviation between the numerical value and the desired value at the boundary is observed in Figure 2C, 2D, 2E, and 2F. The boundary-value deviation is caused by the error of the smooth delta function used in the interpolation process for $\boldsymbol{G}$ and $\boldsymbol{u}$, as shown in Equation 19.[21]

In the LBM framework, Karimipour et al succeeded in implementing the linear Navier boundary condition, which defines the velocity slip Δu_{wall} as follows:

$$\Delta u_{wall} = u_{fluid}(y \to \text{wall}) - u_{wall} = L_s \frac{\partial u_{fluid}(y)}{\partial y}, \tag{49}$$

where u_{wall} is the wall velocity, and L_s is the slip length.[30] In the IB-LBM, u_{wall} in Equation 49 is equivalent to the following velocity $\bar{u}_{j_0}$:

$$\bar{u}_{j_0} = \frac{G_0 \delta_x^2}{2\nu} \frac{h}{2}, \tag{50}$$

where j_0 denotes the position of the wall, and G_0 is the total boundary force calculated by the IBM procedure at the boundary node. The numerical solution of the fluid velocity, u_{j_0}, computed by the LBM is equal to $u_{fluid}(y \to wall)$ in Equation 49. In this study, we can obtain the artificial slip velocity from the formulation, $u^s_{j_0} = u_{j_0} - \bar{u}_{j_0}$.

In our previous study,[21] we obtained analytical solutions for the normalized boundary slip velocity $u^s_{j_0}/u^d$ and the normalized boundary velocity u_{j_0}/u^d:

$$\frac{u^s_{j_0}}{u^d} = \frac{-\frac{1}{4} - \frac{\tau^v_+}{4} - \frac{\tau^v_-}{6} + \frac{\tau^v_+\tau^v_-}{3}}{\frac{h}{4} - \frac{13}{24} + \frac{7\tau^v_+}{48} - \frac{\tau^v_-}{8} + \frac{\tau^v_+\tau^v_-}{4}}, \tag{51}$$

$$\frac{u_{j_0}}{u^d} = \frac{\frac{h}{4} - \frac{1}{4} - \frac{\tau^v_+}{4} - \frac{\tau^v_-}{6} + \frac{\tau^v_+\tau^v_-}{3}}{\frac{h}{4} - \frac{13}{24} + \frac{7\tau^v_+}{48} - \frac{\tau^v_-}{8} + \frac{\tau^v_+\tau^v_-}{4}}, \tag{52}$$

for the TRT direct forcing method, and

$$\frac{u^s_{j_0}}{u^d} = \frac{-\frac{7}{24} - \frac{\tau^v_+}{6} - \frac{\tau^v_-}{6} + \frac{\tau^v_+\tau^v_-}{3}}{\frac{h}{4} - \frac{13}{32} - \frac{\tau^v_+}{8} - \frac{\tau^v_-}{8} + \frac{\tau^v_+\tau^v_-}{4}}, \tag{53}$$

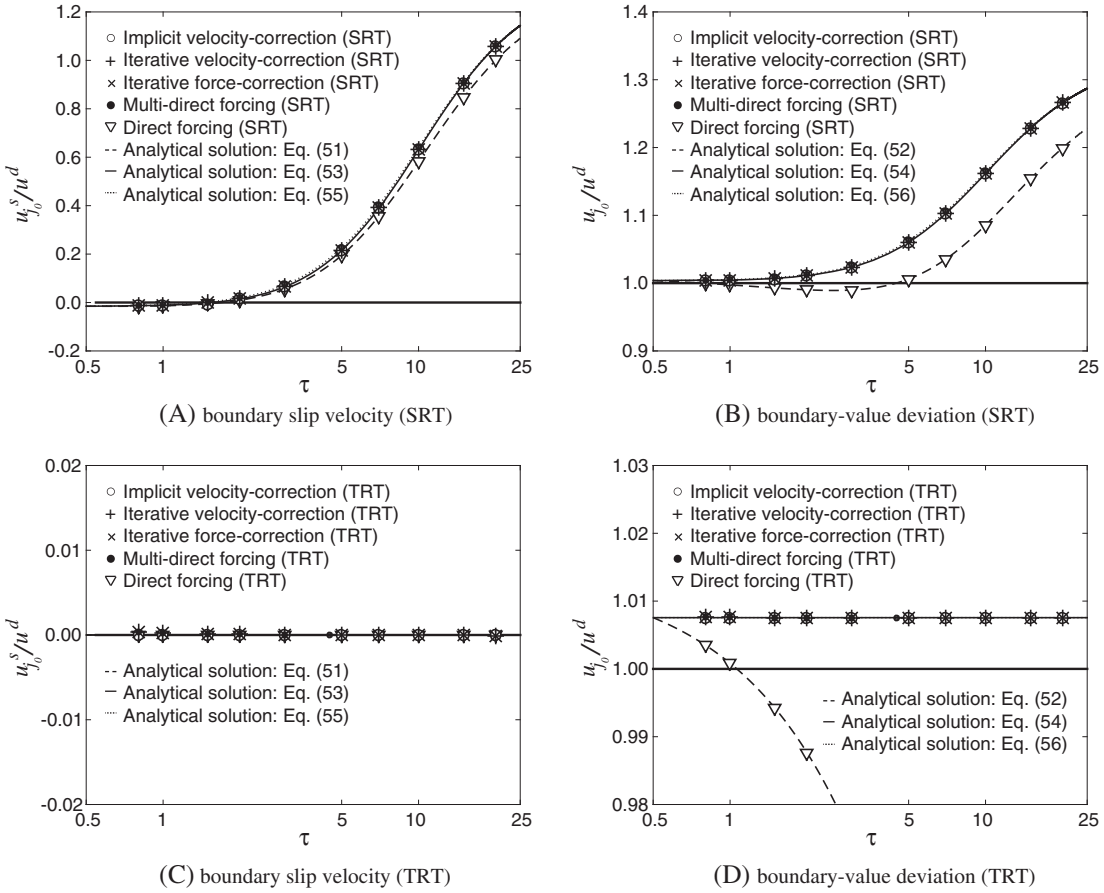
$$\frac{u_{j_0}}{u^d} = \frac{\frac{h}{4} - \frac{7}{24} - \frac{\tau^v_+}{6} - \frac{\tau^v_-}{6} + \frac{\tau^v_+\tau^v_-}{3}}{\frac{h}{4} - \frac{13}{32} - \frac{\tau^v_+}{8} - \frac{\tau^v_-}{8} + \frac{\tau^v_+\tau^v_-}{4}}, \tag{54}$$

for the TRT implicit correction method in the symmetric shear flows. Because the iterative velocity-correction method solves the same Equations 25a through 27 as the implicit correction method, the velocity slip $u^s_{j_0}$ and the boundary value u_{j_0} calculated by the iterative velocity-correction method should obey Equations 53 and 54. Because Equations 5 and 42a are equivalent to Equations 25a through 27, the analytical solutions for the iterative force-correction method are also given by Equations 53 and 54. As indicated in Appendix A, we obtain the analytical solutions for the TRT multi-direct forcing method as follows:

$$\frac{u^s_{j_0}}{u^d} = \frac{-\frac{1}{3} - \frac{\tau^v_+}{12} - \frac{\tau^v_-}{6} + \frac{\tau^v_+\tau^v_-}{3}}{\frac{h}{4} - \frac{7}{16} - \frac{\tau^v_+}{16} - \frac{\tau^v_-}{8} + \frac{\tau^v_+\tau^v_-}{4}}, \tag{55}$$

$$\frac{u_{j_0}}{u^d} = \frac{\frac{h}{4} - \frac{1}{3} - \frac{\tau^v_+}{12} - \frac{\tau^v_-}{6} + \frac{\tau^v_+\tau^v_-}{3}}{\frac{h}{4} - \frac{7}{16} - \frac{\tau^v_+}{16} - \frac{\tau^v_-}{8} + \frac{\tau^v_+\tau^v_-}{4}}. \tag{56}$$

Figure 3 shows the numerical results computed by the IB-LBMs and the analytical solutions. The horizontal axis indicates the relaxation time τ including τ^v_+. The dashed lines, solid lines, and dotted lines represent the analytical solutions for the direct forcing method, the implicit velocity-correction method, and the multi-direct forcing method, respectively. The numerical solutions are denoted by symbols. Substituting $\tau^v_+ = \tau^v_-$ into Equations 51 through 56 gives the analytical solutions for the SRT collision model. In Figure 3A and 3B, any IB-LBMs with the SRT collision operator can not reduce the velocity slip. As predicted from the analytical solutions (51), (53), and (55), the IB-LBMs delete the boundary slip by using the TRT collision operator as shown in Figure 3C. The good agreement for the velocity at the boundary between the numerical and analytical solutions demonstrates the validity of Equations 51 through 56 in Figure 3. In Figure 3C and 3D, the multi-direct forcing method cannot reduce the boundary slip for all values of the relaxation times τ^v_+, because it should use Equation 22 to eliminate the velocity slip with $\tau^v_+ < 5$. The iterative velocity-correction method and the



(A) boundary slip velocity (SRT) (B) boundary-value deviation (SRT)

(C) boundary slip velocity (TRT) (D) boundary-value deviation (TRT)

FIGURE 3 Comparison of analytical and numerical solutions depending on the relaxation time. The horizontal axis indicates the relaxation time τ, including τ_+^v

iterative force-correction method are as effective as the implicit velocity-correction method in reducing the boundary slip and the boundary-value deviation for any relaxation parameters. It is analytically and numerically proved that the TRT collision operator can reduce the boundary slip in Figure 3C, and the iterative procedure based on Guo's forcing term or on Cheng's split forcing scheme is necessary to enforce the boundary condition at the boundary points for any relaxation parameters in Figure 3D.

3.1.2 | Heat transfer between 2 porous plates

In this section, we numerically and theoretically investigate the temperature jump of the IB-TLBM schemes. The convergence criterion for all calculations is

$$max|T(t+\delta_t) - T(t)| \leq 10^{-8}. \tag{57}$$

To verify the effect of the advection velocity v_0 on the temperature jump of the IB-TLBM, we analyze the temperature distribution for simple two-dimensional steady heat transfer between the porous plates:

$$\rho = \text{const}, \quad u = 0, v = \text{const}, \quad \frac{\partial T}{\partial x} = 0, \quad \text{and } \frac{\partial T}{\partial t} = 0. \tag{58}$$

As shown in Figure 1B, a constant advection velocity, v_0, drives the fluid in the vertical direction (y-axis). The velocities of the hot bottom plate with temperature T_0 and the cold top plate with temperature T_1 are set to v_0.

The exact temperature profile in the steady state satisfies

$$T = T_0 + (T_1 - T_0)\frac{e^{\frac{v_0(y-50)}{\chi}} - 1}{e^{\frac{v_0 h}{\chi}} - 1}, \tag{59}$$

for $50 \leq y$ and $y \leq 150$. Here, $h = 100$ is the distance between the 2 plates. To compare the numerical solutions to the same theoretical solution shown in Equation 59 for any relaxation time, we set v_0/χ equal to 0.02.

Figure 4 shows the temperature profiles for the porous-plate problem calculated by the IB-TLBMs. Based on the relation given by Equation 15, the TRT direct forcing method eliminates the temperature jump in Figure 4A. Similarly to the numerical results shown in Figure 2A, the TRT direct forcing method does not eliminate the disagreement between the numerical results and the desired value T^d. Although the boundary-value deviation is eliminated by the iterative temperature-correction method, the SRT collision operator cannot reduce the temperature jump in Figure 4B. In Figure 4C through 4F, the combination of the TRT collision operator and the 4 types of iterative or implicit correction methods succeeds in reducing the temperature jump observed in Figure 4B. Since τ_+^c should be greater than 0.5 in Equation 22, the TRT multi-direct forcing method can delete the temperature jump when $\tau_-^c < 5$.

Following the procedure shown in Appendix B, we obtain analytical solutions for the porous-plate problem. Appendix B derives the following macroscopic temperature equation with a constant velocity:

$$v_0 \frac{T_{j+1} - T_{j-1}}{2\delta_x} = \tilde{\chi} \frac{T_{j+1} - 2T_j + T_{j-1}}{\delta_x^2} + Q_j + \phi(Q_{j+1} - 2Q_j + Q_{j-1}), \tag{60}$$

where $\tilde{\chi} = \chi(1 + \frac{3v_0^2}{c^2})$. Here, ϕ is given by

$$\phi = \frac{2\tau_+^c + 3\tau_-^c - 4\tau_+^c\tau_-^c}{6}, \tag{61}$$

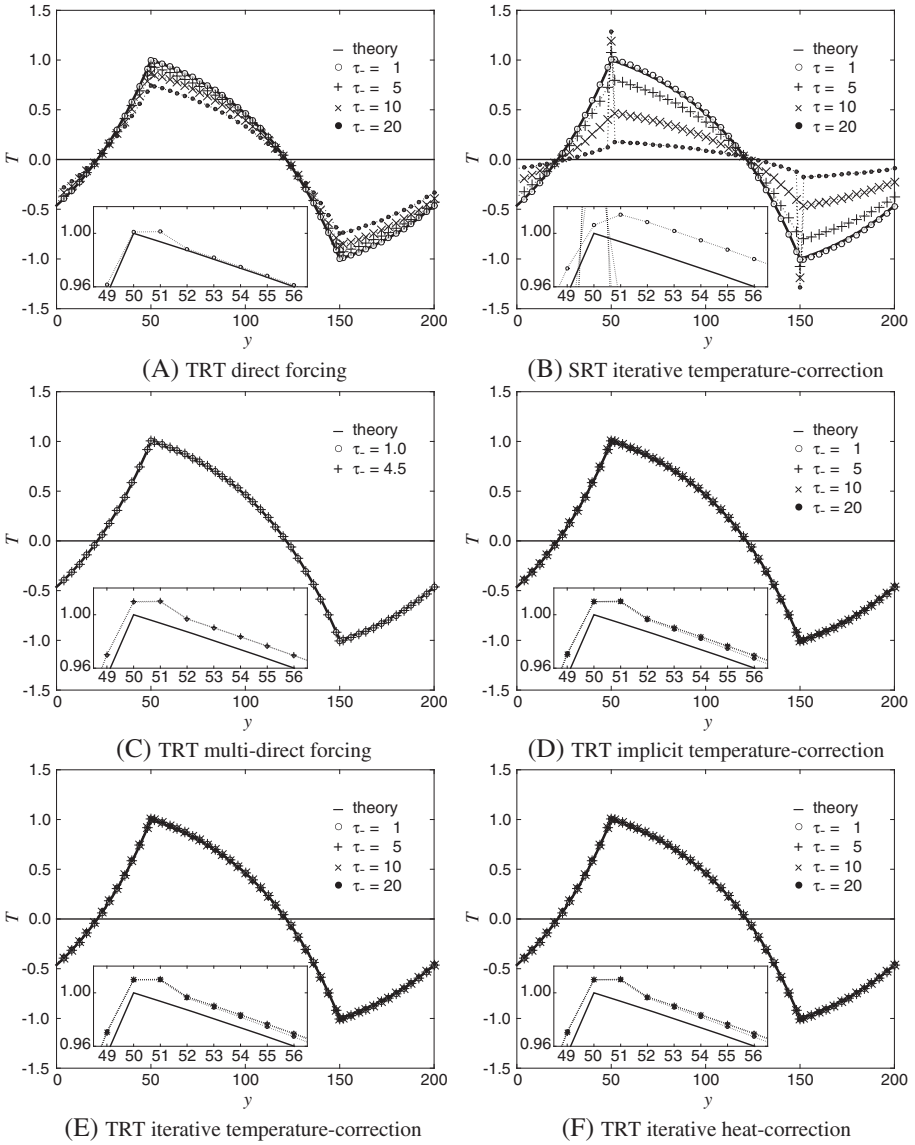


FIGURE 4 Temperature profiles calculated by IB-TLBMs for the porous-plate problem. The solid line indicates the exact temperature profile

for the direct forcing method,

$$\phi = \frac{1 + 2\tau_+^c + \tau_-^c - 4\tau_+^c\tau_-^c}{6}, \tag{62}$$

for the multi-direct forcing method, and

$$\phi = \frac{1 + 4\tau_+^c + 4\tau_-^c - 8\tau_+^c\tau_-^c}{12}, \tag{63}$$

for the implicit temperature-correction method, the iterative temperature-correction method, and the iterative heat-correction method.

