Commun. Comput. Phys.
doi: 10.4208/cicp.090815.170316a

Vol. **20**, No. 5, pp. 1210-1257
November 2016

An Efficient Immersed Boundary-Lattice Boltzmann Method for the Simulation of Thermal Flow Problems

Yang Hu[1], Decai Li[1,*], Shi Shu[2] and Xiaodong Niu[3]

[1] *School of Mechanical, Electronic and Control Engineering, Beijing Jiaotong University, Beijing, China.*
[2] *School of Mathematics and Computational Science, Xiangtan University, Xiangtan, China.*
[3] *College of Engineering, Shantou University, Shantou, Guangdong, China.*

Received 9 August 2015; Accepted (in revised version) 17 March 2016

Abstract. In this paper, a diffuse-interface immersed boundary method (IBM) is proposed to treat three different thermal boundary conditions (Dirichlet, Neumann, Robin) in thermal flow problems. The novel IBM is implemented combining with the lattice Boltzmann method (LBM). The present algorithm enforces the three types of thermal boundary conditions at the boundary points. Concretely speaking, the IBM for the Dirichlet boundary condition is implemented using an iterative method, and its main feature is to accurately satisfy the given temperature on the boundary. The Neumann and Robin boundary conditions are implemented in IBM by distributing the jump of the heat flux on the boundary to surrounding Eulerian points, and the jump is obtained by applying the jump interface conditions in the normal and tangential directions. A simple analysis of the computational accuracy of IBM is developed. The analysis indicates that the Taylor-Green vortices problem which was used in many previous studies is not an appropriate accuracy test example. The capacity of the present thermal immersed boundary method is validated using four numerical experiments: (1) Natural convection in a cavity with a circular cylinder in the center; (2) Flows over a heated cylinder; (3) Natural convection in a concentric horizontal cylindrical annulus; (4) Sedimentation of a single isothermal cold particle in a vertical channel. The numerical results show good agreements with the data in the previous literatures.

AMS subject classifications: 76M28, 76R05, 76R10

Key words: Immersed boundary method, lattice Boltzmann method, thermal flow, Dirichlet boundary condition, Neumann boundary condition, Robin boundary condition, Taylor Green vortices.

*Corresponding author. *Email addresses:* hy304@126.com (Y. Hu), dcli@bjtu.edu.cn (D. Li), shushi@xtu.edu.cn (S. Shu), xdniu@stu.edu.cn (X. Niu)

 ©2016 Global-Science Press

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press

1 Introduction

The treatment of boundary conditions directly affects the accuracy and stability of the numerical simulation. How to resolve complex boundaries accurately and efficiently is a great challenging issue for CFD researchers. There are a large number of flow and heat transfer problems in irregular regions in the fields of science and engineering. Currently there exist two kinds of categories to implement the boundary conditions in the complex domains: body-fitted and fixed grid methods. The key issues of body-fitted methods are mapping the irregular regions to the regular domains by using algebraic and differential methods and then solving the governing equations in the structure meshes. These methods which couple boundary conditions and governing equations are extremely difficult to solve. When resolving moving boundaries in fluid-structure interactions, body-fitted methods require the complex grid remapping at every time step which may involve very large computational costs and cause loss of the computational accuracy. Fixed grid methods are another kinds of techniques to treat the complex boundary problems. The basic idea is to extend the computational domain to a simpler computational domain, and the boundary conditions can be enforced on the boundary. In the CFD simulations, three different fixed grid techniques are very popular in recent years: distributed Lagrange multiplier/fictitious domain method (DLM/FDM), immersed interface method (IIM) and immersed boundary method (IBM).

The DLM/FDM was firstly introduced by Glowinski which is based on the finite element method [1,2]. The main features of DLM/FDM are that the governing equations are discretized in space using a fixed grid, and the boundary conditions on the original domain can be enforced by applying the Lagrange multiplier technique. Glowinski *et al*. had successfully implemented this method to simulate the incompressible particulate flows. Later the DLM/FDM was extended to deal with heat transfer problems [3,4]. The immersed interface method is also a fixed grid method by adopting a Cartesian grid. The IIM was proposed by Leveque and Li which incorporates the jump conditions into finite difference or finite volume schemes [5]. This method establishes the relationships between jumps and singular forces. And it computes the correction terms in the discretized governing equations to obtain the high accuracy solutions. The IIM has been used in the Stokes problem, Navier-Stokes flows, discontinuous viscosity problem [6–8].

In contrast to the fictitious domain method and immersed interface method, the immersed boundary method (IBM) which is also named the discrete delta function approach is used more widely for the interface problems. As early as 1970s, the original IBM was developed to study the blood flow problem in the heart by Peskin [9]. The method has been extended to many fluid and biological problems in complex geometries. The IBM replaces singular Dirac function by a smoother regularization function. The governing equations are discretized and solved on a fixed Eulerian grid and the interface is represented by a set of Lagrangian points. The forces at the Lagrangian points are spread to the near Eulerian points. When resolving the moving boundary problems, we only track the positions of the Lagrangian points. So the IBM reduces the computa-

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press

tional efforts greatly.

In addition, as an alternative computational technique, the lattice Boltzmann method (LBM) has been widely applied to study many flow and heat transfer problems. In the LBM models, the velocity and temperature fields are described by the density and temperature distribution functions, respectively, and the distribution functions are computed by simple collision and stream steps. The lattice Boltzmann scheme can be viewed as a finite difference scheme of the continuous Boltzmann equation [10]. It simulates the motion of particle micelles of the fluid and it is very suitable to parallel arithmetic. Furthermore, the program implementation of LBM is relatively easy. Thus, this method is popular in recent years (for more details, see the review [11]). Now LBM have been used in many different application fields which involves fluid system, such as multicomponent/multiphase flows [12, 13], turbulence [14, 15], microflows [16, 17], fluid-solid interactions [18–22], thermal flows [23–27], porous media flows [28, 29].

Since the standard LBM adopts a uniform Cartesian grid, to combine the advantages of LBM and IBM is a natural idea. The pioneering works were done by Feng and Machaelides in 2004-2005 [30,31]. They employed the penalty method and so-called "Proteus" technique to calculate the force density at the Lagrangian points and successfully simulated the particulate flows. Soon afterwards a series of IB-LBMs are proposed to study the rigid and elastic body problems [18, 32–36].

Due to the high efficiency of IB algorithms, there are many works to extend IBM to investigate the thermal flows. Kim *et al*. developed an immersed-boundary finite-volume method for heat transfer problems in complex geometries [37]. Pacheco *et al*. reported another immersed-boundary finite-volume method on nonstaggered grids to study the heat transfer and fluid flow problems [38, 39]. Zhang *et al*. proposed a computational technique to deal with the thermal flows under different temperature boundary conditions [40]. Feng *et al*. presented a direct numerical simulation method (DNS) to solve the heat transfer equations in particulate flows [41]. Wang *et al*. extended their idea of multi-forcing method to thermal flows and proposed the multi-heat source scheme [42]. Jeong *et al*. proposed an IB-LBM using an equilibrium internal energy density approach to simulate the natural convention in a cavity with various body shapes for different Rayleigh numbers [43]. Ren *et al*. applied the boundary condition-enforced immersed boundary method to study the natural and forced convection problems with the Dirichlet boundary condition [24, 44]. Then they developed a heat flux correction method to treat the heat flux boundary conditions [45].

In the above works, many IBMs are designed for the heat transfer problems with the Dirichlet boundary condition. Comparatively speaking, there are far fewer works to formulate the IB algorithms to resolve the Neumann boundary condition. For sharp-interface IBMs [37–40], many researchers directly discrete the normal derivative of the temperature. The implementation methods of thermal sharp-interface IBMs for the Dirichlet boundary condition and Neumann boundary condition are similar. However, for the diffuse-interface IBM, the treatment method of the Dirichlet boundary condition can not extend to deal with the Neumann boundary condition directly. Recently, Ren *et al*. pro-

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press

posed a heat flux method which treats the Neumann condition directly based on diffuse-interface IBM [45]. Their method roots from physical conservation law. Moreover, to the best of our knowledge, there are not any works of diffuse-interface IBM to investigate the thermal flows with the Robin (mixed) type boundary condition.

In this paper we construct an efficient diffuse-interface immersed boundary method for the simulation of flow in complex geometries with heat transfer. The present IB algorithm involves three parts which deals with the temperature Dirichlet, Neumann and Robin(mixed)-types boundary conditions, respectively. And the present diffuse-interface IBM is employed combined with LBM. Concretely speaking, we extend the improved momentum exchange-based IB-LBM to treat the temperature field. The main advantage of this method is to satisfy the temperature Dirichlet boundary condition accurately. The basic idea is to correct the temperature field using an iterative procedure. For the Neumann and Robin Boundary conditions, the jump conditions of the derivative of temperature is derived. In the framework of IBM, the jumps can be smoothed by the regularized Dirac delta function. Then the implicit equations of the heat flux at the Lagrangian points are established. The temperature field can be corrected by distributing the heat flux on the immersed boundary to the neighboring Eulerian points. Different from the Ref. [45], the present algorithm is based on the mathematical manipulation and has a rigorous mathematical foundation. The differences of the two methods are listed as follow: Firstly, the method proposed by Ren *et al*. is implemented with assistance of physical interpretation. However the present method is based on the jump conditions at the interface. In Ren *et al*.'s method, by using the fractional step procedure, when the Neumann boundary condition is not satisfied by the predicted temperature field, the different of the given normal derivative and the predicted normal derivative will contribute as a heat flux. However, in our method, the difference of the normal derivative from outside of the interface and that from inside will contribute as a heat flux. Secondly, the interpolation formulas are different in the two methods. The formulas used in Ren *et al*.'s method are the conventional computational formulas which can only be used in smooth domain. However, in the present work, the interpolation formulas which hold in the non-smooth domain are used. Thirdly, the jump condition of tangential component is involved.

In addition, we also discuss the accuracy of the diffuse-interface IBM. In many previous studies [22, 46–54], the decay vortices problem was used to test the accuracy of the IBM. Second-order accuracy can be reached when this test example is employed. However, according to the analysis of this study, the Taylor-Green vortices problem is very special and it is not suitable as a test case. Based on this understanding, we select the examples which can obtain the general conclusion to test the accuracy of the present thermal IBM. Some examples including natural and forced convection problems with the three kinds of boundary conditions are simulated. The numerical results are in good agreements with the previous solutions.

This paper is organized as follows. In Section 2.1, we simply describe the evolution equations of flow and thermal fields. Then IBM for velocity field is introduced in Section 2.2. In Sections 2.3, 2.4 and 2.5, the details of the present IBM to implement three types of

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press

boundary conditions are presented. Several numerical examples are simulated in Section 3. Some conclusions are given in Section 4.

2 Numerical method

2.1 The evolution equations of flow and thermal fields

As shown in Fig. 1, a bounded domain $\Omega_1 \subset R^2$ for the viscous incompressible flow is considered

$$\nabla\cdot\mathbf{u}=0, \qquad \mathbf{x}\in\Omega_1, \tag{2.1}$$

$$\rho\Big(\frac{\partial\mathbf{u}}{\partial t}+(\mathbf{u}\cdot\nabla)\mathbf{u}\Big)=-\nabla p+\mu\nabla^2\mathbf{u}+\mathbf{f}_e, \tag{2.2}$$

$$\rho c_p\Big(\frac{\partial T}{\partial t}+\mathbf{u}\cdot\nabla T\Big)=\kappa\nabla^2 T, \tag{2.3}$$

with the velocity boundary condition

$$\mathbf{u}(\mathbf{X}(s,t))=\mathbf{U}_B(\mathbf{X}(s,t)), \tag{2.4}$$

and the thermal boundary conditions on Γ

$$T(\mathbf{X}(s,t))=T_B(\mathbf{X}(s,t)), \tag{2.5}$$

or

$$-\kappa\frac{\partial T}{\partial n}(\mathbf{X}(s,t))=Q_B(\mathbf{X}(s,t)), \tag{2.6}$$

or

$$-\kappa\frac{\partial T}{\partial n}(\mathbf{X}(s,t))=h_e(T_B-T(\mathbf{X}(s,t))), \tag{2.7}$$

where $\mathbf{x}$ is the coordinate of Eulerian node, and $\mathbf{X}$ is the coordinate of Lagrangian node. $\rho,\mathbf{u},p,T$ represent density, velocity, pressure and temperature, respectively, and $\mathbf{U}_B,T_B$ are the velocity and temperature on the immersed boundary. The physical parameter μ,c_p,κ,h_e are the dynamics viscosity, specific heat capacity, thermal conductivity and heat transfer coefficient. $\mathbf{f}_e$ is the body force which acts on the fluid.