Considering $T(\boldsymbol{x}_b) = T_b + \delta_t Q_0$, Appendix B gives the analytical solutions for the temperature jump, $T_{j_0}^J/T^d$, and the boundary value, T_{j_0}/T^d:

$$\frac{T_{j_0}^J}{T^d} = \frac{(\frac{\delta_x^2 v_0^2}{4\tilde{\chi}^2} - 1)\frac{(1-\gamma)(1+\gamma^{h-1}+\phi(1+\gamma)(1+\gamma^{h-2}))}{(1+\gamma)(1-\gamma^{h-1}-\phi(1-\gamma)(1+\gamma^{h-2}))}}{1 - \frac{(2\tilde{\chi}+\delta_x v_0)(1-\gamma)(1+\gamma^h)(\tilde{\chi}-\tilde{\chi}\phi-8\tilde{\chi}^2+2\delta_x^2 v_0^2)}{4\tilde{\chi}^2(1+\gamma)(1-\gamma^{h-1}-\phi(1-\gamma)(1+\gamma^{h-2}))}}, \tag{64}$$

$$\frac{T_{j_0}}{T^d} = \frac{1 - \frac{\delta_x^2 v_0^2}{4\tilde{\chi}^2}}{1 - \frac{(2\tilde{\chi}+\delta_x v_0)(1-\gamma)(1+\gamma^h)(\tilde{\chi}-\tilde{\chi}\phi-8\tilde{\chi}^2+2\delta_x^2 v_0^2)}{4\tilde{\chi}^2(1+\gamma)(1-\gamma^{h-1}-\phi(1-\gamma)(1+\gamma^{h-2}))}}, \tag{65}$$

for the direct forcing method. Using the equation $T_b = T^d$, we obtain the following analytical solutions:

$$\frac{T_{j_0}^J}{T^d} = \frac{(\frac{\delta_x^2 v_0^2}{4\tilde{\chi}^2} - 1)\frac{(1-\gamma)(1+\gamma^{h-1}+\phi(1+\gamma)(1+\gamma^{h-2}))}{(1+\gamma)(1-\gamma^{h-1}-\phi(1-\gamma)(1+\gamma^{h-2}))}}{1 - \frac{(2\tilde{\chi}+\delta_x v_0)(1-\gamma)(1+\gamma^h)(1-\phi)}{4\tilde{\chi}(1+\gamma)(1-\gamma^{h-1}-\phi(1-\gamma)(1+\gamma^{h-2}))}}, \tag{66}$$

$$\frac{T_{j_0}}{T^d} = \frac{1 - \frac{\delta_x^2 v_0^2}{4\tilde{\chi}^2}}{1 - \frac{(2\tilde{\chi}+\delta_x v_0)(1-\gamma)(1+\gamma^h)(1-\phi)}{4\tilde{\chi}(1+\gamma)(1-\gamma^{h-1}-\phi(1-\gamma)(1+\gamma^{h-2}))}}, \tag{67}$$

for the multi-direct forcing method, the implicit temperature-correction method, the iterative temperature-correction method, and the iterative heat-correction method. Here, $\gamma = (2\tilde{\chi} - \delta_x v_0)/(2\tilde{\chi} + \delta_x v_0)$. When τ_+^c and τ_-^c are replaced by τ^c in Equations 64 through 67, we obtain the analytical solutions for the IB-TLBM with the SRT collision operator.

To reduce the temperature jump in Equations 64 and 66, ϕ should satisfy the following relation:

$$\phi = -\frac{1+\gamma^{h-1}}{1+\gamma+\gamma^{h-2}+\gamma^{h-1}}. \tag{68}$$

Because $v_0/\chi = 0.02$, $\delta_x = 1$, and $\gamma \approx 1$, Equation 68 gives $\phi \approx -1/2$. From $\phi \approx -1/2$, we can use the relation between τ_- and τ_+ given by Equations 15, 22, and 32 for the porous-plate problem to reduce the temperature jump. The TRT collision operator is applied to the IB-TLBM to reduce the temperature jump. The multi-direct forcing method, the implicit/iterative temperature-correction methods, and the iterative heat-correction method eliminate the boundary-value deviation from the desired temperature.

Following the definition of the temperature jump in Karimipour,[31] the temperature jump $T_{j_0}^J$ is given by

$$T_{j_0}^J = T_{j_0} - \bar{T}_{j_0}. \tag{69}$$

The temperature $\bar{T}_{j_0}$ at the boundary node is given by the following equation:

$$\bar{T}_{j_0} = \frac{(1-\gamma^h)\delta_x^2 Q_0}{(1-\gamma)(1+\gamma^h)(2\tilde{\chi}+\delta_x v_0)}, \tag{70}$$

where Q_0 is the total heat source calculated by the IB-TLBM.

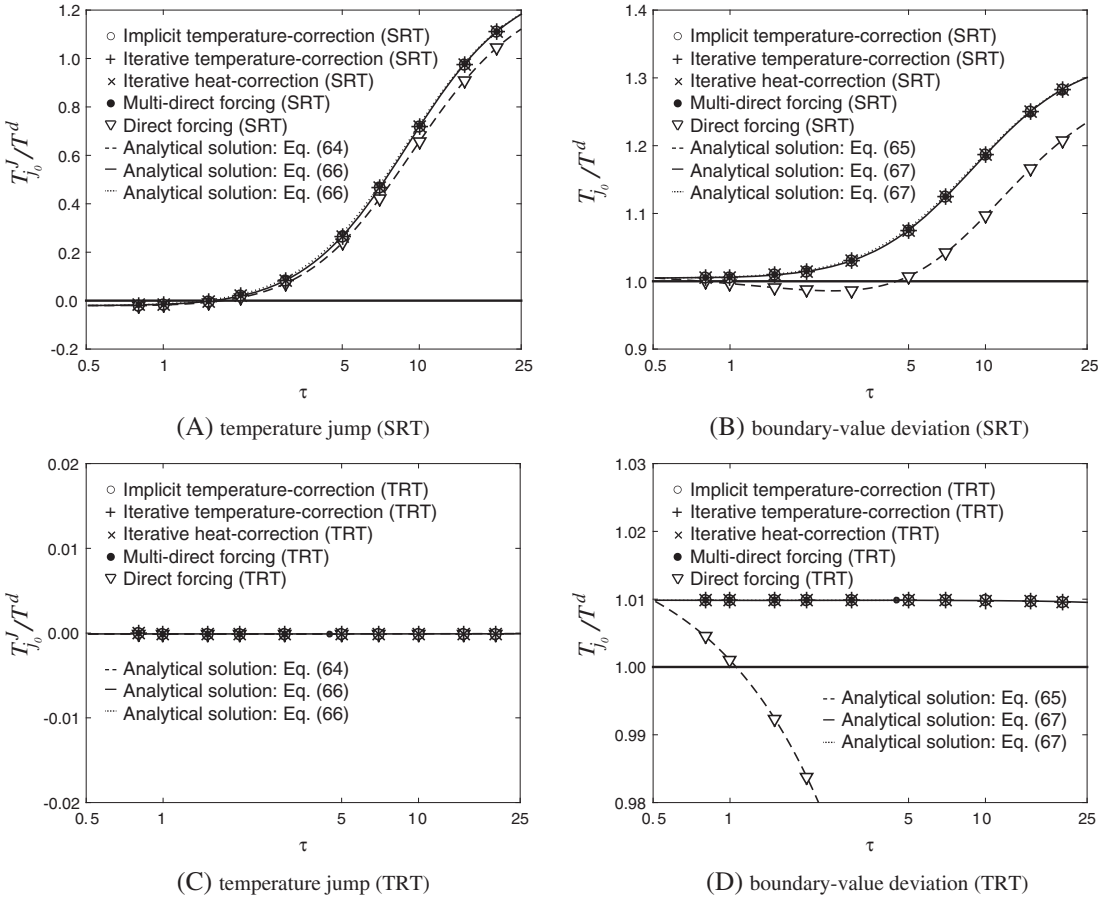


FIGURE 5 Analytical and numerical solutions for the IB-TLBMs depending on relaxation time. The horizontal axis indicates the relaxation time τ including τ_-^c

In Figure 5, we compare the analytical solutions and numerical results. Figure 5A and 5B shows that the numerical results agree well with the expected theoretical results for the SRT collision operator. As predicted based on the analytical solutions given by Equations 64 and 66, the temperature jump disappears for any value of relaxation time τ_-^c when using the TRT collision operator, as shown in Figure 5C. The choice $\Lambda = \frac{9}{8}$ for the iterative temperature/heat-correction methods completely eliminates the temperature jump in Figure 5C, as does the implicit temperature-correction method. Although the implicit temperature-correction method solves the simultaneous equations given by Equation 35a to make $T(\boldsymbol{x}_b) = T^d$, T_{j_0} is not equal to T^d in Figure 5D. The iterative temperature/heat-correction methods also do not succeed in making T_{j_0} equal to T^d in Figure 5D. Similarly to the boundary-value deviation of the velocity shown in Figure 2, the boundary temperature error in T_{j_0}/T^d of the TRT iterative/implicit correction methods is caused by the smoothing technique of the delta function. As in the results shown in Figures 4 and 5, the iterative heat-correction method exhibits the same accurate results as the iterative temperature-correction method. By splitting the heat source term as follows: $Q(\boldsymbol{x}, t)+Q(\boldsymbol{x}+\boldsymbol{e}_k\delta_t, t+\delta_t)$ in Equation 42b, the iterative heat-correction method succeeds in rigorously satisfying the Dirichlet boundary condition for the temperature. The implicit temperature-correction method must solve the simultaneous equation with matrix size $O(N^2)$ to calculate the temperature distribution around the wall. With the low computational cost of $O(N \times N_m)$, the proposed iterative temperature/heat-correction methods effectively reduce the temperature jump and the boundary-value deviation as well as the implicit correction method.

3.2 | Numerical analysis

We numerically investigate the effects of the reduction of the boundary slip on the accuracy and numerical stability of the proposed iterative correction methods in the simulation of the cylindrical Couette flow, the flow past a circular cylinder, and natural convection.

3.2.1 | Cylindrical Couette flow

We examine the accuracy of the IB-TLBMs, which accurately enforce the nonslip boundary condition, in the calculation of the two-dimensional cylindrical Couette flow. As shown in Figure 1C and 1D, 2 rings are placed at the center of the simulation domain and are covered by $200\delta_x \times 200\delta_x$ grid points. The outer ring is at rest, and the velocity of the inner cylinder, u_θ^d, is $0.01c$. The inner and outer cylinders are maintained at temperatures of $T_i = 1.0$ and $T_o = 0.0$, respectively. The initial values are given by $\rho = 1.0$, $\boldsymbol{u} = \boldsymbol{0}$, and $T = 0.5$. The radii of the outer ring, R_o, and the inner ring, R_i, are $70\delta_x$ and $45\delta_x$, respectively. In the calculation of the cylindrical Couette flows, the discrete length Δs on the inner cylinder is approximately equal to the discrete length on the outer cylinder for all IB-TLBMs. We set 282 Lagrangian points on the inner cylinder and 439 points on the outer cylinder, making $\Delta s \approx \delta_x$.

The exact solutions for the tangential velocity and the temperature of the cylindrical Couette flow, respectively, are given by the following equations:

$$\hat{u}_\theta(R) = u_\theta^d \frac{R/R_o - R_o/R}{R_i/R_o - R_o/R_i}, \tag{71a}$$

$$\hat{T}(R) = \frac{T_o log(R/R_i) - T_i log(R/R_o)}{log(R_o/R_i)}, \tag{71b}$$

where R is the radial coordinate. The exact solution for the torque $\hat{M}_\theta$ acting on the inner cylinder is analytically derived as

$$\hat{M}_\theta = \frac{4\pi\nu u_\theta^d R_o}{R_i/R_o - R_o/R_i}. \tag{72}$$

The pressure is known to be constant (ie, $\hat{p} = c^2\rho/3$).

Figure 6 shows the contour lines of the tangential velocity, temperature, and pressure calculated using the IB-TLBMs at $\tau_+^\nu = 5$ and $\tau_+^c = 10$. The left- and right-hand figures show the numerical results calculated by the SRT collision model and the TRT collision model, respectively. The TRT direct forcing method uses the relation between τ_+^ν and τ_-^ν and between τ_+^c and τ_-^c shown in Equation 15. The relaxation parameters $\tau_\pm^\nu$ and $\tau_\pm^c$ satisfy $\Lambda = 9/8$ in the TRT implicit/iterative correction methods. Figure 6 shows that the velocity slip and the temperature jump of the IB-TLBMs with the SRT collision operator induce distortions of the tangential velocity and the temperature around the rings. All IB-TLBMs with the TRT collision operator reduce the boundary slip and the temperature jump and provide smooth profiles of the tangential velocity in Figure 6. The TRT direct forcing method does not succeed in correctly setting the velocity and temperature at the boundary in Figure 6A. As shown in Figure 6B, 6C, and 6D, the TRT implicit/iterative correction methods yield adequate results that agree with the analytical solutions given by Equations 71a and 71b. From the comparison of the results shown in Figure 6B, 6C, and 6D, the iterative correction methods show similar velocity and temperature distributions to the implicit correction method not only for the TRT collision model but also for the SRT collision model. As is clear from the undesirable pressure oscillations around the rings observed in Figure 6(b-3), the pressure profiles of these implicit/iterative correction methods are not similar.

To examine the accuracy of the IB-TLBMs, we define the relative errors as follows:

$$Error = \sqrt{\frac{\sum_{\boldsymbol{x}\in\text{fluid}}(u_\theta(\boldsymbol{x}) - \hat{u}_\theta(\boldsymbol{x}))^2}{\sum_{\boldsymbol{x}\in\text{fluid}}\hat{u}_\theta(\boldsymbol{x})^2}}, \tag{72a}$$

$$Error = \sqrt{\frac{\sum_{\boldsymbol{x}\in\text{fluid}}(T(\boldsymbol{x}) - \hat{T}(\boldsymbol{x}))^2}{\sum_{\boldsymbol{x}\in\text{fluid}}\hat{T}(\boldsymbol{x})^2}}, \tag{72b}$$

$$Error = \sqrt{\frac{(M_\theta - \hat{M}_\theta)^2}{\hat{M}_\theta^2}}, \tag{72c}$$

$$Error = \sqrt{\frac{\sum_{\boldsymbol{x}\in\text{fluid}}(p(\boldsymbol{x}) - \hat{p}(\boldsymbol{x}))^2}{\sum_{\boldsymbol{x}\in\text{fluid}}\hat{p}(\boldsymbol{x})^2}}, \tag{72d}$$

where fluid represents the region between the outer and inner rings shown in Figure 6.