When IBM is applied, the computational domain must extend to a regular domain $\Omega_1\bigcup\Omega_2$. The effect of the boundary conditions on the immersed boundary are loaded as a force term or heat source/sink in the momentum or energy equation. In this view, the following governing equations are solved on the regular domain

$$\nabla\cdot\mathbf{u}=0, \qquad \mathbf{x}\in\Omega_1\bigcup\Omega_2, \tag{2.8}$$

$$\rho\Big(\frac{\partial\mathbf{u}}{\partial t}+(\mathbf{u}\cdot\nabla)\mathbf{u}\Big)=-\nabla p+\mu\nabla^2\mathbf{u}+\mathbf{f}_e+\int_\Gamma\mathbf{F}(\mathbf{X}(s,t))\delta(\mathbf{x}-\mathbf{X}(s,t))ds, \tag{2.9}$$

$$\rho c_p\Big(\frac{\partial T}{\partial t}+\mathbf{u}\cdot\nabla T\Big)=\kappa\nabla^2 T+\int_\Gamma\delta Q(\mathbf{X}(s,t))\delta(\mathbf{x}-\mathbf{X}(s,t))ds, \tag{2.10}$$

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press

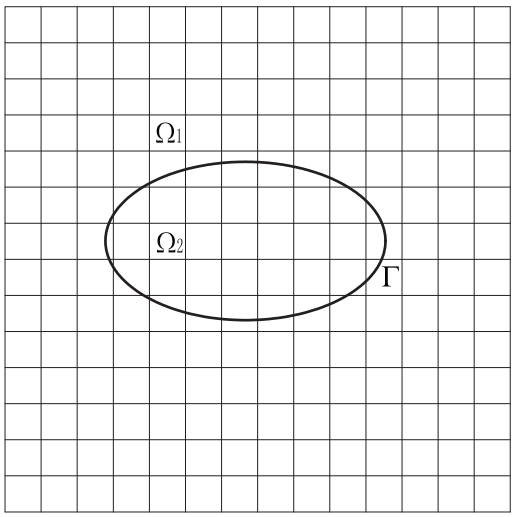


Figure 1: A two-dimensional computational domain contains an immersed boundary.

$\mathbf{F}(\mathbf{X}(s,t)), Q(\mathbf{X}(s,t))$ are the force density and heat flux on the immersed boundary, and $\delta(\mathbf{x}-\mathbf{X}(s,t))$ denotes the Dirac delta function.

The lattice Boltzmann method is chosen as fluid and temperature fields solver. It is an efficient alternative technique of the Navier-Stokes solver. The evolution equations of double-population LBM with source terms can be written as

$$f_\alpha(\mathbf{x}+\mathbf{e}_\alpha\Delta t,t+\Delta t)-f_\alpha(\mathbf{x},t)=-\frac{1}{\tau_f}(f_\alpha(\mathbf{x},t)-f_\alpha^{eq}(\mathbf{x},t))+F_\alpha\Delta t, \tag{2.11}$$

$$g_\alpha(\mathbf{x}+\mathbf{e}_\alpha\Delta t,t+\Delta t)-g_\alpha(\mathbf{x},t)=-\frac{1}{\tau_g}(g_\alpha(\mathbf{x},t)-g_\alpha^{eq}(\mathbf{x},t))+G_\alpha\Delta t, \tag{2.12}$$

where $f_\alpha(\mathbf{x},t), g_\alpha(\mathbf{x},t)$ denote the density and temperature distribution functions for the discrete velocity $\mathbf{e}_\alpha$, Δt is the time step, and τ_f,τ_g are the dimensionless relaxation times of flow and thermal fields, respectively. $f_\alpha^{eq}(\mathbf{x},t), g_\alpha^{eq}(\mathbf{x},t)$ are the local equilibrium density and temperature distribution functions, and they can be calculated by

$$f_\alpha^{eq}(\mathbf{x},t)=\omega_\alpha\rho\left(1+\frac{\mathbf{e}_\alpha\cdot\mathbf{u}}{c_s^2}+\frac{(\mathbf{e}_\alpha\cdot\mathbf{u})^2}{2c_s^4}-\frac{\mathbf{u}^2}{2c_s^2}\right), \tag{2.13}$$

$$g_\alpha^{eq}(\mathbf{x},t)=\omega_\alpha T\left(1+\frac{\mathbf{e}_\alpha\cdot\mathbf{u}}{c_s^2}+\frac{(\mathbf{e}_\alpha\cdot\mathbf{u})^2}{2c_s^4}-\frac{\mathbf{u}^2}{2c_s^2}\right), \tag{2.14}$$

where c_s is the lattice sound speed and ω_α is the weight coefficient which depends on the lattice velocity model.

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press

In our study, the D2Q9 model is used, and the discrete velocity set is defined as

$$\mathbf{e}_\alpha = \begin{cases} (0,0), & \alpha = 0, \\ \left(\cos\left[(\alpha-1)\frac{\pi}{2}\right], \sin\left[(\alpha-1)\frac{\pi}{2}\right]\right)c, & \alpha = 1,2,3,4, \\ \sqrt{2}\left(\cos\left[(2\alpha-1)\frac{\pi}{4}\right], \sin\left[(2\alpha-1)\frac{\pi}{4}\right]\right)c, & \alpha = 5,6,7,8, \end{cases} \tag{2.15}$$

where $c = \Delta x/\Delta t$, Δx is the lattice spacing. Further, we have $c = \sqrt{3}c_s$. The corresponding weight coefficients are expressed as

$$\omega_\alpha = \begin{cases} \frac{4}{9}, & \alpha = 0, \\ \frac{1}{9}, & \alpha = 1,2,3,4, \\ \frac{1}{36}, & \alpha = 5,6,7,8. \end{cases} \tag{2.16}$$

The relaxation times τ_f and τ_g are related to the viscosity and thermal conductivity, and they are defined as

$$\tau_f = \frac{\mu}{\rho c_s^2 \Delta t} + 0.5, \tag{2.17}$$

$$\tau_g = \frac{\kappa}{\rho c_p c_s^2 \Delta t} + 0.5. \tag{2.18}$$

The discrete force term F_α in Eq. (2.11) and the heat source/sink term G_α in Eq. (2.12) are defined as

$$F_\alpha = \left(1 - \frac{1}{2\tau_f}\right)\omega_\alpha \left(\frac{\mathbf{e}_\alpha - \mathbf{u}}{c_s^2} + \frac{\mathbf{e}_\alpha \cdot \mathbf{u}}{c_s^2}\mathbf{e}_\alpha\right) \cdot (\mathbf{f}_e + \mathbf{f}_b), \tag{2.19}$$

$$G_\alpha = \left(1 - \frac{1}{2\tau_g}\right)\omega_\alpha q. \tag{2.20}$$

The macroscopic density, velocity and temperature in the LBM can be calculated by

$$\rho = \sum_\alpha f_\alpha, \tag{2.21}$$

$$\mathbf{u} = \frac{\sum_\alpha \mathbf{e}_\alpha f_\alpha + \frac{1}{2}(\mathbf{f}_b + \mathbf{f}_e)\Delta t}{\rho}, \tag{2.22}$$

$$T = \sum_\alpha g_\alpha + \frac{1}{2} q \Delta t. \tag{2.23}$$

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press

2.2 IBM for the velocity Dirichlet boundary condition

The computational domain is discretized on a Cartesian staggered grid, and the Eulerian points are denoted by $\mathbf{x}_{i,j}=(x_{i,j},y_{i,j})$, $i=1,2,\cdots,M$; $j=1,2,\cdots,N$ with the spatial spacing is $\Delta x=\Delta y=h$. The immersed boundary is represented by a set of Lagrangian points $\mathbf{X}_l=(X_l,Y_l)$, $l=1,2,\cdots,K$ with the boundary element length Δs_l.

In the present study, the non-slip boundary condition of flow field is implemented by applying the improved momentum exchange-based IBM [51]. Now we describe this method as follows. As shown in Fig. 2(a), a control arc element Δs and a control volume $\Delta V=\Delta sh$ are considered. By integrating the force density within the control arc element, we have

$$\begin{aligned}\int_{\Delta V}\mathbf{f}(\mathbf{x})dV&=\int_{\Delta V}\int_{\Delta s}\mathbf{F}(\mathbf{X})\delta(\mathbf{x}-\mathbf{X})dsdV\\&=\int_{\Delta s}\mathbf{F}(\mathbf{X})\int_{\Delta V}\delta(\mathbf{x}-\mathbf{X})dVds=\int_{\Delta s}\mathbf{F}(\mathbf{X})ds.\end{aligned}\tag{2.24}$$

The numerical integration of the above equation gives

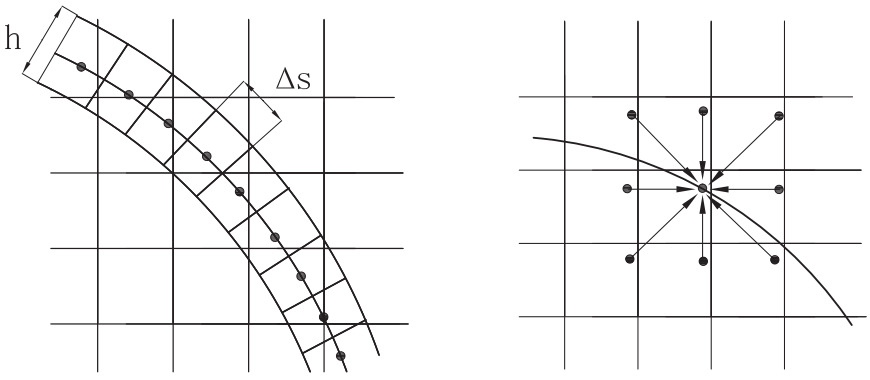
$$\mathbf{F}(\mathbf{X}_l)\Delta s=\mathbf{f}(\mathbf{X}_l)\Delta V.\tag{2.25}$$

As a result, we have

$$\mathbf{F}(\mathbf{X}_l)=\mathbf{f}(\mathbf{X}_l)h.\tag{2.26}$$

As shown in Fig. 2(b), the body force density $\mathbf{f}(\mathbf{X}_l)$ can be computed using the momentum exchange method

$$\mathbf{f}(\mathbf{X}_l)=\frac{\sum_\alpha \mathbf{e}_{-\alpha}\bar{f}_{-\alpha}(\mathbf{X}_l)-\sum_\alpha \mathbf{e}_\alpha f_\alpha(\mathbf{X}_l)}{\Delta t},\tag{2.27}$$



(a) Geometry configuration of the immersed boundary method.

(b) The interactions between fluid particles and immersed boundary

Figure 2: A schematic diagram used to describe the momentum exchange-based IBM.

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press

where

$$\bar{f}_{-\alpha}(\mathbf{X}_l)=f_\alpha(\mathbf{X}_l)-2\omega_{-\alpha}\rho\frac{\mathbf{e}_{-\alpha}\cdot\mathbf{U}_B^d(\mathbf{X}_l)}{c_s^2}, \tag{2.28}$$

where $-\alpha$ denotes the opposite direction of α, namely $\mathbf{e}_{-\alpha}=-\mathbf{e}_\alpha$, and $\mathbf{U}_B^d(\mathbf{X}_l)$ is the velocity on the boundary.