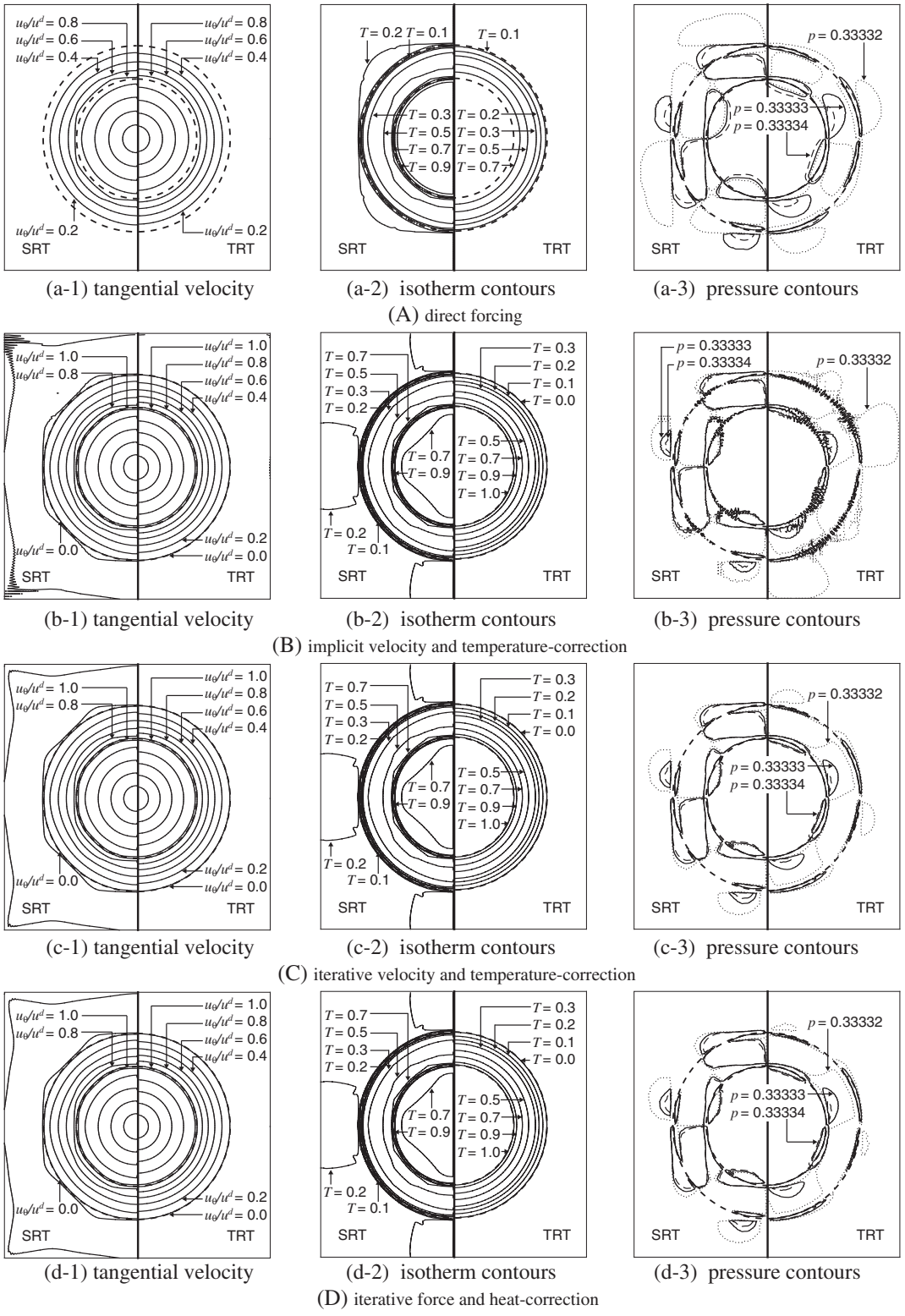


FIGURE 6 Tangential velocity, pressure, temperature profiles as calculated by the IB-TLBMs in cylindrical Couette flows. The dashed circles indicate the position of the rings. The relaxation parameters are given by $\tau_+^v = 5$, and $\tau_-^c = 10$

Figure 7 shows the relation between the accuracy of the IB-TLBMs and the number of grid points on the side of the square computational domain in the simulation of the cylindrical Couette flow. The boundary slip caused by the use of the SRT model adversely affects the accuracy of the IB-TLBM. The symbols (△), (•), and (∘) in Figure 7 indicate the same values for the SRT implicit/iterative correction methods.

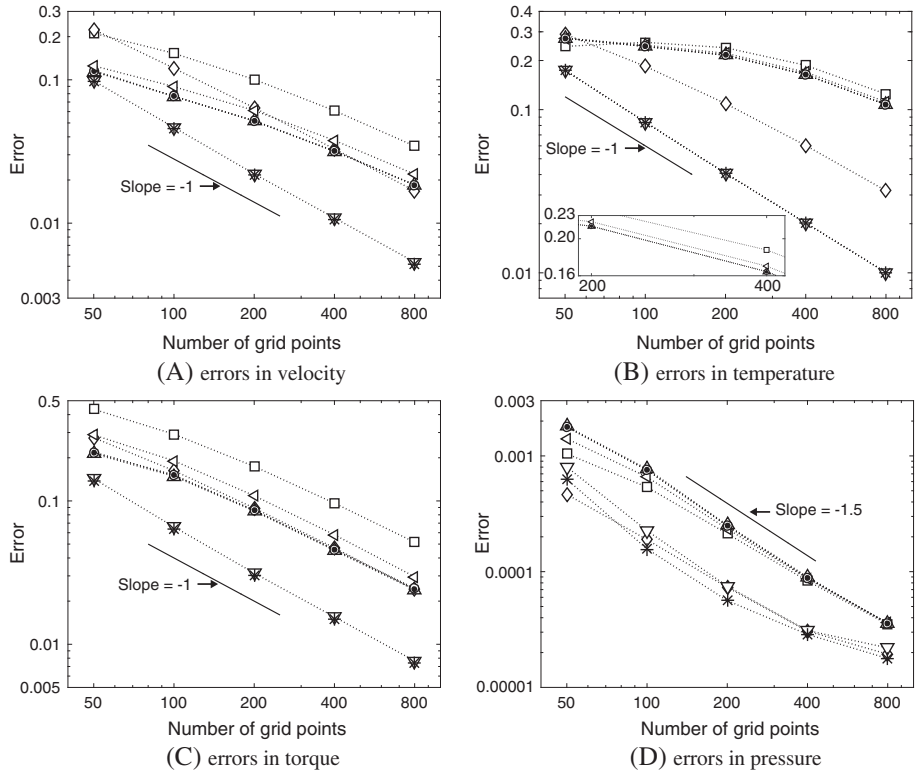


FIGURE 7 Relative error versus number of grid points in the calculation of the cylindrical Couette flows. The symbols (□), (◁), (△), (•), (∘), (⋄), (▽), (+), and (×) denote the numerical results calculated by the SRT direct forcing method, the SRT multi-direct forcing method, the SRT implicit velocity and temperature-correction method, the SRT iterative velocity and temperature-correction method, the SRT iterative force and heat-correction method, the TRT direct forcing method, the TRT implicit velocity and temperature-correction method, the TRT iterative velocity and temperature-correction method, and the TRT iterative force and heat-correction method, respectively

The symbol (◁) denotes the error of the SRT multi-direct forcing method. Since the multi-direct forcing method uses the conventional velocity definition and forcing term given by Equations 5 and 21b, the SRT multi-direct forcing method does not have the same accuracy as SRT implicit/iterative correction methods. The redefinition of the velocity and forcing term given by Equations 26 and 27 can remove the influence of spatial and temporal variations of the force in the macroscopic equations, which is induced by Equations 5 and 21b.[14] The errors denoted by the symbols (▽), (+), and (×) for the TRT implicit/iterative correction methods agree well with each other in Figure 7A, 7B, and 7C. All IB-TLBMs with the TRT collision operator exhibit first-order accuracy in space for the velocity, temperature, and torque. The algorithms of the iterative correction methods are simpler than the implicit correction method, and their accuracy is exactly the same as the implicit correction method for the velocity, the temperature, and the torque. Because of the pressure oscillation shown in Figure 6(b-3), the error in the pressure for the implicit correction method denoted by the symbol (▽) is larger than for the iterative correction methods denoted by the symbols (+) and (×) in Figure 7D. The proposed TRT iterative correction methods show the most accurate results in Figure 7.

We verify the effects of the discrete length Δs on the accuracy and the numerical stability of the IB-TLBMs. Figure 8A and 8B shows the velocity and the temperature profiles as calculated by 2 types of the IB-TLBMs in the cylindrical Couette flows at $\Delta s = 0.4$, respectively. The results of the TRT implicit velocity and temperature-correction method and of the TRT iterative velocity and temperature-correction method are shown in the left- and right-hand sides of Figure 8A and 8B, respectively. The relaxation parameters are given by $\tau_+^v = 5$ and $\tau_-^c = 10$. In Figure 8A and 8B, the TRT implicit correction method induces numerical oscillation of the velocity and the temperature around the rings for the small discrete length Δs. In contrast, the proposed TRT iterative correction method does not exhibit such numerical oscillation. Figure 8C and 8D show the relation between the discrete length and the relative error defined by Equations 72a and 72b. The errors for the TRT iterative correction methods are constant for any values of the discrete length. When $\Delta s < 0.3$, the divergence is demonstrated in the calculation using the TRT implicit correction method, as estimated from the symbol (∘) in Figure 8C and 8D. Since the accuracy of the TRT implicit temperature-correction method depends on the discrete length, we must

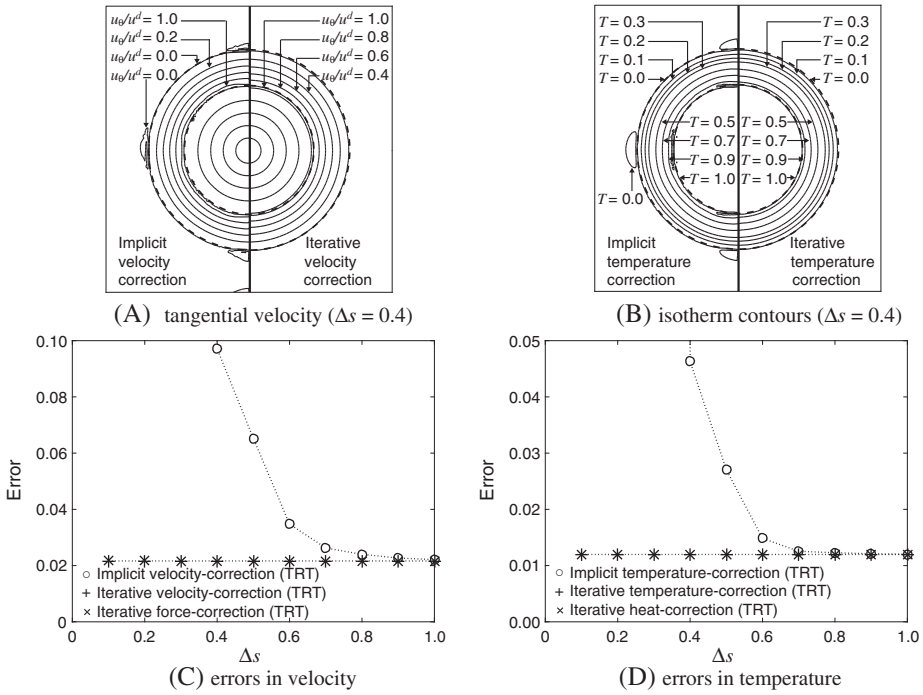


FIGURE 8 Dependence of the accuracy of the IB-TLBMs on the discrete length. A, Velocity distribution as calculated by the IB-TLBMs; B, isotherm contours; C, relative error in velocity versus discrete length; D, relative error in temperature versus discrete length. The symbols (∘), (+), and (×) denote the numerical results calculated by the TRT implicit velocity and temperature-correction method, the TRT iterative velocity and temperature-correction method, and the TRT iterative force and heat-correction method, respectively

choose the discrete length carefully to avoid numerical instability. These iterative correction methods are superior to the implicit correction method in terms of the simplicity of the algorithm and of the numerical stability.

3.2.2 | Flow past a circular cylinder at high viscosity

The fourth test is the calculation of the fluid flow past a circular cylinder at $Re = u_0 D/\nu = 10$, 20, and 40. There are numerous solutions from theoretical, experimental, and numerical approaches to compare with the present results. In our previous study,[21] the implicit correction method with the TRT collision operator yielded reasonable numerical results in fluid flows past a circular cylinder at high viscosity. Because the multi-direct forcing method cannot eliminate the velocity slip for any relaxation parameters, we verify the feasibility and applicability of the iterative velocity/force-correction methods for the numerical simulation of the fluid flows at high viscosity. The characteristic free-stream velocity u_0 is equal to $0.1c$. The diameter of the cylinder is taken to be $D = 100\delta_x$. The cylinder is located at $(16D, 20D)$ in the computational domain with $40D \times 40D$. The fluid density is given by $\rho = 1.0$. Although Peskin[1] indicates that Δs should not be larger than $0.5\delta_x$ to avoid holes in the structures, the TRT implicit correction method succeeded in preventing fluid penetration through the wall for $\Delta s \approx 2\delta_x$.[21] To demonstrate the validity of the proposed iterative correction methods, we use $\Delta s = 100\pi\delta_x/157$ for the cylinder, which makes $\Delta s \approx 2\delta_x$ for all calculations. The relaxation times are 1.25, 2.0, and 3.5 for Re = 40, 20, and 10, respectively.