In summary, the force density on the boundary can be obtained as follow

$$\mathbf{F}(\mathbf{X}_l)=\frac{\sum_\alpha \mathbf{e}_{-\alpha}\bar{f}_{-\alpha}(\mathbf{X}_l)-\sum_\alpha \mathbf{e}_\alpha f_\alpha(\mathbf{X}_l)}{\Delta t}h=2\rho(\mathbf{X}_l)\frac{\mathbf{U}_B^d(\mathbf{X}_l)-\mathbf{U}(\mathbf{X}_l)}{\Delta t}h. \tag{2.29}$$

As explained in Ref. [51], the explicit scheme can not enforce the no-slip boundary condition. Moreover, the performance of the implicit scheme would be effected by the distribution of the Lagrangian points. So the iterative method is used in the present IBM.

Now we describe the iterative procedure in the following subsection. The initial velocity (intermediate velocity) at the Eulerian points can be calculated by

$$\mathbf{u}^{n,0}(\mathbf{x}_{i,j})=\frac{\sum_\alpha \mathbf{e}_\alpha f_\alpha(\mathbf{x}_{i,j})+\frac{1}{2}\mathbf{f}_e(\mathbf{x}_{i,j})\Delta t}{\rho(\mathbf{x}_{i,j})}. \tag{2.30}$$

Step 1 Interpolate

$$\mathbf{U}^{n,k}(\mathbf{X}_l)=\sum_{i,j}D_{i,j}^l\mathbf{u}^{n,k}(\mathbf{x}_{i,j})h^2, \tag{2.31}$$

where $D(\mathbf{x}_{i,j}-\mathbf{X}_l)$ is the discrete delta function [55]

$$D_{ij}^l=D(\mathbf{x}_{i,j}-\mathbf{X}_l)=\frac{1}{h^2}\delta\Big(\frac{x_{i,j}-X_l}{h}\Big)\delta\Big(\frac{y_{i,j}-Y_l}{h}\Big), \tag{2.32}$$

with

$$\delta(r)=\begin{cases}\frac{1}{8}(3-2|r|+\sqrt{1+4|r|-4r^2}), & 0\le|r|<1,\\ \frac{1}{8}(5-2|r|-\sqrt{-7+12|r|-4r^2}), & 1\le|r|<2,\\ 0, & |r|\ge 2.\end{cases} \tag{2.33}$$

Step 2 Calculate force density (momentum exchange rule)

$$\Delta\mathbf{F}^{n,k}(\mathbf{X}_l)=2\frac{h}{\Delta t}\rho(\mathbf{X}_l)(\mathbf{U}_B^d(\mathbf{X}_l)-\mathbf{U}^{n,k}(\mathbf{X}_l)). \tag{2.34}$$

Step 3 Distribute

$$\Delta\mathbf{f}^{n,k}(\mathbf{x}_{i,j})=\sum_l D_{i,j}^l\Delta\mathbf{F}^{n,k}(\mathbf{X}_l)\Delta s_l. \tag{2.35}$$

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press

Step 4 Correct

$$\mathbf{u}^{n,k+1}(\mathbf{x}_{i,j})=\mathbf{u}^{n,k}(\mathbf{x}_{i,j})+\frac{\lambda\Delta\mathbf{f}^{n,k}(\mathbf{x}_{i,j})\Delta t}{2\rho(\mathbf{x}_{i,j})}, \tag{2.36}$$

where λ is the relaxation parameter.

Actually, based on the Eqs. (2.31), (2.34), (2.35) and (2.36), we have

$$\begin{aligned}\mathbf{U}^{n,k+1}(\mathbf{X}_l)=&\left(1-\frac{\lambda h^3\rho(\mathbf{X}_l)}{\rho(\mathbf{x}_{i,j})}\sum_l\sum_{i,j}D^l_{i,j}D^r_{i,j}\Delta s_r\right)\mathbf{U}^{n,k}(\mathbf{X}_r)\\&+\frac{\lambda h^3\rho(\mathbf{X}_l)}{\rho(\mathbf{x}_{i,j})}\sum_r\sum_{i,j}D^l_{i,j}D^r_{i,j}\Delta s_r\mathbf{U}^d_B(\mathbf{X}_r).\end{aligned} \tag{2.37}$$

If we select an appropriate relaxation parameter λ, the iteration converges. Obviously, the limit of velocity on the boundary is

$$\lim_{k\to\infty}\mathbf{U}^{n,k}=\mathbf{U}^d_B. \tag{2.38}$$

The above procedure stop after a fixed number of iterations. Then, we get

$$\mathbf{u}^{n+1}(\mathbf{x}_{i,j})=\mathbf{u}^{n,s}(\mathbf{x}_{i,j}), \tag{2.39}$$

where s is the number of iterations.

In the present simulation, the force density $\mathbf{F}(\mathbf{X}_l)$ is computed as follow

$$\mathbf{F}(\mathbf{X}_l)=\sum_{k=0}^{s}\left(\left(1-\frac{\lambda h^3\rho(\mathbf{X}_l)}{\rho(\mathbf{x}_{i,j})}\sum_r\sum_{i,j}D^l_{i,j}D^r_{i,j}\Delta s_r\right)^k\right)\Delta\mathbf{F}^0(\mathbf{X}_l). \tag{2.40}$$

Extensive numerical tests indicate that the range of optimal relaxation parameter λ is $2-2.5$, and the number of iterations s can be set as $5-10$ (for more details, see [51]).

2.3 IBM for the thermal Dirichlet boundary condition

For the Dirichlet type boundary condition of temperature field, a similar technique is applied to get the heat flux on the boundary.

First of all, we compute the distribution functions at the Lagrangian points by

$$g_\alpha(\mathbf{X}_l)=\sum_{i,j}g_\alpha(\mathbf{x}_{i,j})D(\mathbf{x}_{i,j}-\mathbf{X}_l)h^2. \tag{2.41}$$

Secondly, the new distribution functions are calculated by [23]

$$g^{new}_\alpha(\mathbf{X}_l)=2\omega_{-\alpha}T^d_B(\mathbf{X}_l)-g_{-\alpha}(\mathbf{X}_l), \tag{2.42}$$

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press

where T_B^d is the temperature of the boundary.

Finally, the heat flux on the immersed boundary can be obtained by

$$\delta Q(\mathbf{X}_l)=\frac{\sum_\alpha(g_\alpha^{new}(\mathbf{X}_l)-g_\alpha(\mathbf{X}_l))h}{\Delta t}. \tag{2.43}$$

We can also get the simple formula of the heat flux as follow

$$\delta Q(\mathbf{X}_l)=2\frac{h}{\Delta t}(T_B^d(\mathbf{X}_l)-T(\mathbf{X}_l)). \tag{2.44}$$

As discussed above, only one iteration can not enforce the given temperature on the boundary. So an iterative method is proposed. The iterative procedure is described in the following. The initial temperature (intermediate temperature) at the Eulerian points can be calculated by

$$T^{n,0}(\mathbf{x}_{i,j})=\sum_\alpha g_\alpha(\mathbf{x}_{i,j}). \tag{2.45}$$

Step 1 Interpolate

$$T^{n,k}(\mathbf{X}_l)=\sum_{i,j}D_{i,j}^l T^{n,k}(\mathbf{x}_{i,j})h^2. \tag{2.46}$$

Step 2 Calculate heat flux

$$\Delta Q^{n,k}(\mathbf{X}_l)=2\frac{h}{\Delta t}(T_B^d(\mathbf{X}_l)-T^{n,k}(\mathbf{X}_l)). \tag{2.47}$$

Step 3 Distribute

$$\Delta q^{n,k}(\mathbf{x}_{i,j})=\sum_l D_{i,j}^l\Delta Q^{n,k}(\mathbf{X}_l)\Delta s_l. \tag{2.48}$$

Step 4 Correct

$$T^{n,k+1}(\mathbf{x}_{i,j})=T^{n,k}(\mathbf{x}_{i,j})+\frac{1}{2}\lambda\Delta q^{n,k}(\mathbf{x}_{i,j})\Delta t. \tag{2.49}$$

Similarly, we get the following iterative relationship by Eqs. (2.46), (2.47), (2.48) and (2.49)

$$\begin{aligned}T^{n,k+1}(\mathbf{X}_l)=&\left(1-\lambda h^3\sum_l\sum_{i,j}D_{i,j}^l D_{i,j}^r\Delta s_r\right)T^{n,k}(\mathbf{X}_r)\\&+\lambda h^3\sum_r\sum_{i,j}D_{i,j}^l D_{i,j}^r\Delta s_r T_B^d(\mathbf{X}_r).\end{aligned} \tag{2.50}$$

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press

A good relaxation parameter λ can ensure that the following formula holds

$$\lim_{k\to\infty} T^{n,k} = T_B^d. \tag{2.51}$$

Similarly, the temperature at the time level $n+1$ can be obtained

$$T^{n+1}(\mathbf{x}_{i,j}) = T^{n,s}(\mathbf{x}_{i,j}), \tag{2.52}$$

and the heat flux $\delta Q(\mathbf{X}_l)$ is computed as follow

$$\delta Q(\mathbf{X}_l) = \sum_{k=0}^{s} \left(\left(1 - \frac{\lambda h^3 \rho(\mathbf{X}_l)}{\rho(\mathbf{x}_{i,j})} \sum_r \sum_{i,j} D_{i,j}^l D_{i,j}^r \Delta s_r \right)^k \right) \Delta Q^0(\mathbf{X}_l). \tag{2.53}$$

2.4 IBM for the thermal Neumann boundary condition

The works of treating the Neumann boundary condition in IBM are far fewer than the counterpart which involves the Dirichlet-type boundary conditions. In our work, we propose a novel implementation method of the Neumann boundary condition in IBM which based on the mathematical manipulation. At first, we give the two following lemmas.

Lemma 2.1. *As shown in Fig.* 3, *based on the Eq.* (2.10), *the following equation holds*

$$\delta Q = -\kappa \left[\frac{\partial T}{\partial n} \right], \tag{2.54}$$

$$\left[\frac{\partial T}{\partial \tau} \right] = 0, \tag{2.55}$$

where $\boldsymbol{n} = (n_x, n_y), \boldsymbol{\tau} = (-n_y, n_x)$ *is the unit outward normal vector and unit tangent vector respectively, and* $[\cdot]$ *is the jump function, it can be expressed as*

$$[f(\boldsymbol{X})] = \lim_{\varepsilon->0^+} f(\boldsymbol{X}+\varepsilon\boldsymbol{b}) - \lim_{\varepsilon->0^-} f(\boldsymbol{X}-\varepsilon\boldsymbol{b}), \tag{2.56}$$

where $\boldsymbol{b}$ *is the direction vector.*

The above lemma can be proved using the Green's theorem and the properties of the Dirac function (more details, see Appendix A).

Lemma 2.2. *As shown in Fig.* 3, *the following equations hold*

$$\sum_{(i,j)\in S_l} A_{ij}^l \frac{T_{i+1,j} - T_{i-1,j}}{2h} = \alpha_1 T_x^+(\boldsymbol{X}_l) + (1-\alpha_1) T_x^-(\boldsymbol{X}_l) + \beta_1 [T_y(\boldsymbol{X}_l)], \tag{2.57}$$

$$\sum_{(i,j)\in S_l} A_{ij}^l \frac{T_{i,j+1} - T_{i,j-1}}{2h} = \alpha_2 [T_x(\boldsymbol{X}_l)] + \beta_2 T_y^+(\boldsymbol{X}_l) + (1-\beta_2) T_y^-(\boldsymbol{X}_l), \tag{2.58}$$

where $A_{i,j}^l$ *is the second-order interpolation coefficient,* S_l *is the a second-order interpolation stencil, and the coefficients* $\alpha_1, \beta_1, \alpha_2, \beta_2$ *can be obtained in Appendix B.*

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press

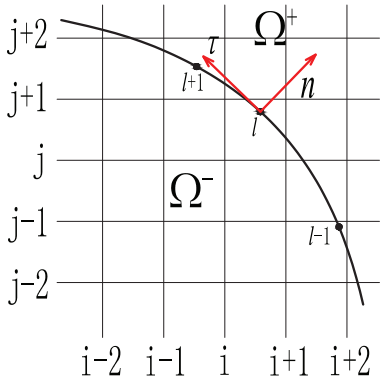


Figure 3: A diagram of the local grid near the immersed boundary.