Figure 9 shows the streamlines in half of the computational domain. The upper and lower figures show the numerical results of the iterative velocity-correction method and the iterative force-correction method, respectively. The thick solid line indicates the zero streamline. As shown in Figure 9A, when the SRT model is used in the IB-LBMs, the streamline is not always zero on the cylinder surface indicated by the gray line. As shown in Figure 9B, by reducing the velocity slip, the TRT iterative correction methods remove the distortion of the streamline observed in Figure 9A. As shown in Figure 9A-D, the 2 TRT iterative correction methods obtain similar distributions of the streamlines, as calculated by the implicit correction method.[21]

The drag and pressure coefficients are defined as follows:

$$C_D = \frac{F_D}{(1/2)\rho u_0^2 D}, C_p = \frac{p_s - p_0}{(1/2)\rho u_0^2}, \tag{74}$$

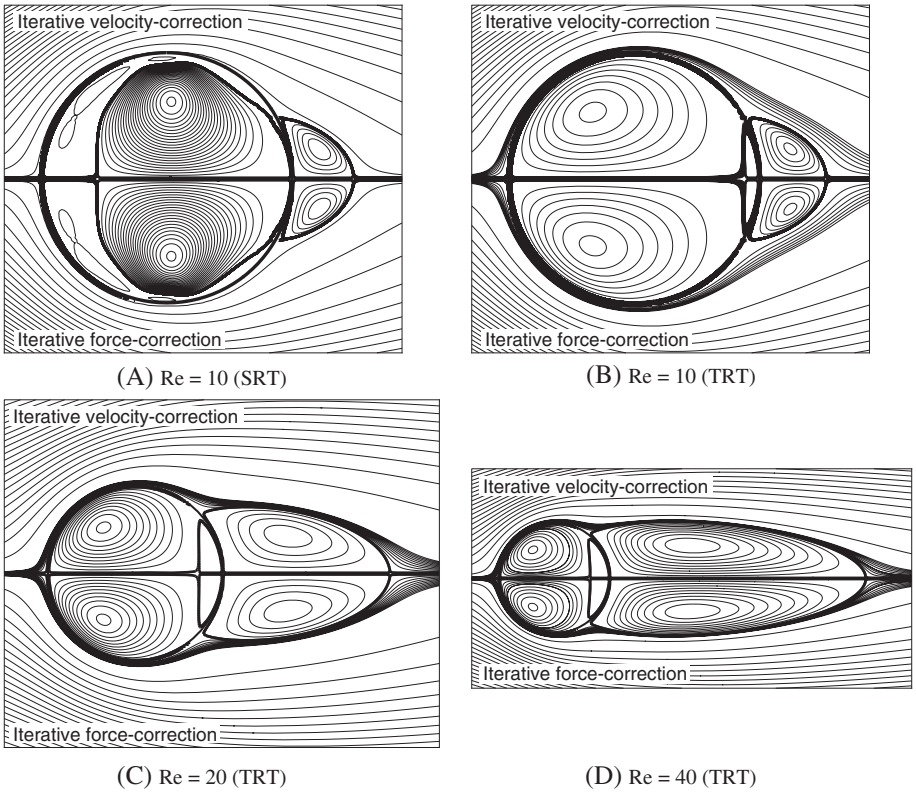


FIGURE 9 Streamlines for flow past a circular cylinder at Re = 10, 20, and 40. The thick line indicates the zero streamline. The gray line indicates the surface of the circular cylinder

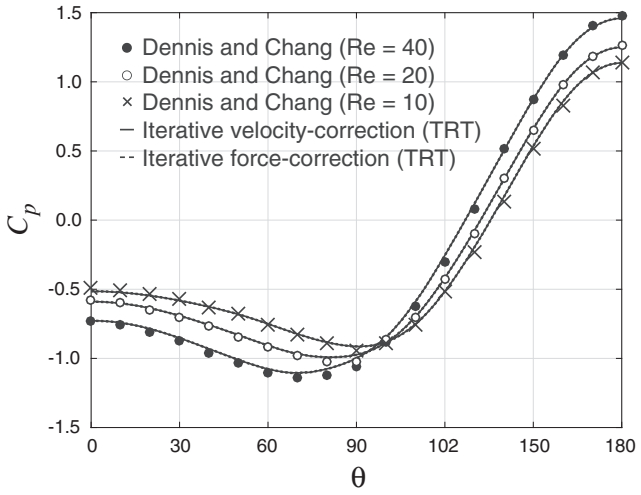


FIGURE 10 Pressure distribution on the cylinder

where F_D is the drag force calculated by

$$F_D = -\int_{\Omega} G_x dx. \tag{75}$$

Here, G_x is the x component of the force density $\boldsymbol{G}$, and p_s denotes the pressure on the surface of a circular cylinder. The pressure p_s is interpolated by the delta function given by Equation 18 at $3\delta_x$ away from the cylinder surface along its normal direction to avoid fluctuation of the pressure profiles on the cylinder surface.[35] The free stream fluid pressure p_0 is given by $p_0 = c^2\rho/3$.

Figure 10 illustrates the normalized pressure distribution on the cylinder. The solid and dotted lines represent the numerical results of the iterative velocity-correction method and the iterative force-correction method, respectively. There is good agreement between these numerical solutions and the previous studies reported by Dennis and Chang.[36] As shown in previous studies,[36-38] the pressure is always highest at the front stagnation point ($\theta = 180°$), and the location of the lowest pressure is near the steady rear stagnation point with increasing Reynolds number.

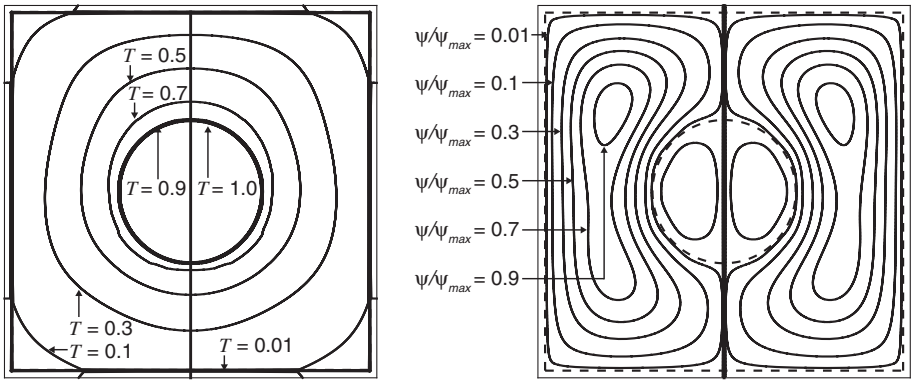
Table 1 shows the drag coefficient C_D, the wake length L normalized with respect to the radius of the cylinder, the separation angle θ_s, the pressure coefficient at the front stagnation point $C_p(\pi)$, and the pressure coefficient at the rear stagnation point $C_p(0)$ in the calculation shown in Figure 9. Dennis and Chang numerically investigated the flow past a circular cylinder using the finite difference method in the polar coordinate system.[36] Nieuwstadt and Keller used a semianalytical method with the stream function and the vorticity in a finite Fourier series.[37] He and Doolen solved the governing equations using the LBM extended to the general curvilinear coordinate system by the interpolated strategy.[38] Good agreement is observed between the results of the TRT iterative correction methods and the results of previous methods.[21,36-38] Without an evaluation of the inverse of a sparse matrix, the iterative correction methods with the TRT collision operator enforce the nonslip condition and yield reasonable streamline and pressure distributions in a high-viscosity fluid, as does the implicit velocity-correction method.

3.2.3 | Natural convection between a hot circular cylinder and a cold square enclosure

We verify the accuracy of the iterative correction methods for the simultaneous development of both velocity and temperature fields. We calculate the natural convection between a hot circular cylinder and a cold square enclosure. The inner cylinder and the square enclosure are set at temperatures $T_i = 1$ and $T_o = 0$, respectively. The IBM is applied to the surfaces of the inner cylinder and of the outer enclosure. The dashed lines indicate the position of the boundaries of the cylinder and the enclosure in Figures 11 through 13. The distance between Lagrangian nodes is approximately equal to δ_x. The length of the enclosure L is $500\delta_x$, and the number of Lagrangian points on the enclosure is 2000. The number of Lagrangian points on the inner cylinder with radius R_i is given by the integral part of $2\pi R_i$. The numbers of Lagrangian points are 314, 628, and 942 for $R_i/L = 0.1$, $R_i/L = 0.2$, and $R_i/L = 0.3$, respectively. The grid size is 525×525 with a periodic boundary. To implement the buoyancy force in the simulations, we use the Boussinesq approximation, $\boldsymbol{G} = \beta\overline{g}(T - T_m)\boldsymbol{j}$, where β is the thermal expansion coefficient, $\overline{g}$ is the acceleration due to gravity, $T_m = 0.5$ is the mean

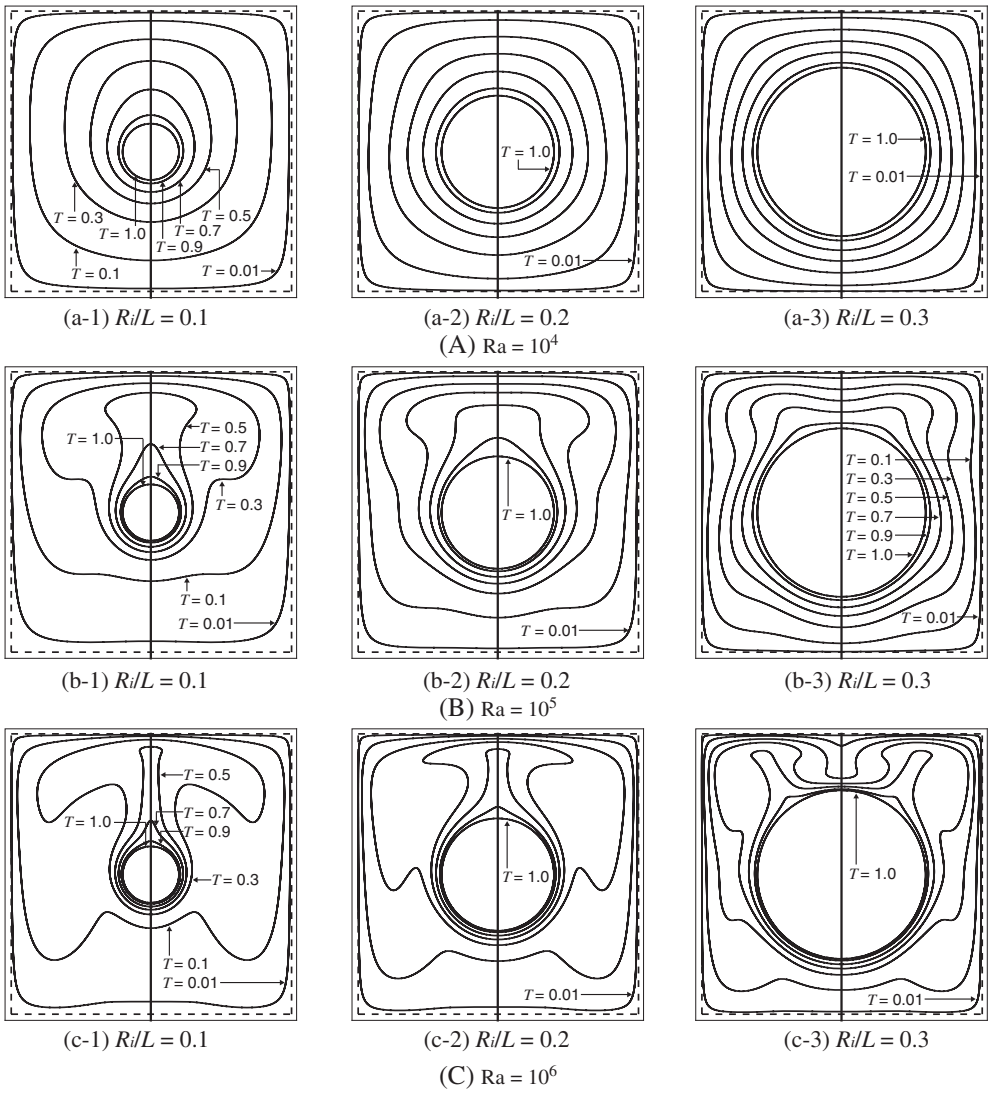
TABLE 1 Comparison of drag coefficient C_D, wake length $2L/D$, separation angle θ_s, pressure coefficient at the front stagnation point $C_p(\pi)$, and pressure coefficient at the rear stagnation point $C_p(0)$ as calculated by the proposed IB-TLBMs and in previous studies

Re	References	C_D	$2L/D$	θ_s	$C_p(\pi)$	$-C_p(0)$
10	Dennis and Chang[36]	2.846	0.53	29.6	1.489	0.742
	Nieuwstadt and Keller[37]	2.828	0.434	27.96	1.500	0.692
	He and Doolen[38]	3.170	0.474	26.89	1.393	0.687
	Implicit velocity-correction[21]	2.920	0.520	26.36	1.459	0.728
	Iterative velocity-correction	2.921	0.520	26.31	1.456	0.728
	Iterative force-correction	2.921	0.520	26.31	1.456	0.728
20	Dennis and Chang[36]	2.045	1.88	43.7	1.269	0.589
	Nieuwstadt and Keller[37]	2.053	1.786	43.37	1.274	0.582
	He and Doolen[38]	2.152	1.842	42.96	1.233	0.563
	Implicit velocity-correction[21]	2.104	1.900	40.89	1.249	0.590
	Iterative velocity-correction	2.103	1.900	40.78	1.247	0.590
	Iterative force-correction	2.103	1.900	40.78	1.248	0.590
40	Dennis and Chang[36]	1.522	4.69	53.8	1.144	0.509
	Nieuwstadt and Keller[37]	1.550	4.357	53.34	1.117	0.554
	He and Doolen[38]	1.499	4.490	52.84	1.133	0.487
	Implicit velocity-correction[21]	1.568	4.640	50.38	1.136	0.515
	Iterative velocity-correction	1.568	4.640	50.33	1.134	0.515
	Iterative force-correction	1.568	4.640	50.33	1.134	0.515



(A) isotherm contours (B) streamlines

FIGURE 11 Isotherm contours and streamlines for $R_i/L = 0.2$ at $\mathrm{Ra} = 10^4$. Pr = 0.71 and $\tau_- = 14.7414$. The results of the SRT iterative velocity and temperature-correction method, and of the SRT iterative force and heat-correction method are shown on the left- and right-hand sides, respectively



(a-1) $R_i/L = 0.1$ (a-2) $R_i/L = 0.2$ (a-3) $R_i/L = 0.3$

(A) $\mathrm{Ra} = 10^4$

(b-1) $R_i/L = 0.1$ (b-2) $R_i/L = 0.2$ (b-3) $R_i/L = 0.3$

(B) $\mathrm{Ra} = 10^5$

(c-1) $R_i/L = 0.1$ (c-2) $R_i/L = 0.2$ (c-3) $R_i/L = 0.3$

(C) $\mathrm{Ra} = 10^6$

FIGURE 12 Isotherm contours for different values of R_i/L at $\mathrm{Ra} = 10^4$, $\mathrm{Ra} = 10^5$, and $\mathrm{Ra} = 10^6$. Relaxation times are given by $\tau_-^c = 14.7414$, $\tau_-^c = 5.0035$, and $\tau_-^c = 1.9241$ for $\mathrm{Ra} = 10^4$, $\mathrm{Ra} = 10^5$, and $\mathrm{Ra} = 10^6$, respectively. The results of the TRT iterative velocity and temperature-correction method, and of the TRT iterative force and heat-correction method are shown on the left- and right-hand sides, respectively

temperature, and $\boldsymbol{j}$ denotes the direction opposite to the force of gravity. The buoyancy force is added to the fluid between the inner cylinder and the outer enclosure. The Prandtl number is set to 0.71. The Rayleigh number Ra is set to 10^4, 10^5, and 10^6. The Rayleigh number and Prandtl number are defined as $\mathrm{Ra} = \beta\bar{g}\Delta TL^3/\nu\chi$ and $\mathrm{Pr} = \nu/\chi$, respectively. Using the above definitions of the Rayleigh number and the Prandtl number, the definition of the viscosity and the definition of the thermal diffusivity given by Equation 14, we set the relaxation times τ_+^v and τ^c_- as follows:

$$\tau_+^v = \frac{3\sqrt{\beta\bar{g}\Delta TL^3}\sqrt{\mathrm{Pr}}}{c^2\delta_x\sqrt{\mathrm{Ra}}} + 0.5, \quad \tau^c_- = \frac{3\sqrt{\beta\bar{g}\Delta TL^3}}{c^2\delta_x\sqrt{\mathrm{Pr}\,Ra}} + 0.5, \tag{76}$$

where $\Delta T = T_i - T_o = 1$ is the temperature difference between the hot and cold boundaries. Substituting $\beta\bar{g} = 0.00128$ and $L = 500$ into Equation 76, we obtain the relaxation times $\tau_+^v = 10.611$ and $\tau^c_- = 14.7414$ for $\mathrm{Ra} = 10^4$, $\tau_+^v = 3.6975$ and $\tau^c_- = 5.0035$ for $\mathrm{Ra} = 10^5$, and $\tau_+^v = 1.5111$ and $\tau^c_- = 1.9241$ for $\mathrm{Ra} = 10^6$. We use $\Lambda = \frac{9}{8}$ for the TRT collision model.