Here $A_{i,j}$,S_l are selected as

$$A_{i,j}=D_{i,j}, \tag{2.59}$$
$$S_l=\{(i,j)\,|\,|\mathbf{x}_{i,j}-\mathbf{X}_l|\le 2h\}. \tag{2.60}$$

Eqs. (2.57) and (2.58) can be viewed as generalized interpolation formulas of central difference which hold in the non-smooth domain (for more details, see Appendix B).

Following Lemma 2.2, the equations hold

$$\sum_{(i,j)\in S_l} A_{ij}^l \frac{T_{i+1,j}^n-T_{i-1,j}^n}{2h}=\alpha_1 T_x^{n+}+(1-\alpha_1)T_x^{n-}+\beta_1[T_y^n], \tag{2.61}$$

$$\sum_{(i,j)\in S_l} A_{ij}^l \frac{T_{i,j+1}^n-T_{i,j-1}^n}{2h}=\alpha_2[T_x^n]+\beta_2 T_y^{n+}+(1-\beta_2)T_y^{n-}. \tag{2.62}$$

Without loss of generality, if $n_x,n_y\ge 0$, we have

$$(\alpha_1-1)[T_x^n]+\beta_1[T_y^n]=\sum_{(i,j)\in S_l} A_{ij}^l \frac{T_{i+1,j}^n-T_{i-1,j}^n}{2h}-T_x^{n+}, \tag{2.63}$$

$$\alpha_2[T_x^n]+(\beta_2-1)[T_y^n]=\sum_{(i,j)\in S_l} A_{ij}^l \frac{T_{i,j+1}^n-T_{i,j-1}^n}{2h}-T_y^{n+}. \tag{2.64}$$

The heat flux $\delta Q_l(=\delta Q(\mathbf{X}_l))$ can be expressed as

$$\delta Q_l=-\kappa\left[\frac{\partial T}{\partial n}\right]=-\kappa([T_x^n]n_x+[T_y^n]n_y). \tag{2.65}$$

And based on Eqs. (2.6) and (2.55), we have

$$-\kappa(T_x^{n+}n_x+T_y^{n+}n_y)=Q_B, \tag{2.66}$$
$$-[T_x^n]n_y+[T_y^n]n_x=0. \tag{2.67}$$

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press

In the framework of the IBM, the jump can be treated by a smoothing technique. In this study, we replace singular function $\delta(\mathbf{x}-\mathbf{X})$ in Eq. (2.10) by a smoothed Dirac delta function $\delta_h(\mathbf{x}-\mathbf{X})$. The form of the smoothed function in Eq. (2.32) is selected. If we note

$$T^*=\sum_{\alpha} g_{\alpha}, \tag{2.68}$$

the temperature field can be corrected by

$$T_{i,j}^{n+1}=T_{i,j}^{*}+\frac{1}{2}\sum_{l}\delta Q_l D_{i,j}^{l}\Delta s_l. \tag{2.69}$$

In summary, the Eqs. (2.63), (2.64), (2.65), (2.66) and (2.67) form a linear equation set. And its solution can be expressed as

$$\delta Q_l=-\frac{\kappa\sum_{(i,j)\in S_l}A_{ij}^{l}\frac{T_{i+1,j}^{n}-T_{i-1,j}^{n}}{2h}n_x+\kappa\sum_{(i,j)\in S_l}A_{ij}^{l}\frac{T_{i,j+1}^{n}-T_{i,j-1}^{n}}{2h}n_y+Q_B}{(\alpha_1-1)n_x^2+(\beta_1+\alpha_2)n_xn_y+(\beta_2-1)n_y^2}. \tag{2.70}$$

Once the heat flux on the boundary has been obtained, the temperature field can be corrected using the Eq. (2.69).

2.5 IBM for the thermal Robin (mixed) boundary condition

As similar as the treatment of the thermal Neumann boundary condition, the heat flux Q_l can be obtained by solving a linear system. In this case, instead of the Eq. (2.66), the following equations are added.

$$\kappa(T_x^{n+}n_x+T_y^{n+}n_y)=h_e(T_l-T_B), \tag{2.71}$$

$$\sum A_{i,j}^{l}T_{i,j}^{n}=T_l. \tag{2.72}$$

As a result, we have

$$\delta Q_l=-\frac{\kappa\sum_{(i,j)\in S_l}A_{ij}^{l}\frac{T_{i+1,j}^{n}-T_{i-1,j}^{n}}{2h}n_x+\kappa\sum_{(i,j)\in S_l}A_{ij}^{l}\frac{T_{i,j+1}^{n}-T_{i,j-1}^{n}}{2h}n_y+h_e(T_B-\sum A_{i,j}^{l}T_{i,j}^{n})}{(\alpha_1-1)n_x^2+(\beta_1+\alpha_2)n_xn_y+(\beta_2-1)n_y^2}. \tag{2.73}$$

Similarly, the temperature field can be corrected using Eq. (2.69).
In summary, a computational procedure can be outlined below.

(1) Compute the discrete force term and heat source/sink term using Eqs. (2.19) and (2.20)

(2) Compute the post-collision state of density and temperature distribution functions using Eqs. (2.11) and (2.12).

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press

(3) Compute the intermediate velocity and temperature using Eqs. (2.30), (2.45) or (2.68).

(4) Correct the velocity field using the iterative procedure which described in Section 2.2.

(5) Correct the temperature field using the iterative procedure which described in Section 2.3 or solving the linear equations in Section 2.4 or 2.5.

(6) Repeat Steps 1-5.

3 Numerical examples

To validate the capacity of the present method, some numerical examples are simulated.

3.1 Accuracy Verification

To our best knowledge, there are some different viewpoints on the accuracy of the diffuse-interface IBM. Some researchers believe that the diffuse-interface IBM is only first-order accuracy in space. However the second order convergence rate has been observed in other studies. In the present work, we first test the diffuse-interface IBM for the velocity boundary condition using two different examples: Taylor-Green vortices and rotation flow problems. Taylor-Green vortices problem is used in many previous studies to test the accuracy of IBM [22, 46–54]. Its analytical solution can be expressed as

$$u(x,y,t) = -U\cos(\pi x)\sin(\pi y)e^{-2\pi^2 t/Re}, \tag{3.1}$$

$$v(x,y,t) = U\sin(\pi x)\cos(\pi y)e^{-2\pi^2 t/Re}, \tag{3.2}$$

$$\rho(x,y,t) = \rho_0 - \rho_0\frac{U^2}{4c_s^2}[\cos(2\pi x)+\cos(2\pi y)]e^{-2\pi^2 t/Re}. \tag{3.3}$$

In this section, as shown in Fig. 4, the size of the computational domain is set as $[0,2]\times[0,2]$, and a circle with a diameter of 1 is embedded at the center. Basically, three factors have influence on the accuracy of the solution: the treatment of the outer boundary condition, the accuracy of the lattice Boltzmann scheme (2.11) and the treatment of the inner boundary condition using the IBM. In the following study, when only lattice Boltzmann scheme is used, both the computational domain and the solution domain are $\Omega = \Omega_1 \bigcup \Omega_2$. When the inner boundary condition is considered, the present IBM is used. In this case, the computational domain is also $\Omega = \Omega_1 \bigcup \Omega_2$. However, Ω_1 is the solution domain and Ω_2 is the continuation computational domain.

For the Taylor-Green vortices problem, the Reynolds number and the dimensionless relaxation time are set as $Re = 10, \tau = 0.65$ respectively, which are same as the previous works. Four different uniform meshes with mesh spacing of $h = 1/10$, $1/20$, $1/40$, $1/80$

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press

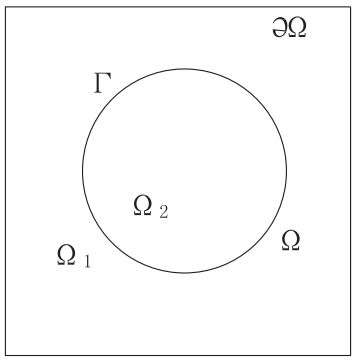


Figure 4: The computational domain for accuracy verification.

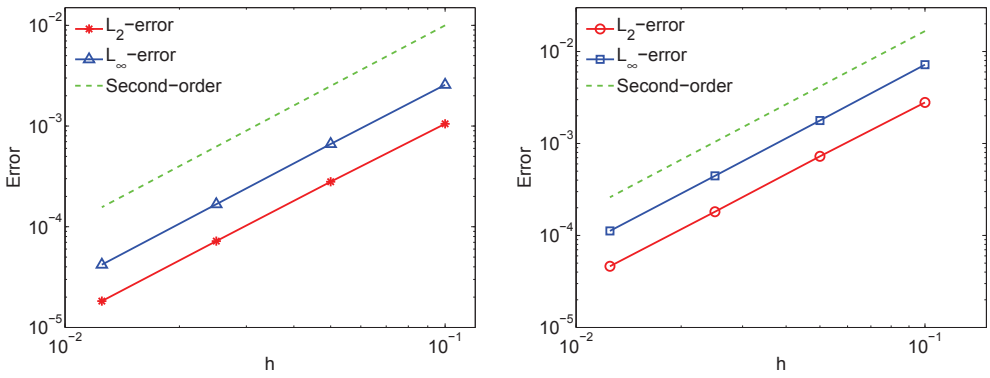
are used. The lengths of Lagrangian elements are sets as Δs=0.7h in all cases. We compare the overall error of u using the L_2-norm and $L_\infty-norm$, which are calculated by

$$L_2-error=\sqrt{\frac{\sum(u^n-u^a)^2}{N}}, \tag{3.4}$$

$$L_\infty-error=max|u^n-u^a|, \tag{3.5}$$

where u^n,u^a denote the numerical and analytical solutions, and N is the number of the Eulerian nodes of the whole computational domain.

Fig. 5 plots the L_2-error and L_∞-error for the Taylor Green vortices problem at t=1. As shown in Fig. 5(a), the accuracy of solution when IBM is not used is almost second-order. It indicates that the lattice Boltzmann scheme is second-order accuracy. It is consistent with the previous studies. When the inner boundary condition is considered, the present iterative IBM is adopted. In Fig. 5(b), we can observed that the present iterative IBM is



(a) LBM (b) IB-LBM

Figure 5: Grid convergence of the present method for Taylor Green vortex problem.

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press

second-order accurate in maximum and L_2 norms. It seems that the present iterative IBM has second-order accuracy.

Moreover, we also select the rotation flow as another test example. The computational domain is same as the above example. The exact steady solution which is defined in Ω_1 can be expressed as

$$u_\theta(r,\theta) = Ar + \frac{B}{r}, \tag{3.6}$$

$$u_r(r,\theta) = 0, \tag{3.7}$$

where u_θ, u_r are components of the velocity in cylindrical coordinates, and A, B are defined as

$$A = \frac{R_2U_2 - R_1U_1}{R_2^2 - R_1^2}, \tag{3.8}$$

$$B = \frac{R_1R_2^2U_1 - R_1^2R_2U_2}{R_2^2 - R_1^2}, \tag{3.9}$$

where $R_1 = 1$ is the radius of the inner cylinder, and $R_2 = 2\sqrt{2}$, $U_1 = 0.1$, $U_2 = 0.02$. The velocities on the outer boundaries are assigned using the exact solution.

We first simulate the case without considering the inner boundary condition. However, the exact steady solution is unknown. So we apply the following norms to evaluate the errors [56]

$$e_p[u;N] = ||u^N - I_{2N}^N u^{2N}||_p, \tag{3.10}$$

where $p = 2,\infty$ mean the L_2 and L_∞ norms respectively. N is the number of grid points in the x or y directions. I_{2N}^N is the interpolation operator from $2N \times 2N$ Cartesian grid to $N \times N$ Cartesian grid. In this study, $N = 20,40,80,160,320$ are selected.

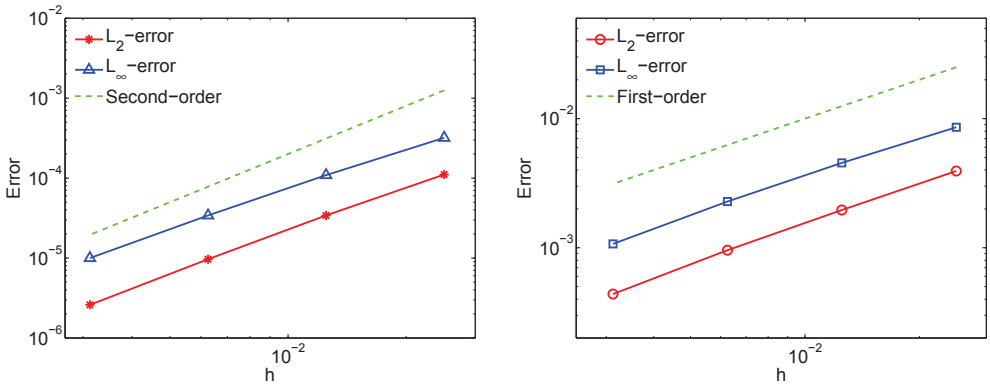
The convergence criterion for steady solution is set to be:

$$\frac{\sum|T^{n+1} - T^n|}{\sum T^{n+1}} \leq 10^{-7}. \tag{3.11}$$

As shown in Fig. 6(a), when only the outer boundaries conditions are considered, the convergence rate of the L_2 and L_∞ errors are 1.81 and 1.67 respectively. However, in Fig. 6(b), only first-order accuracy is observed with considering the inner boundary condition. It indicates that the IBM has only first-order accuracy. This conclusion contradicts with the result of the above test case.

The basic reason of phenomenon is the difference of smoothness near the boundary. For Taylor Green vortex problem, We notice that the expression of exact solution in the continuation domain Ω_2 is same as the expression of exact solution in the solution domain Ω_1. So the velocity field is sufficiently smooth in the local domain near the

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press



(a) LBM (b) IB-LBM

Figure 6: Grid convergence of the present method for rotation flow problem.

boundary. Obviously, we have

$$\left[\frac{\partial u}{\partial x}\right]\Big|_{r=1}=0, \tag{3.12}$$

$$\left[\frac{\partial u}{\partial y}\right]\Big|_{r=1}=0. \tag{3.13}$$

However, for Rotation flow problem, the expression of exact solution in the continuation domain Ω_2 can be described as follow.

$$u_\theta(r,\theta)=U_1 r, \tag{3.14}$$

$$u_r(r,\theta)=0, \tag{3.15}$$

Obviously, simple computational result shows

$$\left[\frac{\partial u}{\partial x}\right]\Big|_{r=1}\neq 0, \tag{3.16}$$

$$\left[\frac{\partial u}{\partial y}\right]\Big|_{r=1}\neq 0. \tag{3.17}$$

It indicates that the velocity field is lack of sufficient smoothness near the boundary in the rotation flow problem. In Ref. [57], based on the one-dimensional model, Beyer *et al*. analyzed the accuracy of IBM. Their result indicates that lack of smoothness leads to loss of accuracy. Although the case of two-dimension is more complex than one-dimension case, it still tells us that the relations in Eqs. (3.12), (3.16) lead to different conclusions. For sufficiently smooth case, the Eqs. (2.31) and (2.35) have also second-order accuracy. Thus second order accuracy can be reached when applies the Taylor-Green vortices problem as the test example. Our result is consistent with the analysis of the accuracy of direct forcing IBMs with projection methods by Guy *et al*. [58]. The Taylor Green vortex problem,

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press

as a test example to study the accuracy of IBM by many researchers, can not get general conclusion.

Based on the understanding of the above statement, we select the test examples for thermal IBM which are similar to the rotation flow problem. The heat diffusion problem is considered to test accuracy of the thermal IBM. A two-dimensional square domain with a size of $[0,2]\times[0,2]$ which is embed a circle at the center is chosen as computational domain. Here the diameter of the circle is 1. The boundary conditions on the inner cylinder are set as the Dirichlet, Neumann and Robin types, respectively. The dimensionless boundaries conditions on inner circle are described as follow

$$T=1, \tag{3.18}$$

$$\frac{\partial T}{\partial n}=-1, \tag{3.19}$$

$$T-\frac{\partial T}{\partial n}=1. \tag{3.20}$$

The steady solutions with different boundary conditions are only radius functions, and these can are expressed as

$$T(r)=\begin{cases} -\dfrac{2\ln r}{3\ln 2}+1, & T=1, \\ \dfrac{3}{2}\ln 2-\ln r, & \dfrac{\partial T}{\partial n}=-1, \\ \dfrac{3\ln 2-2\ln r}{3\ln 2+2}, & T-\dfrac{\partial T}{\partial n}=1. \end{cases} \tag{3.21}$$

The above exact solutions also provide the velocity values on the outer boundary. In our simulation, the L_2 and L_∞ norms are selected to study the grid convergence of the present method.

The temperature relaxation parameter is fixed at $\tau_g=0.8$. As shown in Figs. 7, 8 and 9, the L_2 and L_∞ errors are almost first-order convergence rate in all cases.

3.2 Natural convection in a cavity with a circular cylinder in the center

In order to verify the capability of the present IBM for the Dirichlet temperature boundary condition, the natural convection in an eccentric annulus between a square outer cylinder and circular cylinder is studied. This problem has been simulated by various numerical methods [59, 60]. A schematic view of entire computational domain about this problem is shown in Fig. 10. The wall of enclosed square cavity satisfies low-temperature boundary condition, and high-temperature is imposed on the inner circular cylinder boundary. Certainly, all walls satisfy no-slip boundary condition. Based on the mechanism of natural convection, it can be observed a series of complex flow phenomena. To determine the geometric configuration of natural convection problems uniquely, we must select the geometric parameters of this problem, including the radius r of the circular cylinder, the

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press

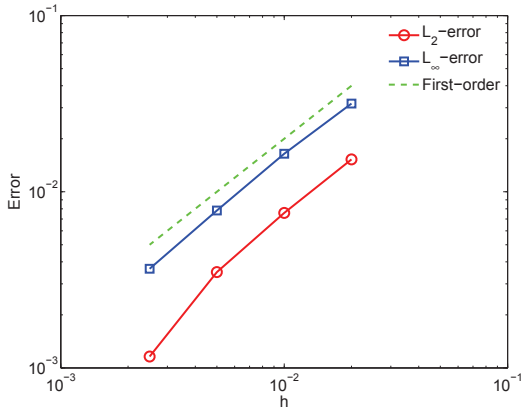


Figure 7: Grid convergence of the present method for the Dirichlet boundary condition.

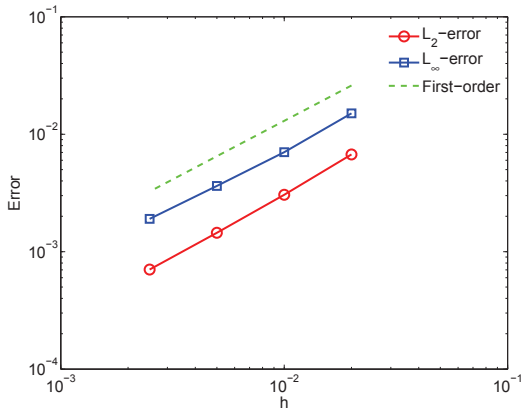


Figure 8: Grid convergence of the present method for the Neumann boundary condition.

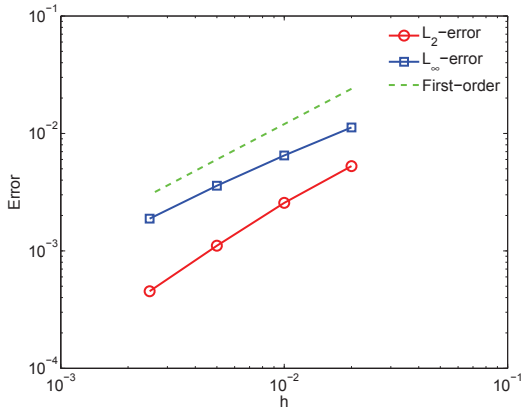


Figure 9: Grid convergence of the present method for the Robin boundary condition.

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press

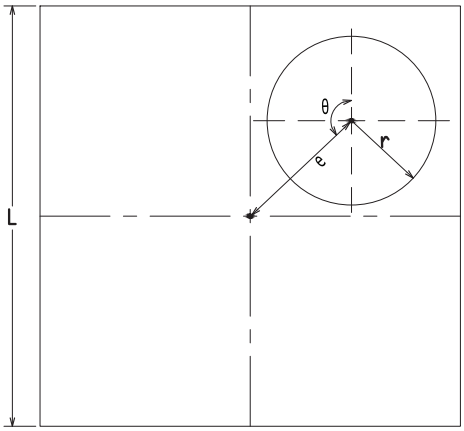


Figure 10: A schematic view of an eccentric annulus between a square outer cylinder and circular cylinder.

sides of length L, the distance between the center of the circular cylinder and the center of the square cylinder e, and the angular direction θ, the aspect ratio $Ar=L/(2r)$ and eccentricity $\varepsilon=L/e$. In the present work, Ar is fixed at 2.6 in all cases, and the different eccentricity ε and angular position θ are considered. In addition to the geometric parameters, the Prandtl number and Rayleigh number are fixed at $Pr=0.7$ and $Ra=3\times10^5$, respectively. The Rayleigh number and Prandtl number are defined as

$$Ra=\frac{\rho c_p g\beta L_c^3(T_h-T_c)}{\kappa\mu},\qquad Pr=\frac{c_p\mu}{\kappa},\tag{3.22}$$

where $L_c=L/2-r$ is the reference length, and T_h,T_c denote the high-temperature and low-temperature, respectively. Temperature can be normalized by

$$\tilde{T}=\frac{T-T_c}{T_h-T_c}.\tag{3.23}$$

For the thermal flows, the Nusselt number Nu is an important dimensionless parameter, and the local Nusselt number is given by

$$Nu(\mathbf{X},t)=-\frac{\partial\tilde{T}}{\partial n}.\tag{3.24}$$

To test the effect of the number of iterations, relaxation parameter on the solutions, the case with $\theta=\pi/4$ and $\varepsilon=0.25$ is selected to simulate. 250×250 grid is used in this numerical example. In Table 2, we investigate the effects of the relaxation parameters and the number of iterations on the velocity_errors and temperature_errors. The velocity_error

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press

Table 1: Comparison of average Nusselt number.

Cases		References		
θ	ε	Present	Shu *et al*. [59]	Ding *et al*. [60]
0	0.25	6.7275	6.81	6.74
	0.50	6.8974	6.93	6.92
	0.75	7.6321	7.65	7.63
$\frac{\pi}{4}$	0.25	6.6853	6.64	6.64
	0.50	6.6228	6.63	6.68
	0.75	6.7749	6.78	6.78
	0.95	7.3313	7.30	7.29
$\frac{\pi}{2}$	0.25	6.4611	6.47	6.48
	0.50	6.4239	6.56	6.42
	0.75	7.0755	7.04	7.03
	0.95	11.5063	11.37	-
$\frac{3\pi}{4}$	0.25	6.2672	6.29	6.29
	0.50	6.0001	6.04	6.01
	0.75	5.9715	5.99	5.96
	0.95	6.4150	6.40	6.36
π	0.25	6.7114	6.72	6.74
	0.50	6.1387	6.14	6.15
	0.75	6.6458	6.62	6.62

Table 2: The effects of the relaxation parameter and the number of iterations on velocity_errors and temperature_errors.

Iterative parameters	velocity_error	temperature_error
$s=1$(Non-iterative)	2.8140×10^{-3}	2.6914×10^{-3}
$\lambda=1,s=5$	1.2425×10^{-4}	8.8147×10^{-5}
$\lambda=2.5,s=5$	1.8098×10^{-6}	9.3555×10^{-7}

and temperature_error are defined as

$$velocity_error=\sqrt{\frac{||\mathbf{U}^n-\mathbf{U}^a||_2^2}{M}}, \tag{3.25}$$

$$temperature_error=\sqrt{\frac{||T^n-T^a||_2^2}{M}}, \tag{3.26}$$

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press

Table 3: The effects of the relaxation parameter and the number of iterations on the average Nusselt number.