The following equations are used to judge the convergence of the numerical solution:

$$max||u(t+\delta_t)| - |u(t)|| \leq 10^{-8}, \tag{77}$$

$$max|T(t+\delta_t) - T(t)| \leq 10^{-8}. \tag{78}$$

Figures 11 through 13 show the streamlines and isotherm contours for the different ratios R_i/L and Ra. The left-hand side and the right-hand side of the figures show the numerical results of the iterative velocity and temperature-correction method and the iterative force and heat-correction method, respectively. In Figures 11 through 13, the contour lines

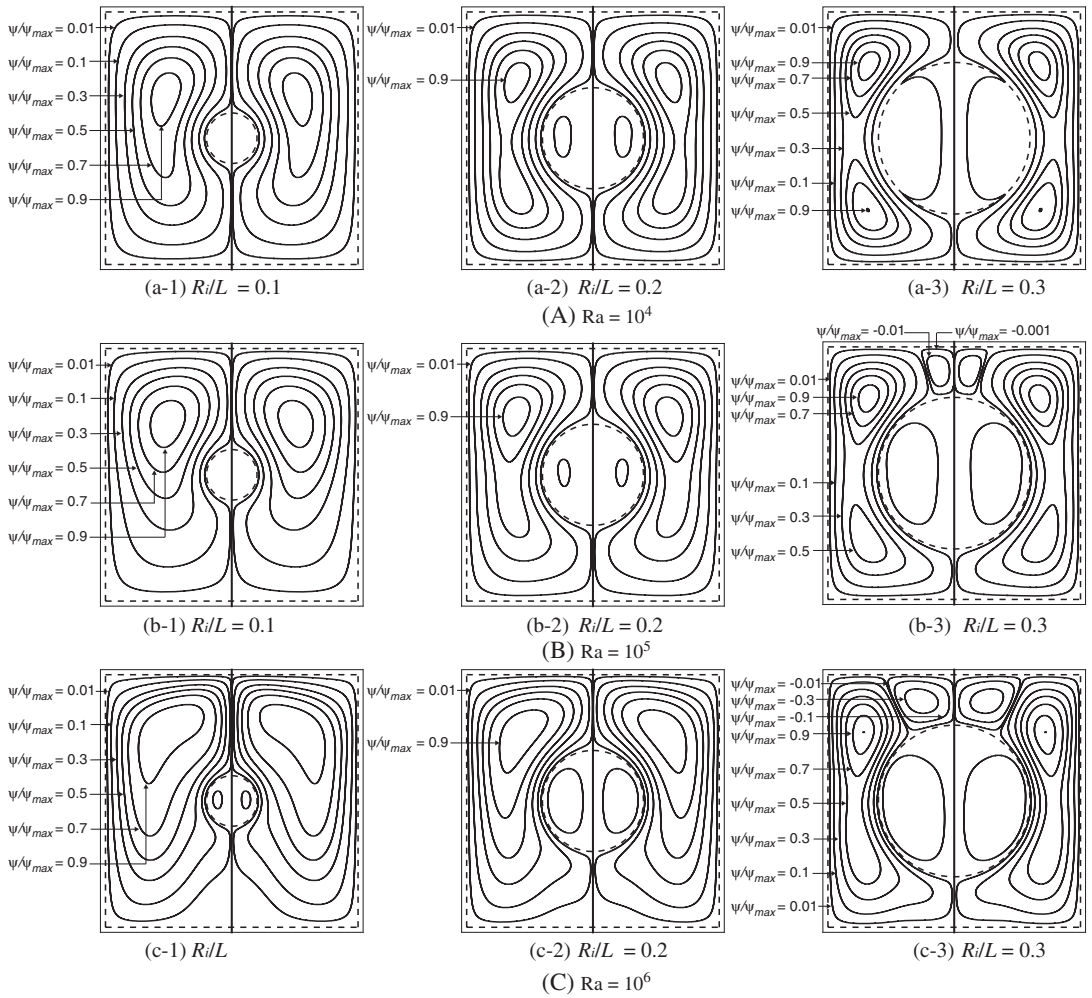


FIGURE 13 Streamlines for different values of R_i/L at $\mathrm{Ra} = 10^4$, $\mathrm{Ra} = 10^5$, and $\mathrm{Ra} = 10^6$. The results of the TRT iterative velocity and temperature-correction method, and of the TRT iterative force and heat-correction method are shown on the left- and right-hand sides, respectively

for the streamlines normalized by the maximum values ψ_{max} are set to 0.01, 0.1, 0.3, 0.5, 0.7, and 0.9. The isotherm contours are set to 0.01, 0.1, 0.3, 0.5, 0.7, 0.9, and 1.0 for comparison with the results obtained using the differential quadrature method[39] and those obtained by a control volume-based numerical procedure using the semi-implicit method for pressure-linked equations (SIMPLE) algorithm.[40] Although the SRT collision model causes the temperature to be united at the inner cylinder, the temperature around the cylinder is distorted because of the temperature jump as shown in Figure 11. Figures 12 and 13 reveal that the isotherms and streamlines obtained by the TRT collision model are in good agreement with the results reported in the previous studies.[39,40]

We verify the average Nusselt number on the hot cylinder $\bar{\mathrm{N}}\mathrm{u}$ and the maximum absolute values of the stream function ψ_{max}. The Nusselt number Nu is defined as follows:

$$\mathrm{Nu} = -\frac{\partial T}{\partial n}\bigg|_{wall}, \bar{\mathrm{N}}\mathrm{u} = \frac{1}{W}\int_0^W Nuds, \tag{79}$$

where $n = (n_x, n_y)$ is an outward unit vector normal to the cylinder surface, and W is the circumferential length of the inner cylinder surface. We use the second-order approximation $\partial T/\partial x \approx (T_{i+1,j} - T_{i-1,j})/2\delta_x$ to calculate the gradient at the Eulerian nodes near the cylinder. At the Lagrangian nodes, the derivatives in the direction of the x-axis and the y-axis are obtained by interpolation of the gradient at the Eulerian nodes using the delta function shown in Equations 18 and 19. The directional derivative in the direction of the normal vector is given by the inner product, $\frac{\partial T}{\partial n} = n_x\frac{\partial T}{\partial x} + n_y\frac{\partial T}{\partial y}$. The definition, $\psi = -\int vdx + \int udy$, is used to calculate the stream function ψ. Before calculating the stream function, we normalize the velocity by the reference velocity χ/H.[24] The numerical integration is performed using Simpson's rule. Table 2 shows the average Nusselt number and maximum absolute values of the stream function. Good agreement between the results of the TRT correction methods and the results reported in the previous studies is observed for all Rayleigh numbers and all R_i/L ratios.

We measure the computational time of the proposed IB-TLBM using the double-population method. To obtain the convergent solution with $R_i/L = 0.3$, $N_m = 20$, and at $\mathrm{Ra} = 10^4$, the TRT implicit correction method requires 3429 [s] and 9851 iterations, and the TRT iterative velocity and temperature-correction method requires 3522 [s] and 9851 iterations. The calculations are performed on a 3.1-GHz Linux workstation. Because the size of the coefficient matrix is N^2, the computational time of the implicit correction method to obtain the body force $\boldsymbol{G}$ is expected to be proportional to N^2. In the iterative correction method, the computational time increases with the order of $O(N)$. There is no significant difference in computational time between the iterative method and the implicit method for 942 Lagrangian points on the inner cylinder, ie, $\Delta s \approx \delta_x$. This result demonstrates that the computational cost of the IBM procedure constitutes a small amount of all computational processes in the IB-TLBM. When we increase the number of Lagrangian points on the inner cylinder to 1884 ($\Delta s \approx 0.5\delta_x$), the implicit correction method requires 16 128 [s] and 42 146 iterations owing to the slight oscillation near the convergence state. In contrast, the iterative correction method requires 3723 [s] and 9851 iterations. The Nusselt numbers of the implicit correction method and the iterative correction method are 4.312 and 5.384, respectively. In the same manner as for the numerical results on the influence of the discrete length on the numerical stability shown in Figure 8, it is necessary to consider the distance between Lagrangian points carefully to obtain an adequate Nusselt number in the implicit correction method. As shown in Equation 35b, because the elements of matrix A are based on the nearby interaction between 2 Lagrangian points in the square area $2\delta_x \times 2\delta_x$, the coefficient

TABLE 2 Surface-averaged Nusselt number and maximum absolute value of stream function

	Ra	**10^4**			**10^5**			10^6		
	R_i/L	**0.1**	**0.2**	**0.3**	**0.1**	**0.2**	**0.3**	**0.1**	**0.2**	**0.3**
Shu and Zhu[39]	$\bar{N}u$	2.08	3.24	5.40	3.79	4.86	6.21	6.11	8.90	12.00
	ψ_{max}	1.71	0.97	0.49	9.93	8.10	5.10	20.98	24.13	20.46
Moukalled and Acharya[40]	$\bar{N}u$	2.071	3.331	5.826	3.825	5.08	6.212	6.107	9.374	11.62
	ψ_{max}	1.73	1.02	0.50	10.15	8.38	5.10	25.35	24.07	21.30
Implicit velocity and temperature-correction	$\bar{N}u$	2.072	3.233	5.385	3.786	4.932	6.218	6.081	8.902	11.99
	ψ_{max}	1.775	1.004	0.499	10.22	8.391	5.073	21.34	24.93	20.82
Iterative velocity and temperature-correction	$\bar{N}u$	2.073	3.230	5.384	3.786	4.928	6.218	6.083	8.895	12.00
	ψ_{max}	1.775	1.004	0.499	10.22	8.391	5.073	21.34	24.93	20.82
Iterative force and heat-correction	$\bar{N}u$	2.073	3.230	5.384	3.786	4.928	6.218	6.083	8.896	12.00
	ψ_{max}	1.772	1.004	0.499	10.22	8.391	5.073	21.34	24.93	20.82

matrix A is sparse in the implicit correction method. We must implement a high-performance solution procedure and optimization for the zero and nonzero elements in the sparse matrix.[41] The advantage of the proposed method over the implicit correction method is its robustness against the number of Lagrangian points and the simplicity of the algorithm without the cost of the sparse matrix computation.

3.2.4 | Multiblock method

Yu et al succeeded in obtaining numerical solutions on a high-resolution grid in the LBM framework by using local grid refinement near a wall.[42] In applying the multiblock approach in the IB-LBM, Peng et al used the MRT collision scheme for better numerical stability.[43] We apply the multiblock method to the proposed IB-TLBM in the calculation of the cylindrical Couette flow. To maintain a constant viscosity and a constant thermal diffusivity throughout the flow field with different grid sizes, the relation between relaxation times on the fine block and relaxation times on the coarse block must obey the following equations:

$$\tau_+^{\nu(f)} = \frac{1}{2} + m\left(\tau_+^{\nu(c)} - \frac{1}{2}\right), \tau_-^{c(f)} = \frac{1}{2} + m\left(\tau_-^{c(c)} - \frac{1}{2}\right), \tag{80}$$

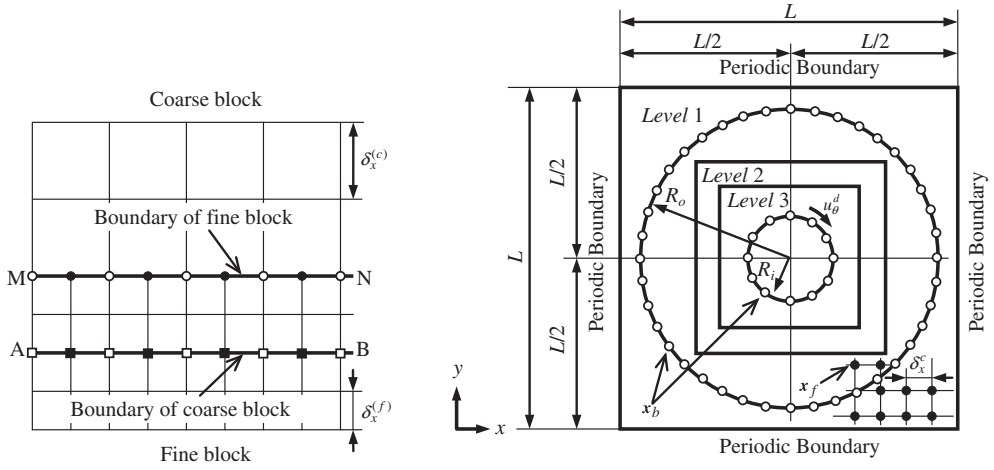
where m is the ratio of the lattice space between the 2-grid systems, defined by $m = \delta_x^{(c)}/\delta_x^{(f)}$. The superscripts (f) and (c) denote the fine block and the coarse block, respectively. We verify the effect of the TRT collision operator on the accuracy of the numerical results calculated by the IB-TLBM using the multiblock scheme, in which the relaxation parameter increases with decreasing grid size following Equation 80.

The typical interface structure is shown in Figure 14A. The lines AB and MN are the boundaries of the coarse block and the fine block, respectively. For the information exchange from the fine block to the coarse block at the nodes denoted by the open square symbol "□" on line AB in Figure 14A, Equations 25a and 29a give the postcollision distribution functions $\tilde{f}_k^{(c)}$ and $\tilde{g}_k^{(c)}$ as follows:

$$\begin{aligned}\tilde{f}_k^{(c)}(\boldsymbol{x},t) = f_k^{(eq,f)}(\boldsymbol{x},t) &+ \frac{\tau_+^{\nu(f)}-1}{m(\tau_+^{\nu(c)}-1)}[f_k^{+(f)}(\boldsymbol{x},t) - f_k^{+(eq,f)}(\boldsymbol{x},t)] \\ &+ \frac{\tau_-^{\nu(f)}-1}{m(\tau_-^{\nu(c)}-1)}[f_k^{-(f)}(\boldsymbol{x},t) - f_k^{-(eq,f)}(\boldsymbol{x},t)],\end{aligned} \tag{81a}$$

$$\begin{aligned}\tilde{g}_k^{(c)}(\boldsymbol{x},t) = g_k^{(eq,f)}(\boldsymbol{x},t) &+ \frac{\tau_+^{\nu(f)}-1}{m(\tau_+^{\nu(c)}-1)}[g_k^{+(f)}(\boldsymbol{x},t) - g_k^{+(eq,f)}(\boldsymbol{x},t)] \\ &+ \frac{\tau_-^{\nu(f)}-1}{m(\tau_-^{\nu(c)}-1)}[g_k^{-(f)}(\boldsymbol{x},t) - g_k^{-(eq,f)}(\boldsymbol{x},t)],\end{aligned} \tag{81b}$$

for the TRT collision model.



(a) Interface structure between two blocks (b) Multi-block arrangement

FIGURE 14 Schematic diagram of multiblock arrangement for cylindrical Couette flow

At the nodes denoted by the open circle symbol "∘" on the line NM in Figure 14A, the data are transferred from the coarse block to the fine block by the following equations:

$$\tilde{f}_k^{(f)}(\boldsymbol{x},t) = f_k^{(eq,c)}(\boldsymbol{x},t) + m\frac{\tau_+^{\nu(f)}-1}{\tau_+^{\nu(c)}-1}[f_k^{+(c)}(\boldsymbol{x},t) - f_k^{+(eq,c)}(\boldsymbol{x},t)] + m\frac{\tau_-^{\nu(f)}-1}{\tau_-^{\nu(c)}-1}[f_k^{-(c)}(\boldsymbol{x},t) - f_k^{-(eq,c)}(\boldsymbol{x},t)]. \tag{82a}$$

$$\tilde{g}_k^{(f)}(\boldsymbol{x},t) = g_k^{(eq,c)}(\boldsymbol{x},t) + m\frac{\tau_+^{\nu(f)}-1}{\tau_+^{\nu(c)}-1}[g_k^{+(c)}(\boldsymbol{x},t) - g_k^{+(eq,c)}(\boldsymbol{x},t)] + m\frac{\tau_-^{\nu(f)}-1}{\tau_-^{\nu(c)}-1}[g_k^{-(c)}(\boldsymbol{x},t) - g_k^{-(eq,c)}(\boldsymbol{x},t)]. \tag{82b}$$

On the nodes denoted by the solid circle symbol "•" in Figure 14A, spatial interpolation is required to obtain the corresponding information $\tilde{f}_k^{(f)}$ and $\tilde{g}_k^{(f)}$. We use a symmetric cubic spline fitting for the spatial interpolation of the distribution functions, as in Yu et al.[42]

When the ratio of the lattice space m equals 2, the kinetic Equations 25a, 25b, 29a, and 29b are computed twice on the fine block within one time step δ_t. Because the magnitude of the discrete velocity, c, remains constant at any grid sizes, the distribution functions $f_k^{(f)}$ and $g_k^{(f)}$ stream to the neighboring node within half the time step, $\delta_t/2$, in the fine grid where $\delta_x^{(f)} = \delta_x^{(c)}/2$. Following Yu et al,[42] the 3-point Lagrange interpolation is used to calculate $\tilde{f}_k^{(f)}(\boldsymbol{x}, t+\delta_t/2)$ and $\tilde{g}_k^{(f)}(\boldsymbol{x}, t+\delta_t/2)$ on all nodes denoted by the square symbols "□" and "■" on the line AB in Figure 14A.