Sources	$s=1$	$\lambda=1,s=5$	$\lambda=2.5,s=5$	Shu et al. [59]	Ding et al. [60]
$\bar{Nu}$	7.0865	6.7014	6.6853	6.64	6.64

Table 4: Comparison of CPU time cost with different relaxation parameters and the numbers of iterations.

Sources	$s=1$	$\lambda=1,s=5$	$\lambda=2.5,s=5$
CPU times cost by LBM(s)	2258.6	2309.4	2258.6
CPU times cost by IBM(s)	2490.2	2561.4	2490.2
CPU times cost by others(s)	315.8	327.0	326.5

where n,a denote numerical and analytical values, respectively. Obviously, the boundary errors which are obtained by the iterative method are less than the results which are computed using the non-iterative method. Moreover, With the approximate relaxation parameter, the boundary errors can be reduced to get more optimal results.

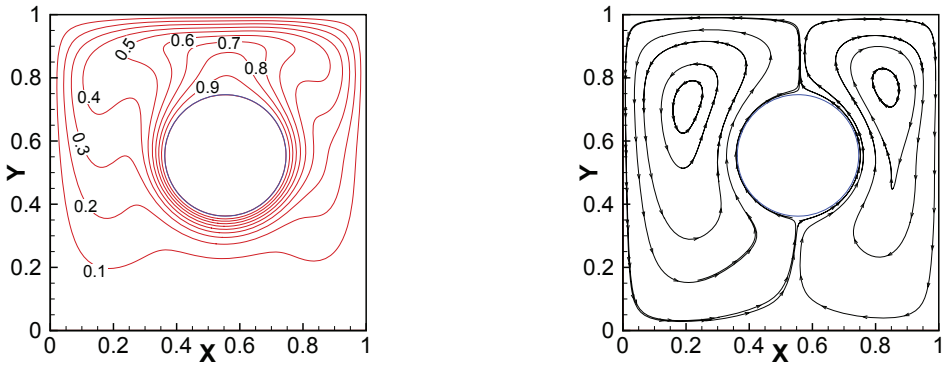
In Table 3, the effects of the relaxation parameter and the number of iterations on the average Nusselt number are investigated. Compared with the non-iterative method, we notice that $\bar{Nu}$ which is obtained using the iterative method are closer to the results of Shu *et al*. and Ding *et al*. [59, 60]. Similarly, when the relaxation parameter λ is set as 2.5, the relative errors of $\bar{Nu}$ between the present results and the results of Shu *et al*. and Ding *et al*. are within 0.7%.

To evaluate the efficiency of the present iterative IBM, we compare the computational time for the above test case. The simulations are implemented on the Intel CPU(Core(TM) i7-4770, 3.40GHz). It can been observed that the computational costs for the iterative method and non-iterative are almost same. In other words, the iterative method is no additional computation overhead.

In this simulation, 18 subcases are investigated. The initial setup of the velocity and temperature are set as $\mathbf{u}=\mathbf{0}$, $T=T_c$. This problem has been studied by Shu *et al.* [59]. and Ding *et al*. [60]. As indicated in Table 1, the average Nusselt numbers are compared between the present and previous results. It is clearly seen that our results are in good agreement with the data of Shu *et al*. and Ding *et al*. We can draw a conclusion that the heat transfer become more intensely when the cylinder is closer to the wall. We find that almost mean Nusselt numbers are in the interval $[5.9,7.7]$. However the average Nusselt number of the case with $\theta=\frac{\pi}{2}$, $\varepsilon=0.95$ increases to 11.5.

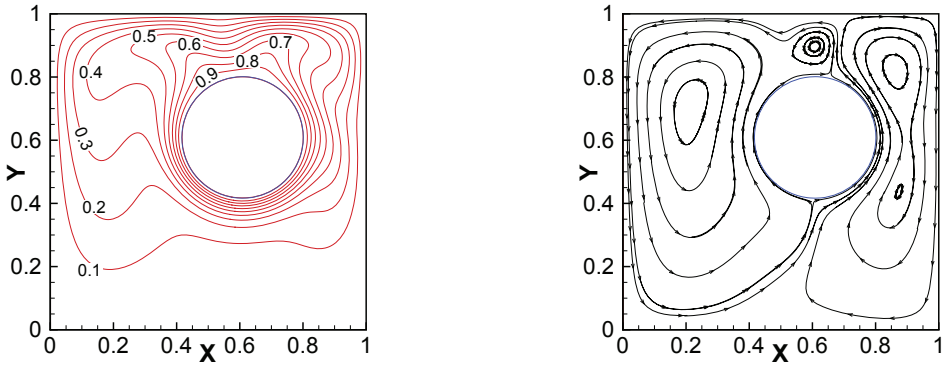
To describe the details of velocity and temperature fields, the isotherms and streamlines in the cases with $\theta=3\pi/4$ and $\varepsilon=0.25,0.5,0.75,0.95$ are plotted in Figs. 11, 12, 13 and 14. When the ε increases, the eddy on the right splits into two parts. Particularly, a secondary eddy appears when $\varepsilon=0.5$. The present method captures this flow detail successfully. Moreover, as the eccentricity increases, the temperature boundary layer on the top cylinder becomes thinner since the limit of the space.

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press



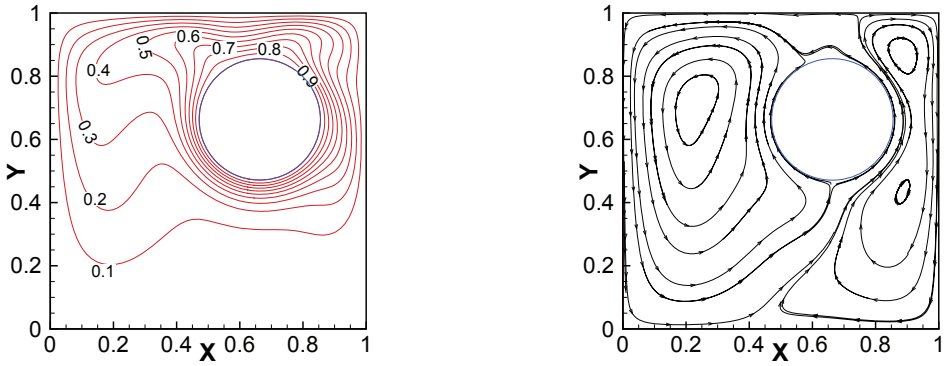
(a) (b)

Figure 11: The isotherms (a) and streamlines (b) for $\theta=3\pi/4$, $\varepsilon=0.25$.



(a) (b)

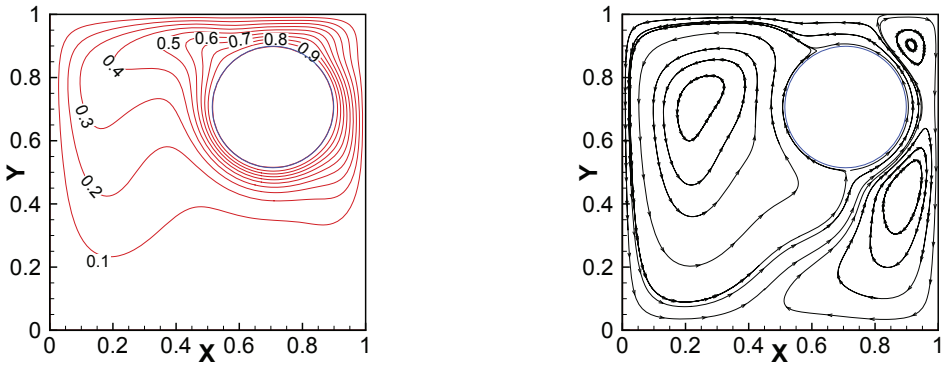
Figure 12: The isotherms (a) and streamlines (b) for $\theta=3\pi/4$, $\varepsilon=0.50$.



(a) (b)

Figure 13: The isotherms (a) and streamlines (b) for $\theta=3\pi/4$, $\varepsilon=0.75$.

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press



(a) (b)

Figure 14: The isotherms (a) and streamlines (b) for $\theta=3\pi/4$, $\varepsilon=0.95$.

3.3 Heat convection with flow over a cylinder

3.3.1 Flow around a stationary cylinder

The present method is also used to investigate the problem of flows around a stationary cylinder. It is a classical problem and many researchers had investigated it. For this problem, only velocity field can affect temperature field while the temperature field has no effect on the velocity field. So the velocity and temperature fields are decouple. The key parameters of this forced convection problem are the Reynolds number Re and the Prandtl number Pr. The Reynolds number is defined as

$$Re=\frac{\rho UD}{\mu}, \tag{3.27}$$

where U is the free stream velocity and D is diameter of the cylinder. The definition of the Prandtl number is same as the Section 3.2.

A rectangular with a size of $32D\times 18D$ is chosen as computational domain. The center of the cylinder is located at origin coordinate and the inlet boundary is located at $x=-6D$. In this study, the cases with three kinds of temperature boundary conditions are simulated. For the first and third types boundary conditions, the non-dimensionalization procedure are same as that in the Section 3.2. For the second type boundary condition, the temperature is normalized by

$$\tilde{T}=\frac{T-T_c}{Q_BL/\kappa}, \tag{3.28}$$

and the local Nusselt number on the boundary can be obtained by

$$Nu=\frac{1}{\tilde{T}}. \tag{3.29}$$

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press

Moreover, the dimensionless form of the Robin boundary condition can be expressed as follow

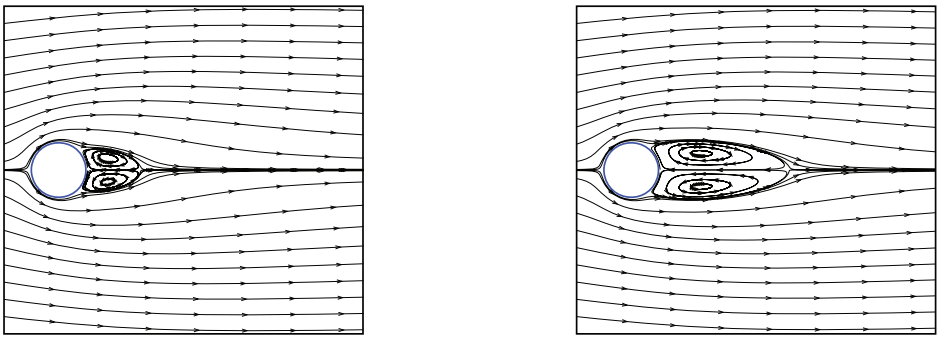
$$Bi\tilde{T}-\frac{\partial\tilde{T}}{\partial n}=Bi, \tag{3.30}$$

where $Bi=h_eD/\kappa$ is a parameter which is related to the Biot number, and the local Nusselt number on the cylinder surface can be calculated by

$$Nu=Bi\Big(\frac{1}{\tilde{T}}-1\Big). \tag{3.31}$$

At the inlet, a free stream velocity $U=0.1$ is given. The non-equilibrium bounce-back scheme is used together with the top and bottom boundary conditions, and the boundary at the outflow is implemented using the extrapolation scheme. The temperature boundary conditions are implemented as same as the velocity boundary conditions on the outlet boundary. The temperature boundary conditions on the cylinder surface are set as the constant wall temperature (Isothermal), uniform heat flux (Isoflux) and the Robin boundary condition with uniform Bi, respectively.

Several different values of the Reynolds number are selected in our study. Both steady and unsteady cases are simulated. As shown in Fig. 15, the streamlines at $Re=20$,40 are plotted. Two symmetric vortices behind the cylinder can be observed clearly. For unsteady flow, an instantaneous streamlines and the vortex structures at $Re=200$ are plotted in Fig. 16. The Karman vortex streets occur in this case. In unsteady-flow analysis, the flow and temperature fields change periodicity, so the time-averaged Nusselt number is used in the following. As shown in Tables 5, 6 and 7, we compare the average Nusselt numbers for the cases with three types of boundary conditions. Table 5 also lists the mean Nusselt numbers for different grid sizes. The present results are consistent with



$Re=20$ $Re=40$

Figure 15: The streamlines at $Re=20$ and 40.