Figure 14B shows a schematic diagram of the multiblock arrangement in the numerical calculation of the cylindrical Couette flows. We set 3 levels of blocks. The block around the static outer cylinder belongs to the first level with the coarsest grid $\delta_x^{(c)}$. We use the finest grid in the third level around the rotating inner ring. The ratio of mesh spacing between adjacent levels is 2. The diameter of the outer cylinder $D_o = 100\delta_x^{(c)}$. The inner cylinder has a diameter D_i of $50\delta_x^{(c)}$. The tangential velocity of the inner cylinder u_θ^d is $0.01c$. The number of grid points in the first, second, and third levels is 111×111, 125×125, and 225×225, respectively. Because the coefficient matrix depends on the discrete length for each Lagrangian point Δs and on the Eulerian mesh size δ_x, the implicit correction method easily suffers from numerical instability in the multiblock method with various grid sizes, δ_x. We apply the TRT iterative velocity and temperature-correction method to the multiblock structured grids.

Figure 15A and 15B shows the contour lines of the tangential velocity and of the temperature with $\tau_+^{\nu(1)} = \tau_-^{c(1)} = 2.0$, respectively. The superscripts (1), (2), and (3) denote the grid level shown in Figure 14B. The left- and right-hand figures correspond to the numerical results with the SRT collision operator and with the TRT collision operator, respectively. When $\tau_+^{\nu(1)} = \tau_-^{c(1)} = 2.0$, the SRT model induces the velocity slip and temperature jump around the rings as shown in Figure 15A and 15B. By substituting $\tau_+^{\nu(1)} = \tau_-^{c(1)} = 2.0$ into Equation 80, one obtains the relaxation parameters $\tau_+^{\nu(2)} = \tau_-^{c(2)} = 3.5$ and $\tau_+^{\nu(3)} = \tau_-^{c(3)} = 6.5$ for the second and third level, respectively. Figure 15A and 15D shows the profiles of the tangential velocity and the temperature, respectively, at the horizontal centerline of the computational domain. In Figure 15C and 15D, the triangle symbol "△" indicates the tangential velocity and temperature profiles calculated by the IB-TLBM with the SRT collision operator, which does not use the multiblock approach. As shown in Figure 15C and 15D, the IB-TLBM without the multiblock method gives a reasonable numerical result for $\tau_+^{\nu} = \tau_-^{c} = 2.0$. Substituting $\tau_+^{\nu(1)} = \tau_-^{c(1)} = 1.1$ into Equation 80 yields $\tau_+^{\nu(2)} = \tau_-^{c(2)} = 1.7$ and $\tau_+^{\nu(3)} = \tau_-^{c(3)} = 2.9$. Although the relaxation times $\tau_+^{\nu(3)} = \tau_-^{c(3)} = 2.9$ around the inner cylinder, no velocity and temperature distortions are observed in Figures 15C and 15D for $\tau_+^{\nu(1)} = \tau_-^{c(1)} = 1.1$. Figure 15 clarifies that the TRT collision operator enables the numerical calculation of the IB-TLBM with high resolution in the multiblock method by eliminating the boundary slip for a long relaxation time.

Figure 16 shows the relative errors defined by Equations 72a and 72b versus the number of grid points along the centerline of the outer ring. The TRT iterative velocity and temperature-correction method with $\tau_+^{\nu(1)} = \tau_-^{c(1)} = 2.0$ shows the same accuracy as with $\tau_+^{\nu(1)} = \tau_-^{c(1)} = 1.1$. These numerical results demonstrate that the iterative velocity and temperature-correction method with the TRT collision operator is suitable for computation in the scheme of the IB-TLBMs combined with the multiblock method.

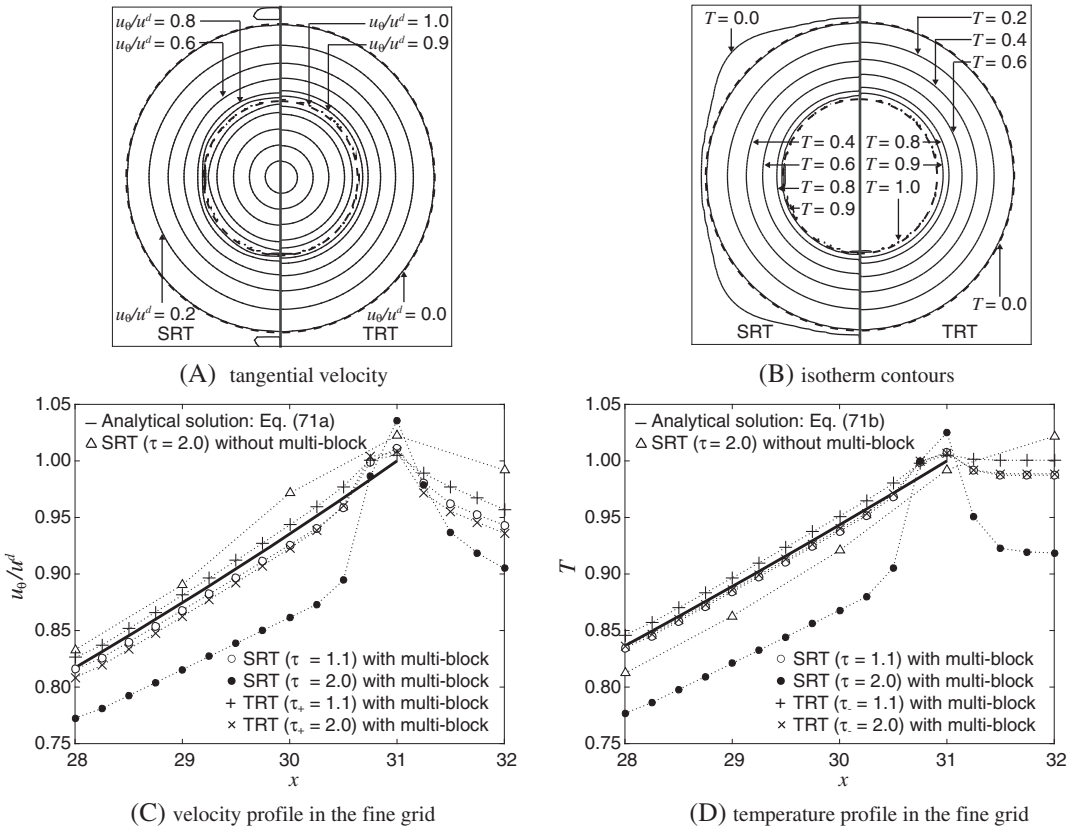


FIGURE 15 Tangential velocity profiles and temperature profiles in calculations of cylindrical Couette flows in the multiblock scheme

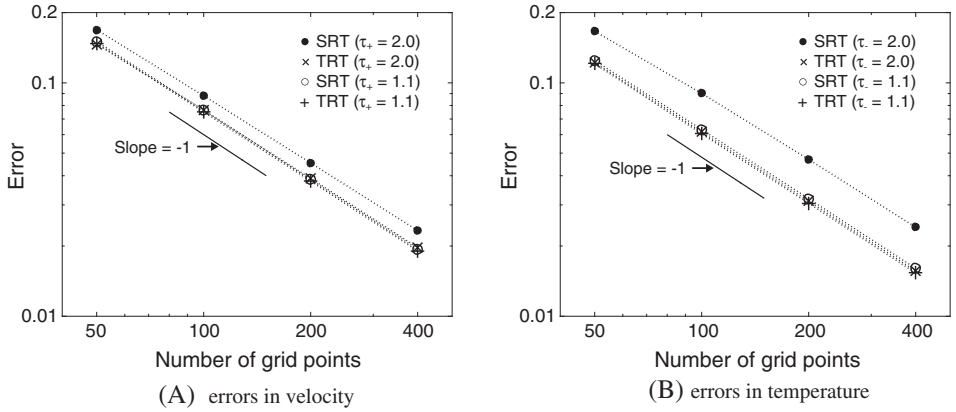


FIGURE 16 Relative error versus number of grid points in the calculation of cylindrical Couette flows in the multiblock scheme

4 | CONCLUSION

We have proposed 2 types of the iterative correction methods, and we have theoretically and numerically demonstrated the boundary slip of the IB-TLBM based on the double-population model. One iterative correction method is based on Guo's forcing term, and the other is based on Cheng's split forcing approach. To verify the effectiveness of the combination of the iterative method with the TRT collision operator on the reduction of the velocity slip and the temperature jump, we derived the analytical solutions of the IB-TLBMs in the symmetric shear flows and in the heat transfer between 2 porous plates. The analytical solutions demonstrated that the commonly used multi-direct forcing method was unable to reduce the velocity slip for any relaxation time. The proposed iterative correction methods combined with the TRT collision operator eliminated the velocity slip, the temperature jump, and the boundary-value deviation even at high relaxation times. In the numerical simulations of the cylindrical Couette flows and the flows past a circular cylinder, the TRT iterative

methods showed approximately the same accuracy as the TRT implicit correction method in terms of the velocity and temperature distributions. The pressure oscillations were observed around the boundary only in the numerical results calculated by the TRT implicit correction method. The numerical results for the cylindrical Couette flows demonstrated that the proposed iterative correction methods were more numerically stable than the implicit correction method.

The drag coefficient, wake length, separation angle, and pressure coefficient in the numerical simulation of the flow past a circular cylinder using 2 types of TRT iterative correction methods show good agreement with the results of previous methods.[21,36-38]

We applied the proposed methods to the simulation of the natural convection between a hot circular cylinder and a cold square enclosure. The proposed methods adequately set the boundary values and provided reasonable average Nusselt numbers and maximum absolute values of the stream function. The choice of $\Lambda = \frac{9}{8}$ successfully eliminated both the velocity slip and the temperature jump and obtained the most accurate results in the numerical calculations in this study. Although the relaxation time on the fine block in the multiblock framework was high, the TRT iterative correction method gave a fine-resolution numerical solution without velocity or temperature distortion around the boundaries.

By iteratively reducing the boundary-value deviation, the multi-direct forcing method and the proposed iterative correction methods satisfied the nonslip boundary condition. The macroscopic equations corresponding to Equations 5 and 21b had additional terms including the temporal and spatial change of the body force.[14] To obtain the correct Navier-Stokes equations, the velocity definition and the forcing term should be given by Equations 26 and 27. Since the multi-direct forcing method used the conventional forcing term given by Equation 21b, the body force did not adequately set the boundary condition without boundary slip in the IB-TLBM based on the TRT collision operator.

The major achievements of this study are the following: (1) The analytical solutions for the velocity slip, temperature jump, and boundary-value deviation in the IB-TLBM are obtained; (2) the multi-direct forcing method and iterative correction methods can reduce the boundary-value deviation; (3) the TRT collision operator is effective in reducing the temperature jump as well as the velocity slip; (4) to eliminate the boundary slip, the IB-TLBMs should calculate the force and heat from a solid structure to a fluid using the forcing approach proposed by Guo or Cheng[14,15]; (5) the reduction of the boundary slip improves the accuracy of the IB-TLBM for a long relaxation time.

For the rapid computation of linear systems of equations, we should efficiently parallelize and optimize the coefficient matrix in the implicit correction method. In the fluid flow in complex geometry in the multiblock approach, optimally displaying both the zero and nonzero elements in a coefficient matrix using the implicit correction method is a cumbersome procedure. Because the iterative correction methods are robust against the arrangement of Lagrangian points and the Eulerian mesh size, the proposed method is suitable for applications with complex structures such as the aggregation of blood platelets. We demonstrated the high accuracy and the computational efficiency of the proposed iterative methods using the TRT collision operator.

The proposed IB-TLBMs calculated the incompressible thermal flows around a rigid wall with the Dirichlet boundary condition. In future work, we will develop an implementation of Neumann type boundary conditions for a flexible fiber in compressible fluid flows in the framework of the proposed IB-TLBMs to examine the velocity, pressure, and temperature distributions in an inflated balloon under a heat flux condition.[11,44]

ACKNOWLEDGMENTS

This study was supported by JSPS KAKENHI grant JP16K06070. The authors would like to thank Prof R. Takahashi, Prof E. Takegosi, Prof K. Okui, and Prof Y. Abe for their helpful suggestions.

ORCID


Takashi Seta http://orcid.org/0000-0003-3183-5564

REFERENCES

1. Peskin CS. Flow patterns around heart valves: a numerical method. *J Comput Phys*. 1972;10(2):252-271.
2. Chen S, Doolen GD. Lattice Boltzmann method for fluid flows. *Ann Rev Fluid Mechan*. 1988;30(1):329-362.
3. Feng ZG, Michaelides EE. The immersed boundary-lattice Boltzmann method for solving fluid-particles interaction problems. *J Comput Phys*. 2004;195(2):602-628.
4. Feng ZG, Michaelides EE. Proteus: a direct forcing method in the simulations of particulate flows. *J Comput Phys*. 2005;202(1):20-51.