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press

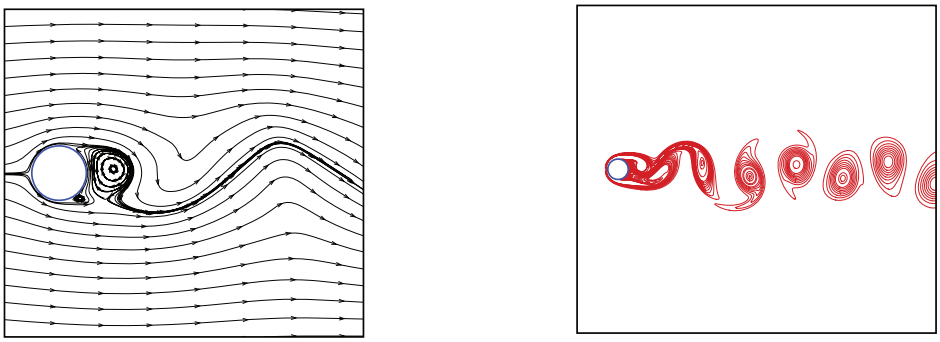


Figure 16: Instantaneous streamlines and vorticity contours at $Re=200$.

Table 5: Comparison of average Nusselt number with the previous studies with constant wall temperature boundary condition.

References	Grid size (h)	$Re=10$	$Re=20$	$Re=40$
Present	1/20	1.9037	2.5217	3.3456
	1/30	1.8867	2.5101	3.3313
	1/40	1.8736	2.5014	3.3220
Bharti *et al*. [61]	-	1.8623	2.4653	3.2825
Sparrow *et al*. [62]	-	1.6026	2.2051	3.0821
Ren *et al*. [44]	-	1.9150	2.5238	3.3519
Haeri *et al*. [4]	-	1.80	2.38	3.15

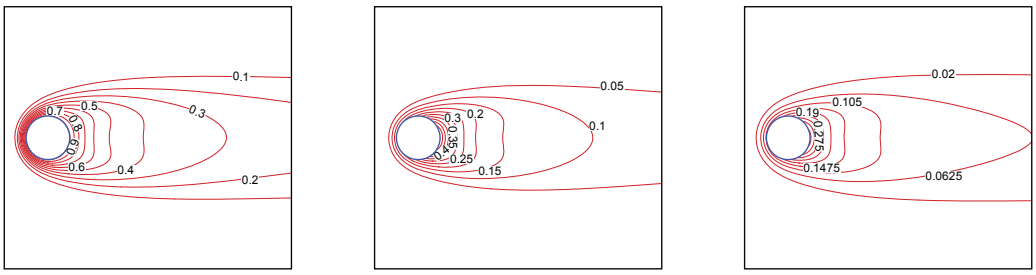
Table 6: Comparison of the average Nusselt number with the previous studies with uniform heat flux boundary condition.

References	$Re=10$	$Re=20$	$Re=40$
Present	2.0269	2.7318	3.6449
Bharti *et al*. [61]	2.0400	2.7788	3.7755
Ahmad *et al*. [63]	2.0410	2.6620	3.4720
Ren *et al*. [44]	2.0265	2.7413	3.7407

Table 7: Comparison of average Nusselt number with the previous studies under the Robin boundary condition with $Bi=1$.

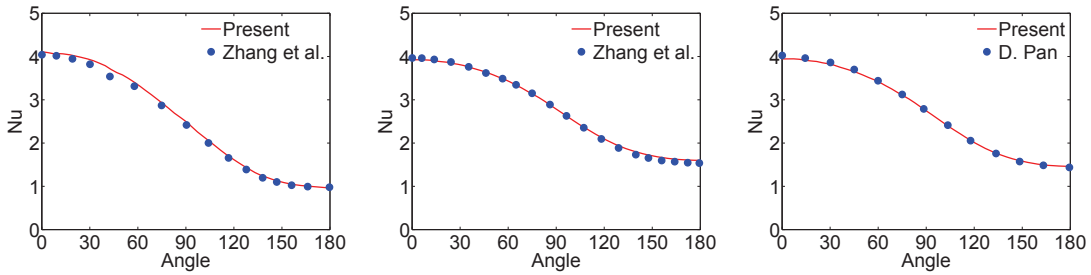
References	$Re=20$	$Re=40$	$Re=100$	$Re=200$
Present	2.7113	3.6735	5.7807	8.1983
Pan *et al*. [64]	2.7202	3.7078	5.9255	8.9353

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press



(a) (b) (c)

Figure 17: Temperature contours for $Re=40$: (a) Constant wall temperature boundary condition; (b) Uniform heat flux boundary condition; (c) Robin boundary condition with $Bi=1$.



(a) (b) (c)

Figure 18: Comparison of the local Nusselt number distribution on the cylinder surface with $Re=20$: (a) Constant wall temperature boundary condition; (b) Uniform heat flux boundary condition; (c) Robin boundary condition with $Bi=1$.

the previous studies. Fig. 17 shows the steady-state isotherms at $Re=40$ with the three types boundary conditions. To obtain more details, in Fig. 18, the local Nusselt number on the cylinder surface at $Re=20$ under three types of boundary conditions are compared with the results in [40] and [64]. In the cases of flows over a cylinder, the angle θ is defined in the clockwise sense starting from the front stagnation point. Our results are very good agreement with the data of Zhang *et al*. [40] and Pan [64]. In the three cases, the Nusselt number distributions are monotonous. The maximum Nusselt numbers of three cases are located at the front stagnation point of the cylinder. The minimal values of Nusselt number are located at the rear stagnation point. In addition to the small region near the front stagnation point, the Nusselt numbers in the Dirichlet boundary condition are smaller than the counterpart with the Neumann and Robin boundary conditions. Furthermore, Fig. 19 plots the temperature distribution on the cylinder surface with the Robin boundary condition at $Bi=3,5$. We find the temperature are higher at the high Bi. This is because the heat transfer resistances of the cylinder surface is smaller with the high Bi, and the temperature on the cylinder surface is more closer to T_h. In Fig. 20, the

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press

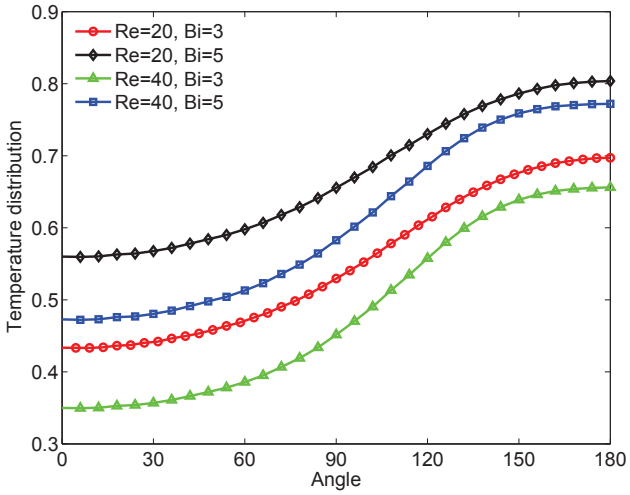


Figure 19: Temperature distributions on the cylinder surface with $Re=20,40$ and $Bi=3,5$.

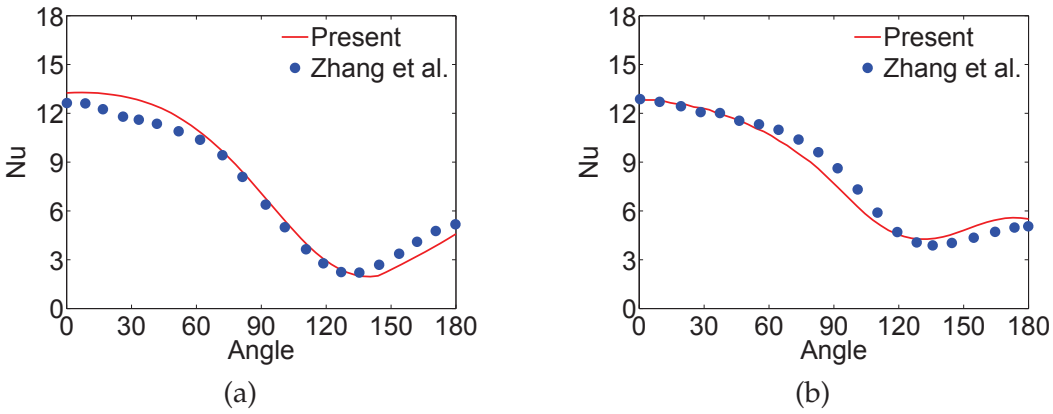


Figure 20: Comparison of the time-average local Nusselt number on the cylinder surface at $Re=200$: (a) Constant temperature boundary condition; (b) Uniform heat flux boundary condition.

present results of the time-average Nusselt numbers under the Dirichlet and Neumann boundary conditions at $Re=200$ are compared with the data of Zhang *et al*. [40]. Different from the cases at the low Reynolds numbers, the minimum Nusselt numbers in both cases do not appear at the rear stagnation point, and it occurs at the angle interval $[130^o,140^o]$. Moreover, the temperature distributions on the cylinder surface with the Robin boundary condition at $Bi=3,5$ are plotted in Fig. 21. We can also see the inflection point at the angle interval $[130^o,140^o]$. It indicates that the flow is more complex at the high Reynolds number.

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press

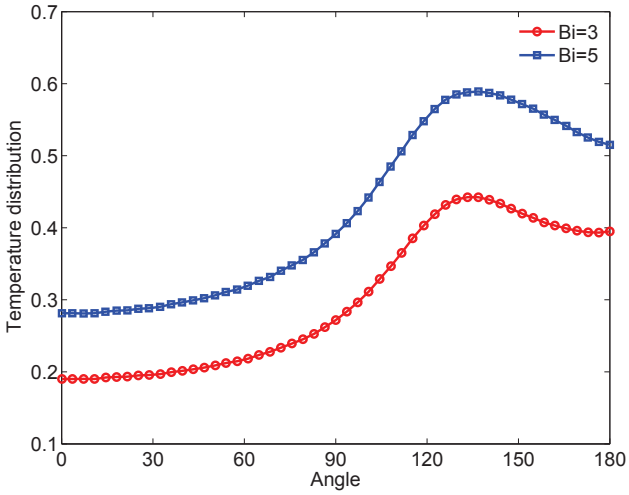


Figure 21: The temperature distributions on the cylinder surface under the Robin boundary condition at Bi=3,5.

3.3.2 Flow over an oscillating cylinder

In this section, the flow over an oscillating cylinder is considered. Compared with the stationary case, the position of the cylinder is changed. However, in the framework of IBM, we only track the position of the moving boundary. So it adds very little computational overhead. The dimensionless equation of motion of the oscillating cylinder in the cross-wise direction can be written as

$$d_y(t) = A\sin(2\pi ft), \tag{3.32}$$

where A is the dimensionless oscillating amplitude, f is the dimensionless oscillating frequency. Same as the case in Ref. [49], $A = 0.15$ is selected, and the frequency is set as $f = 0.2$ which is close to the vortex shedding frequency of the stationary cylinder at $Re = 200$.