5. Niu XD, Shu C, Chew YT, Peng Y. A momentum exchange-based immersed boundary-lattice Boltzmann method for simulating incompressible viscous flows. *Phys Lett A*. 2006;354(3):173-182.
6. Wu J, Shu C. Implicit velocity correction-based immersed boundary-lattice Boltzmann method and its applications. *J Comput Phys*. 2009;228(6):1963-1979.
7. Suzuki K, Inamuro T. Effect of internal mass in the simulation of a moving body by the immersed boundary method. *Comput Fluids*. 2011;49(1):173-187.
8. Kang SK, Hassan YA. A comparative study of direct-forcing immersed boundary-lattice Boltzmann methods for stationary complex boundaries. *Int J Numer Meth Fluids*. 2011;66(9):1132-1158.
9. Zhang C, Cheng Y, Zhu L, Wu J. Accuracy improvement of the immersed boundary-lattice Boltzmann coupling scheme by iterative force correction. *Comput Fluids*. 2016;124(2):246-260.
10. Jeong HK, Yoon HS, Ha MY, Tsutahara M. An immersed boundary-thermal lattice Boltzmann method using an equilibrium internal energy density approach for the simulation of flows with heat transfer. *J Comput Phys*. 2010;229(7):2526-2543.
11. Cheng Y, Zhu L, Zhang C. Numerical study of stability and accuracy of the immersed boundary method coupled to the lattice Boltzmann BGK model. *Commun Comput Phys*. 2014;16(1):136-168.
12. Ladd AJC. Numerical simulations of particulate suspensions via a discretized Boltzmann equation. Part 1. Theoretical foundation. *J Fluid Mechan*. 1994;271:285-309.
13. Wang Z, Fan J, Luo K, Cen K. Immersed boundary method for the simulation of flows with heat transfer. *Int J Heat and Mass Trans*. 2009;52(19-20):4510-4518.
14. Guo Z, Zheng C, Shi B. Discrete lattice effects on the forcing term in the lattice Boltzmann method. *Phys Rev E*. 2002;65:046308 (6 pages).
15. Cheng Y, Li J. Introducing unsteady non-uniform source terms into the lattice Boltzmann model. *Int J Numer Meth Fluids*. 2008;56(6):629-641.
16. Le G, Zhang J. Boundary slip from the immersed boundary lattice Boltzmann models. *Phys Rev E*. 2009;79:026701 (8pages).
17. Lallemand P, Luo LS. Theory of the lattice Boltzmann method: dispersion, dissipation, isotropy, Galilean invariance, and stability.*Phys Rev E*. 2000;61:6546-6562.
18. Lu J, Han H, Shi B, Guo Z. Immersed boundary lattice Boltzmann model based on multiple relaxation times. *Phys Rev E*. 2012;85:016711 (12 pages).
19. Ginzburg I, d′Humières D, Kuzmin A. Optimal stability of advection-diffusion lattice Boltzmann models with two relaxation times for positive/negative equilibrium. *J Stat Phys*. 2010;139(6):1090-1143.
20. Hayashi K, Rojas R, Seta T, Tomiyama A. Immersed boundary-lattice Boltzmann method using two relaxation times. *J Comput Multiphase Flows*. 2012;4(2):193-209.
21. Seta T, Rojas R, Hayashi K, Tomiyama A. Implicit-correction-based immersed boundary-lattice Boltzmann method with two relaxation times. *Phys Rev E*. 2014;89:023307 (22 pages).
22. He X, Chen S, Doolen GD. A novel thermal model for the lattice Boltzmann method in incompressible limit. *J Comput Phys*. 1998;146(1):282-300.
23. Shi Y, Zhao TS, Guo ZL. Thermal lattice Bhatnagar-Gross-Krook model for flows with viscous heat dissipation in the incompressible limit. *Phys Rev E*. 2004;70:066310 (10 pages).
24. Peng Y, Shu C, Chew YT. Simplified thermal lattice Boltzmann model for incompressible thermal flows. *Phys Rev E*. 2003;68:026701 (8 pages).
25. Goodarzi M, Safaei MR, Karimipour A, et al. Comparison of the finite volume and lattice Boltzmann methods for solving natural convection heat transfer problems inside cavities and enclosures. *Abstract and Applied Analysis*. 2014:762184.
26. Karimipour A, Nezhad AH, D′Orazio A, Shirani E. Investigation of the gravity effects on the mixed convection heat transfer in a microchannel using lattice Boltzmann method. *Int J Thermal Sci*. 2012;54:142-152.
27. Karimipour A, Esfe MH, Safaei MR, et al. Mixed convection of copper-water nanofluid in a shallow inclined lid driven cavity using the lattice Boltzmann method. *Physica A*. 2014;402:150-168.
28. Shamshirband S, Malvandi A, Karimipour A, et al. Performance investigation of micro- and nano-sized particle erosion in a 90° elbow using an ANFIS model. *Powder Technol*. 2015;284:336-343.
29. Goodarzi M, Safaei MR, Vafai K, et al. Investigation of nanofluid mixed convection in a shallow cavity using a two-phase mixture model. *Int J Thermal Sci*. 2014;75:204-220.
30. Karimipour A, Nezhad AH, D′Orazio A, Esfe MH, Safaei MR, Shirani E. Simulation of copper-water nanofluid in a microchannel in slip flow regime using the lattice Boltzmann method. *European J Mechanics - B/Fluids*. 2015;49:89-99.
31. Karimipour A. New correlation for Nusselt number of nanofluid with Ag / Al2O3 / Cu nanoparticles in a microchannel considering slip velocity and temperature jump by using lattice Boltzmann method. *Int J Thermal Sci*. 2015;91:146-156.
32. Seta T. Implicit temperature correction-based immersed boundary-thermal lattice Boltzmann method for the simulation of natural convection. *Phys Rev E*. 2013;87:063304 (16 pages).
33. Sterling JD, Chen S. Stability analysis of lattice Boltzmann methods. *J Comput Phys*. 1996;123(1):196-206.
34. Ginzburg I. Consistent lattice Boltzmann schemes for the Brinkman model of porous flow and infinite Chapman-Enskog expansion. *Phys Rev E*. 2008;77:066704 (12 pages).
35. Shu C, Liu N, Chew YT. A novel immersed boundary velocity correction-lattice Boltzmann method and its application to simulate flow past a circular cylinder. *J Comput Phys*. 2007;226(2):1607-1622.

36. Dennis SCR, Chang GZ. Numerical solutions for steady flow past a circular cylinder at Reynolds numbers up to 100. *J Fluid Mechanics*. 1970;42(3):471-489.
37. Nieuwstadt F, Keller HB. Viscous flow past circular cylinders. *Comput Fluids*. 1973;1(1):59-71.
38. He X, Doolen GD. Lattice Boltzmann method on curvilinear coordinates system: flow around a circular cylinder. *J Comput Phys*. 1997;134(2):306-315.
39. Shu C, Zhu YD. Efficient computation of natural convection in a concentric annulus between an outer square cylinder and an inner circular cylinder. *Int J Numer Meth Fluids*. 2002;38(5):429-445.
40. Moukalled F, Acharya S. Natural convection in the annulus between concentric horizontal circular and square cylinders. *J Thermophysics and Heat Transfer*. 1996;10(3):524-531.
41. Bunch JR, Rose DJ. *Sparse Matrix Computations*. New York, San Francisco, London: Academic Press; 2014.
42. Yu D, Mei R, Shyy W. A multi-block lattice Boltzmann method for viscous fluid flows. *Int J Numer Meth Fluids*. 2002;39(2):99-120.
43. Peng Y, Shu C, Chew YT, Niu XD, Lu XY. Application of multi-block approach in the immersed boundary-lattice Boltzmann method for viscous fluid flows. *J Comput Phys*. 2006;218(2):460-478.
44. Watari M, Tsutahara M. Possibility of constructing a multispeed Bhatnagar-Gross-Krook thermal model of the lattice Boltzmann method. *Phys Rev E*. 2004;70:016703 (9 pages).
45. He X, Zou Q, Luo LS, Dembo M. Analytic solutions of simple flows and analysis of nonslip boundary conditions for the lattice Boltzmann BGK model. *J Stat Phys*. 1997;87(1-2):115-136.

How to cite this article: Seta T, Hayashi K, Tomiyama A. Analytical and numerical studies of the boundary slip in the immersed boundary-thermal lattice Boltzmann method. *Int J Numer Meth Fluids*. 2018;86:454–490. https://doi.org/10.1002/fld.4462

APPENDIX A: ANALYTICAL SOLUTIONS OF THE MULTI-DIRECT FORCING METHOD FOR SYMMETRIC SHEAR FLOWS

To obtain the analytical solution of the velocity slip in the the multi-direct forcing method with 2 relaxation times, we follow the procedure of Le et al[16] and He et al.[45] Because $f_k(\boldsymbol{x}+\boldsymbol{e}_k\delta_t, t+\delta_t) = f_k(\boldsymbol{x}, t)$ for $k = 1$ and 3 in a steady state, Equations 21a, 21b, and 47 yield

$$0 = -\frac{\tau_+^v+\tau_-^v}{2\tau_+^v\tau_-^v}f_1^j + \frac{\tau_+^v-\tau_-^v}{2\tau_+^v\tau_-^v}f_3^j + \frac{\rho}{9\tau_+^v}\left(1+\frac{3u_j^2}{c^2}\right) + \frac{\rho u_j}{3c\tau_-^v} + \delta_t\frac{\rho G_j}{3c}, \tag{A1}$$

and

$$0 = -\frac{\tau_+^v+\tau_-^v}{2\tau_+^v\tau_-^v}f_3^j + \frac{\tau_+^v-\tau_-^v}{2\tau_+^v\tau_-^v}f_1^j + \frac{\rho}{9\tau_+^v}\left(1+\frac{3u_j^2}{c^2}\right) - \frac{\rho u_j}{3c\tau_-^v} - \delta_t\frac{\rho G_j}{3c}, \tag{A2}$$

for the TRT multi-direct forcing method. To indicate the position along the y-axis, we use the subscript j for f_k, u, and G. The subtraction of Equation A1 from Equation A2 gives

$$f_1^j - f_3^j = \frac{2\rho}{3c}\left(u_j + \delta_t\tau_-^v G_j\right). \tag{A3}$$

According to Equations 7, 9, 10, 21a, and 21b in the steady state, we have

$$f_5^j = \frac{2\tau_+^v\tau_-^v-\tau_+^v-\tau_-^v}{2\tau_+^v\tau_-^v}f_5^{j-1} + \frac{\tau_+^v-\tau_-^v}{2\tau_+^v\tau_-^v}f_7^{j-1} + \frac{\rho}{36\tau_+^v}\left(1+\frac{3u_{j-1}^2}{c^2}\right) + \frac{\rho u_{j-1}}{12c\tau_-^v} + \delta_t\frac{\rho G_j}{12c}, \tag{A4}$$

$$f_6^j = \frac{2\tau_+^v\tau_-^v-\tau_+^v-\tau_-^v}{2\tau_+^v\tau_-^v}f_6^{j-1} + \frac{\tau_+^v-\tau_-^v}{2\tau_+^v\tau_-^v}f_8^{j-1} + \frac{\rho}{36\tau_+^v}\left(1+\frac{3u_{j-1}^2}{c^2}\right) - \frac{\rho u_{j-1}}{12c\tau_-^v} - \delta_t\frac{\rho G_j}{12c}, \tag{A5}$$

$$f_7^j = \frac{2\tau_+^v\tau_-^v-\tau_+^v-\tau_-^v}{2\tau_+^v\tau_-^v}f_7^{j+1} + \frac{\tau_+^v-\tau_-^v}{2\tau_+^v\tau_-^v}f_5^{j+1} + \frac{\rho}{36\tau_+^v}\left(1+\frac{3u_{j+1}^2}{c^2}\right) - \frac{\rho u_{j+1}}{12c\tau_-^v} - \delta_t\frac{\rho G_j}{12c}, \tag{A6}$$

$$f_8^j = \frac{2\tau_+^v\tau_-^v-\tau_+^v-\tau_-^v}{2\tau_+^v\tau_-^v}f_8^{j+1} + \frac{\tau_+^v-\tau_-^v}{2\tau_+^v\tau_-^v}f_6^{j+1} + \frac{\rho}{36\tau_+^v}\left(1+\frac{3u_{j+1}^2}{c^2}\right) + \frac{\rho u_{j+1}}{12c\tau_-^v} + \delta_t\frac{\rho G_j}{12c}. \tag{A7}$$

Using the momentum density in the x direction given by Equation 5, we obtain

$$\rho u = c(f_1 - f_3 + f_5 - f_6 - f_7 + f_8). \tag{A8}$$

Substituting Equations A3 through A7 into Equation A8 at the 2 positions $j \pm 1$ yields

$$\begin{aligned} &\frac{c}{\rho}(2\tau_+^\nu\tau_-^\nu - \tau_+^\nu - \tau_-^\nu + 1)(f_5^j - f_6^j - f_7^j + f_8^j) = \frac{\tau_-^\nu(2\tau_+^\nu - 1)}{6}(u_{j+1} + u_{j-1}) \\ &- \frac{\tau_+^\nu - 1}{3}u_j + \frac{\tau_-^\nu(1 + \tau_+^\nu + 2\tau_-^\nu - 4\tau_+^\nu\tau_-^\nu)}{6}\delta_t(G_{j+1} + G_{j-1}) + \frac{\tau_+^\nu\tau_-^\nu}{3}\delta_t G_j. \end{aligned} \tag{A9}$$

Substituting Equations A3 and A9 into Equation A8 at position j, and considering Equation 14, we obtain

$$0 = \nu\frac{u_{j+1} - 2u_j + u_{j-1}}{\delta_x^2} + G_j + \frac{1 + \tau_+^\nu + 2\tau_-^\nu - 4\tau_+^\nu\tau_-^\nu}{6}(G_{j+1} - 2G_j + G_{j-1}). \tag{A10}$$

When the total boundary force is given by $\boldsymbol{G}(\boldsymbol{x}_b) = (G_0, 0)$, Equations 18 and 19 yield

$$G_{j_0} = \frac{G_0}{2}, G_{j_0\pm1} = \frac{G_0}{4}, G_{|j-j_0|\geq 2} = 0, \tag{A11}$$

where j_0 denotes the position of the moving surface. Using the symmetry of the flow, we obtain

$$u_{j_0+1} = u_{j_0-1}. \tag{A12}$$

Substitution of Equations A11 and A12 into Equation A10 gives the velocity difference in the immersed boundary layer (IBL):

$$u_{j_0} - u_{j_0+1} = \frac{G_0\delta_x^2}{\nu}\left(\frac{1}{4} - \frac{1 + \tau_+^\nu + 2\tau_-^\nu - 4\tau_+^\nu\tau_-^\nu}{24}\right), \tag{A13}$$

$$u_{j_0+1} - u_{j_0+2} = \frac{G_0\delta_x^2}{\nu}\left(\frac{1}{2} - \frac{1 + \tau_+^\nu + 2\tau_-^\nu - 4\tau_+^\nu\tau_-^\nu}{24}\right). \tag{A14}$$

Because the body force does not spread beyond the IBL, the velocity gradient in the bulk fluid is constant.

$$u_{j-1} - u_j = \frac{G_0\delta_x^2}{2\nu}, \quad \text{for } j > j_0 + 2. \tag{A15}$$

Equation A15 indicates that the force applied at the boundary induces the bulk velocity gradient given by

$$\left(\frac{du}{dy}\right)_{bulk} = \frac{G_0\delta_x}{2\nu}. \tag{A16}$$

Because the velocity is zero on the center plane between the plates, Equation A16 gives the exact solution for the velocity at j_0:

$$\bar{u}_{j_0} = \frac{G_0\delta_x^2}{2\nu}\frac{h}{2}. \tag{A17}$$

Equation A15 gives the velocity at $j_0 + 2$ beyond the IBL as follows:

$$u_{j_0+2} = \frac{G_0\delta_x^2}{2\nu}\left(\frac{h}{2} - 2\right). \tag{A18}$$

Combining Equations A13, A14, and A18, we have

$$u_{j_0} = \frac{G_0\delta_x^2}{\nu}\left(\frac{h}{4} - \frac{1}{4} - \frac{1 + \tau_+^\nu + 2\tau_-^\nu - 4\tau_+^\nu\tau_-^\nu}{12}\right), \tag{A19}$$

and

$$u_{j_0\pm1} = \frac{G_0\delta_x^2}{\nu}\left(\frac{h}{4} - \frac{1}{2} - \frac{1 + \tau_+^\nu + 2\tau_-^\nu - 4\tau_+^\nu\tau_-^\nu}{24}\right). \tag{A20}$$

Subtracting Equation A17 from Equation A19, we obtain the artificial slip velocity as follows:

$$u^s_{j_0} = u_{j_0} - \bar{u}_{j_0} = \frac{G_0\delta_x^2}{\nu}\left(-\frac{1}{3} - \frac{\tau^v_+}{12} - \frac{\tau^v_-}{6} + \frac{\tau^v_+\tau^v_-}{3}\right). \tag{A21}$$

from which we can deduce the following relation between τ^v_- and τ^v_+:

$$\tau^v_- = \frac{\tau^v_+ + 4}{4\tau^v_+ - 2}, \tag{A22}$$

so that $u^s_{j_0} = 0$. The boundary velocity $u_b = u(\boldsymbol{x}_b)$ is given by Equation 17, as follows:

$$u_b = \frac{u_{j_0}}{2} + \frac{u_{j_0+1}}{4} + \frac{u_{j_0-1}}{4}. \tag{A23}$$

Substituting Equations A19 and A20 into Equation A23, we obtain

$$u_b = \frac{G_0\delta_x^2}{\nu}\left(\frac{h}{4} - \frac{7}{16} - \frac{\tau^v_+}{16} - \frac{\tau^v_-}{8} + \frac{\tau^v_+\tau^v_-}{4}\right). \tag{A24}$$

The desired velocity u^d is equal to u_b in the multi-direct forcing method. From Equations A19, A21, and A24, we obtain the analytical solutions given by Equations 55 and 56.

APPENDIX B: ANALYTICAL SOLUTIONS FOR THE POROUS-PLATE PROBLEM

Similarly, we give the analytical solution of the temperature jump in the IB-TLBM based on the iterative correction method with 2 relaxation times. Following the procedure in Appendix A, Equations 29a and 29b give

$$g^j_0 = \frac{4\rho T_j}{9}\left(1 - \frac{3v_j^2}{2c^2}\right) + \frac{4\tau^c_+ - 2}{9}\delta_t\rho Q_j, \tag{B1}$$

$$0 = -\frac{\tau^c_+ + \tau^c_-}{2\tau^c_+\tau^c_-}g^j_1 + \frac{\tau^c_+ - \tau^c_-}{2\tau^c_+\tau^c_-}g^j_3 + \frac{\rho T_j}{9\tau^c_+}\left(1 - \frac{3v_j^2}{2c^2}\right) + \frac{2\tau^c_+ - 1}{18\tau^c_+}\delta_t\rho Q_j, \tag{B2}$$

$$0 = -\frac{\tau^c_+ + \tau^c_-}{2\tau^c_+\tau^c_-}g^j_3 + \frac{\tau^c_+ - \tau^c_-}{2\tau^c_+\tau^c_-}g^j_1 + \frac{\rho T_j}{9\tau^c_+}\left(1 - \frac{3v_j^2}{2c^2}\right) + \frac{2\tau^c_+ - 1}{18\tau^c_+}\delta_t\rho Q_j. \tag{B3}$$

To indicate the position along the y-axis, we use the subscript j for g_k, v, T, and Q. Combining Equations B1 through B3, we obtain

$$g^j_0 + g^j_1 + g^j_3 = \frac{2\rho T_j}{3}\left(1 - \frac{3v_j^2}{2c^2}\right) + \frac{2\tau^c_+ - 1}{3}\delta_t\rho Q_j. \tag{B4}$$

Under the condition given by Equation 58, Equations 29a and 29b yield the following:

$$g^j_2 = \frac{2\tau^c_+\tau^c_- - \tau^c_+ - \tau^c_-}{2\tau^c_+\tau^c_-}g^{j-1}_2 + \frac{\tau^c_+ - \tau^c_-}{2\tau^c_+\tau^c_-}g^{j-1}_4 + \frac{\rho T_{j-1}v_{j-1}}{3c\tau^c_-} + \frac{\rho T_{j-1}}{9\tau^c_+}\left(1 + \frac{3v^2_{j-1}}{c^2}\right) + \frac{2\tau^c_+ - 1}{18\tau^c_+}\delta_t\rho Q_{j-1}, \tag{B5}$$

$$g^j_4 = \frac{2\tau^c_+\tau^c_- - \tau^c_+ - \tau^c_-}{2\tau^c_+\tau^c_-}g^{j+1}_4 + \frac{\tau^c_+ - \tau^c_-}{2\tau^c_+\tau^c_-}g^{j+1}_2 - \frac{\rho T_{j+1}v_{j+1}}{3c\tau^c_-} + \frac{\rho T_{j+1}}{9\tau^c_+}\left(1 + \frac{3v^2_{j+1}}{c^2}\right) + \frac{2\tau^c_+ - 1}{18\tau^c_+}\delta_t\rho Q_{j+1}, \tag{B6}$$

$$g_5^j = \frac{2\tau_+^c\tau_-^c - \tau_+^c - \tau_-^c}{2\tau_+^c\tau_-^c} g_5^{j-1} + \frac{\tau_+^c - \tau_-^c}{2\tau_+^c\tau_-^c} g_7^{j-1} + \frac{\rho T_{j-1} v_{j-1}}{12c\tau_-^c} + \frac{\rho T_{j-1}}{36\tau_+^c}\left(1 + \frac{3v_{j-1}^2}{c^2}\right) + \frac{2\tau_+^c - 1}{72\tau_+^c}\delta_t \rho Q_{j-1}, \tag{B7}$$

$$g_6^j = \frac{2\tau_+^c\tau_-^c - \tau_+^c - \tau_-^c}{2\tau_+^c\tau_-^c} g_6^{j-1} + \frac{\tau_+^c - \tau_-^c}{2\tau_+^c\tau_-^c} g_8^{j-1} + \frac{\rho T_{j-1} v_{j-1}}{12c\tau_-^c} + \frac{\rho T_{j-1}}{36\tau_+^c}\left(1 + \frac{3v_{j-1}^2}{c^2}\right) + \frac{2\tau_+^c - 1}{72\tau_+^c}\delta_t \rho Q_{j-1},$$

$$g_7^j = \frac{2\tau_+^c\tau_-^c - \tau_+^c - \tau_-^c}{2\tau_+^c\tau_-^c} g_7^{j+1} + \frac{\tau_+^c - \tau_-^c}{2\tau_+^c\tau_-^c} g_5^{j+1} - \frac{\rho T_{j+1} v_{j+1}}{12c\tau_-^c} + \frac{\rho T_{j+1}}{36\tau_+^c}\left(1 + \frac{3v_{j+1}^2}{c^2}\right) + \frac{2\tau_+^c - 1}{72\tau_+^c}\delta_t \rho Q_{j+1},$$

$$g_8^j = \frac{2\tau_+^c\tau_-^c - \tau_+^c - \tau_-^c}{2\tau_+^c\tau_-^c} g_8^{j+1} + \frac{\tau_+^c - \tau_-^c}{2\tau_+^c\tau_-^c} g_6^{j+1} - \frac{\rho T_{j+1} v_{j+1}}{12c\tau_-^c} + \frac{\rho T_{j+1}}{36\tau_+^c}\left(1 + \frac{3v_{j+1}^2}{c^2}\right) + \frac{2\tau_+^c - 1}{72\tau_+^c}\delta_t \rho Q_{j+1}. \tag{B10}$$

Substituting Equations B4 through B10 into Equation 30 and considering the definition of the thermal diffusivity given by Equation 14, we obtain

$$\begin{aligned}\frac{T_{j+1}v_{j+1} - T_{j-1}v_{j-1}}{2\delta_x} = \chi\frac{3(T_{j+1}v_{j+1}^2 - 2T_j v_j^2 + T_{j-1}v_{j-1}^2)}{c^2\delta_x^2} + \chi\frac{T_{j+1} - 2T_j + T_{j-1}}{\delta_x^2} \\ + Q_j + \frac{1 + 4\tau_-^c + 4\tau_+^c - 8\tau_+^c\tau_-^c}{12}(Q_{j+1} - 2Q_j + Q_{j-1}).\end{aligned} \tag{B11}$$

Substituting $v(\boldsymbol{x}, t) = v_0$ into Equation B11, we obtain

$$v_0\frac{T_{j+1} - T_{j-1}}{2\delta_x} = \chi\left(1 + \frac{3v_0^2}{c^2}\right)\frac{T_{j+1} - 2T_j + T_{j-1}}{\delta_x^2} + Q_j + \frac{1 + 4\tau_-^c + 4\tau_+^c - 8\tau_+^c\tau_-^c}{12}(Q_{j+1} - 2Q_j + Q_{j-1}). \tag{B12}$$

When the total heat source is given by $Q(\boldsymbol{x}_b) = Q_0$, Equations 18 and 19 yield

$$Q_{j_0} = \frac{Q_0}{2},\ Q_{j_0\pm1} = \frac{Q_0}{4},\ Q_{|j-j_0|\geq 2} = 0, \tag{B13}$$

where j_0 denotes the position of the hot surface. Substituting Equation B13 into Equation B12 gives the following equations in the immersed boundary layer (IBL):

$$\frac{v_0}{2\delta_x}(T_{j_0+1} - T_{j_0-1}) = \frac{Q_0}{2}(1 - \phi) + \frac{\tilde{\chi}}{\delta_x^2}(T_{j_0+1} - 2T_{j_0} + T_{j_0-1}), \tag{B14}$$

$$\frac{v_0}{2\delta_x}(T_{j_0+2} - T_{j_0}) = \frac{Q_0}{4} + \frac{\tilde{\chi}}{\delta_x^2}(T_{j_0+2} - 2T_{j_0+1} + T_{j_0}), \tag{B5}$$

$$\frac{v_0}{2\delta_x}(T_{j_0+3} - T_{j_0+1}) = \frac{Q_0\phi}{4} + \frac{\tilde{\chi}}{\delta_x^2}(T_{j_0+3} - 2T_{j_0+2} + T_{j_0+1}), \tag{B16}$$

where $\tilde{\chi} = \chi(1 + \frac{3v_0^2}{c^2})$ and $\phi = \frac{1+4\tau_-^c+4\tau_+^c-8\tau_+^c\tau_-^c}{12}$.

Because the heat source is equal to zero beyond the IBL, Equation B12 gives

$$\frac{v_0}{2\delta_x}(T_{j+1} - T_{j-1}) = \frac{\tilde{\chi}}{\delta_x^2}(T_{j+1} - 2T_j + T_{j-1}), \tag{B17}$$

for $j_0 + h - 2 > j > j_0 + 2$.

In the IBL around the cool plate at position $j_0 + h$, the total heat source is given by $-Q_0$ due to the symmetry of the boundary-thermal condition, as follows:

$$\frac{v_0}{2\delta_x}(T_{j_0+h-1} - T_{j_0+h-3}) = -\frac{Q_0\phi}{4} + \frac{\tilde{\chi}}{\delta_x^2}(T_{j_0+h-1} - 2T_{j_0+h-2} + T_{j_0+h-3}), \tag{B18}$$

$$\frac{v_0}{2\delta_x}(T_{j_0+h} - T_{j_0+h-2}) = -\frac{Q_0}{4} + \frac{\tilde{\chi}}{\delta_x^2}(T_{j_0+h} - 2T_{j_0+h-1} + T_{j_0+h-2}). \tag{B19}$$

The sum of Equations B14 through B19 over j gives

$$\frac{v_0}{2\delta_x}(T_{j_0+h-1} - T_{j_0-1} + T_{j_0+h} - T_{j_0}) = \frac{Q_0}{2}(1-\phi) + \frac{\tilde{\chi}}{\delta_x^2}(T_{j_0-1} - T_{j_0+h-1} + T_{j_0+h} - T_{j_0}). \tag{B20}$$

Because the temperature distribution exhibits reverse symmetry with respect to the plates, as shown in Figure 4, we obtain

$$T_{j_0+h} = -T_{j_0}, T_{j_0+h-1} = -T_{j_0-1}. \tag{B21}$$

Substituting Equation B21 into Equation B20 yields

$$\frac{2\tilde{\chi} + \delta_x v_0}{\delta_x^2} T_{j_0-1} - \frac{2\tilde{\chi} - \delta_x v_0}{\delta_x^2} T_{j_0} = -\frac{Q_0(1-\phi)}{2}. \tag{B22}$$

Combining Equations B14 through B19 and B22, we obtain

$$T_{j_0} - \gamma^h T_{j_0+h} = \frac{\delta_x^2 Q_0(1 + \gamma + \gamma^2 + \cdots + \gamma^{h-1})}{2\tilde{\chi} + \delta_x v_0} - \frac{\delta_x^2 Q_0((1+\phi)(1+\gamma^{h-1}) + \phi(\gamma + \gamma^{h-2}))}{2(2\tilde{\chi} + \delta_x v_0)}, \tag{B23}$$

where $\gamma = (2\tilde{\chi} - \delta_x v_0)/(2\tilde{\chi} + \delta_x v_0)$.

Equations B21 and B23 give

$$T_{j_0} = \bar{T}_{j_0} + T^J_{j_0}, \tag{B24}$$

where $\bar{T}_{j_0}$ is the analytical solution of the temperature at j_0, and $T^J_{j_0}$ is the solution of the temperature jump, given by

$$\bar{T}_{j_0} = \frac{(1-\gamma^h)\delta_x^2 Q_0}{(1-\gamma)(1+\gamma^h)(2\tilde{\chi} + \delta_x v_0)}, \tag{B25}$$

$$T^J_{j_0} = -\frac{\delta_x^2 Q_0((1+\phi)(1+\gamma^{h-1}) + \phi(\gamma + \gamma^{h-2}))}{2(1+\gamma^h)(2\tilde{\chi} + \delta_x v_0)}, \tag{B26}$$

in the porous-plate problem ($v_0 \neq 0$ and $\gamma \neq 1$).

The boundary temperature $T_b = T(\boldsymbol{x}_b)$ is given by Equations 17, 18, and 19 as follows:

$$T_b = \frac{T_{j_0}}{2} + \frac{T_{j_0+1}}{4} + \frac{T_{j_0-1}}{4}. \tag{B27}$$

Equation B22 gives

$$T_{j_0-1} = \frac{2\tilde{\chi} - \delta_x v_0}{2\tilde{\chi} + \delta_x v_0} T_{j_0} - \frac{\delta_x^2 Q_0(1-\phi)}{2(2\tilde{\chi} + \delta_x v_0)}. \tag{B28}$$

The sum of Equations B14 and B22 yields

$$T_{j_0+1} = \frac{2\tilde{\chi} + \delta_x v_0}{2\tilde{\chi} - \delta_x v_0} T_{j_0} - \frac{\delta_x^2 Q_0(1-\phi)}{2(2\tilde{\chi} - \delta_x v_0)}. \tag{B29}$$

Substituting Equations B28 and B29 into Equation B27 gives

$$T_b = \frac{\tilde{\chi}}{4\tilde{\chi}^2 - \delta_x^2 v_0^2}\left(4\tilde{\chi} T_{j_0} - \frac{\delta_x^2 Q_0(1-\phi)}{2}\right). \tag{B30}$$

By substituting Equations B24, B25, and B26 into Equation B30, we obtain

$$T_b = \frac{\delta_x^2 Q_0 \tilde{\chi}}{4\tilde{\chi}^2 - \delta_x^2 v_0^2}\left[\frac{\phi - 1}{2} + \frac{4\tilde{\chi}}{(1+\gamma^h)(2\tilde{\chi} + \delta_x v_0)}\left(\frac{1-\gamma^h}{1-\gamma} - \frac{(1+\phi)(1+\gamma^{h-1}) + \phi(\gamma + \gamma^{h-2})}{2}\right)\right], \tag{B31}$$

for the porous-plate problem.

Because the desired temperature T^d is equal to the boundary temperature T_b in the iterative correction method, Equation B31 gives T^d. The analytical solutions 66 and 67 of the porous-plate problem ($v_0 \neq 0$) are obtained using Equations B24-B26, and B31. For the direct forcing method, we obtain Equations 64 and 65 using a similar procedure, with $T^d = T_b + \delta_t Q_0$.