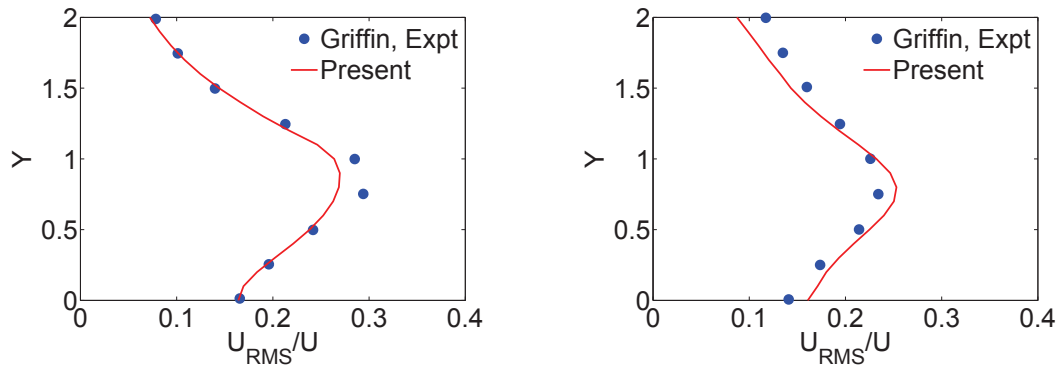
In the present work, the Reynolds number and Prandtl number are set as $Re = 200$, $Pr = 0.7$, respectively. Like Section 3.3.1, three kinds of temperature boundary conditions on the cylinder surface are investigated. Firstly we verify the fidelity of flow field which compares with the experiment data of Griffin [66]. We use the root-mean-square (RMS) velocity-magnitude fluctuations as the representative of the flow quantity. It can be calculated by

$$\bar{U} = \frac{\sum_i \sqrt{u^2(t_i)+v^2(t_i)}}{N}, \tag{3.33}$$

$$U_{RMS} = \frac{\sum_i (\sqrt{u^2(t_i)+v^2(t_i)}-\bar{U})^2}{N}, \tag{3.34}$$

where N is the number of the time sampling.

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press



(a) $x=2.5$ (b) $x=5.4$

Figure 22: Comparison RMS velocity-magnitude fluctuation profiles: (a) $x=2.5$; (b) $x=5.4$.

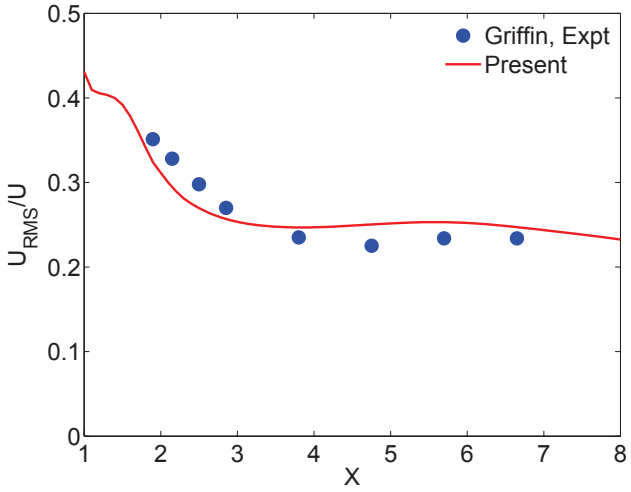


Figure 23: Comparison of maximum RMS velocity-magnitude fluctuations as a function of stream-wise locations x.

As shown in Fig. 22, the RMS velocity magnitude fluctuation profiles at two different stream-wise locations are compared with the experiment data of Griffin [66]. Fig. 23 plots the comparison of maximum RMS velocity-magnitude fluctuations as a function of stream-wise locations x. The results from Griffin's experiment and the present method are presented. Fig. 24 shows the RMS velocity-magnitude fluctuations on the wake axis ($y=0$) as a function of the downstream locations as well as Griffin's results. Good agreements have been achieved from the above comparisons.

In Fig. 25, the time-average Nusselt numbers for the Dirichlet boundary condition are plotted as well as the results of Zhang *et al.* [40]. Fig. 26 is a comparison of surface temperature for the Neumann boundary condition. It can be seen clearly that the present results agree well with the data in Ref. [40]. As same as the case of stationary cylinder, the

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press

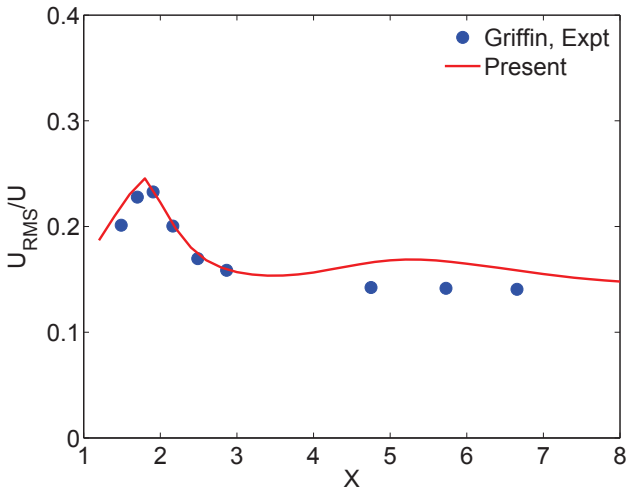


Figure 24: Comparison of RMS velocity-magnitude fluctuations at wake axis ($y=0$) as a function of stream-wise locations x with results from Griffin's experiment and the present method.

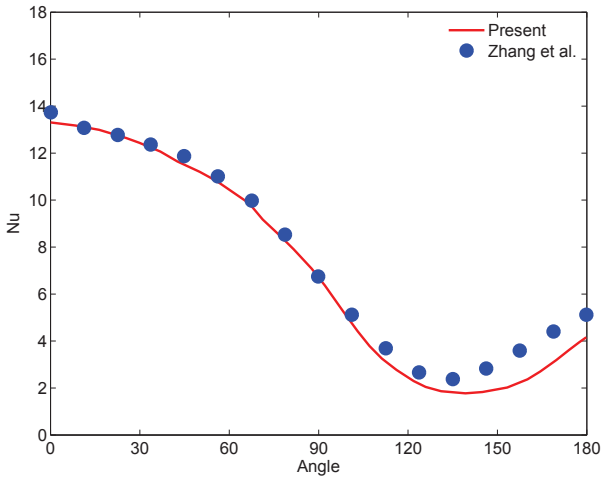


Figure 25: Comparison of local Nusselt on the oscillating cylinder surface with $Re=200$ for the constant temperature boundary condition.

Nusselt number or the temperature distribution exists an inflection point near the separation point of velocity boundary layer. In addition, for the Robin boundary condition, the influence of the Bi on time-average temperature distribution on the cylinder surface is shown in Fig. 27. We can find that the temperature distribution shares a similar pattern with the stationary cases.

As shown in Fig. 28, the time histories of average Nusselt number $\bar{Nu}$ for different boundary conditions are plotted. Obviously, it can be observed that the average Nusselt number $\bar{Nu}$ in all cases displays cyclic variations over time. Furthermore, we can

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press

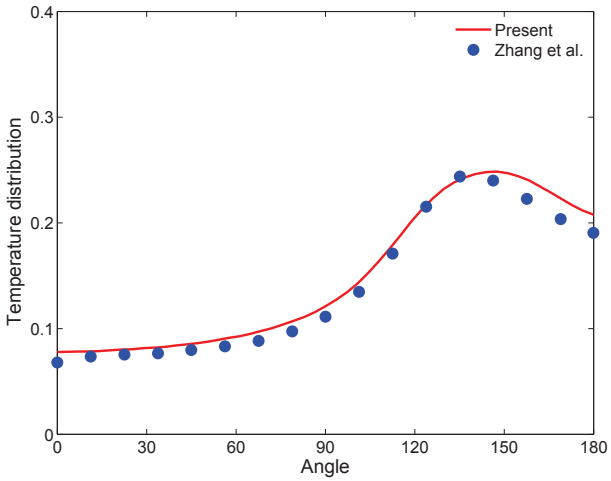


Figure 26: Comparison of temperature distribution on the oscillating cylinder surface with $Re = 200$ for the uniform heat flux boundary condition.

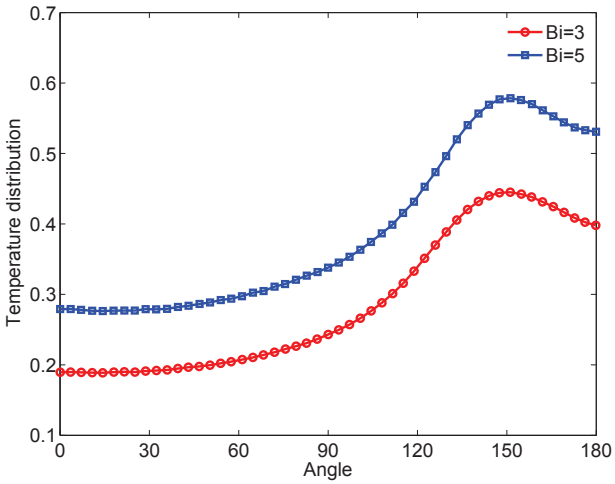


Figure 27: Comparison of temperature distribution on the oscillating cylinder surface with $Re = 200$ for the Robin boundary condition.

see clearly that $\bar{Nu}$(Isoflux)$>\bar{Nu}$(Robin, Bi=3)$>\bar{Nu}$(Robin, Bi=5)$>\bar{Nu}$(Isothermal). It indicates that the heat transfer of the cylinder surface with the iso-heat-flux boundary is most intense. Rather, it is weakest in the case with the isothermal boundary condition.

3.4 Natural convection in a concentric horizontal cylindrical annulus

Natural convection between horizontal concentric cylinders also has been studied by many researchers. Different from the heat flows around a cylinder, for this problem,

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press

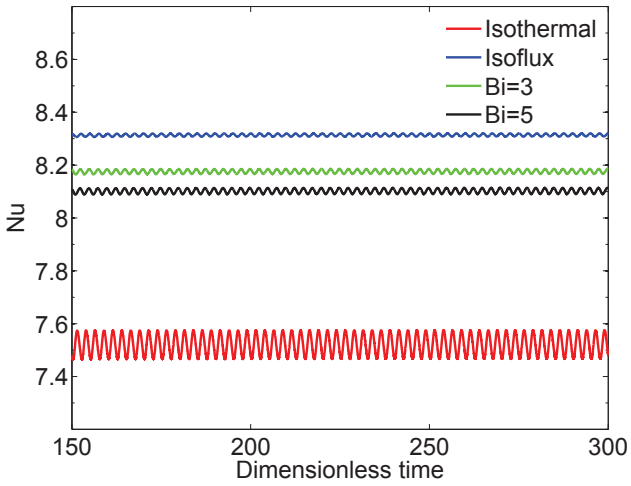


Figure 28: Comparison of average Nusselt number for different boundary conditions.

the temperature and velocity fields are strongly coupled. The fluid is filled between two concentric cylinder with different radiuses r_i,r_o. The Boussinesq approximation is used to estimate the effect of the buoyancy. Three different temperature boundary conditions on the inner cylinder are considered. The outer cylinder is kept a constant low temperature T_c. For this problem, the Rayleigh number Ra and Prandtl number Pr are the key parameters. The Rayleigh number is defined as

$$Ra=\frac{\rho c_p\beta R_g^3(T_h-T_c)}{\kappa\mu} \quad \text{or} \quad \frac{c_p\rho^2 g\beta R_g^4 Q_B}{\kappa^2\mu}, \tag{3.35}$$

where $R_g=R_o-R_i$ is the gap width of the annulus. The Prandtl number is fixed as 0.7. In all cases, the initial velocity and temperature are set as $\mathbf{u}=\mathbf{0}, T=T_c$. And the angle is defined in the clockwise sense starting from the top point of the inner cylinder.

Firstly, the case with Dirichlet-type boundary condition is studied. Two cylinders are embedded in a computational domain with a size of $4R_g\times 4R_g$. The radiuses of the outer and inner cylinder are set as $R_o=0.625R_g$ and $R_i=1.625R_g$, respectively. The Rayleigh number is set to be $Ra=5\times 10^4$. Fig. 29 shows the isotherms and the streamlines. To get more details, the comparison of the local temperature at the $\theta=2\pi/3$ with the results of Gan *et al*. [67] and Wang *et al*. [42]. can be observed in Fig. 30. We can draw a conclusion that the good agreement is obtained.

Then, the example with constant heat flux is considered. The solutions obtained by using the vorticity-streamfunction method and IBM based N-S solver as the benchmark data [45, 68]. As same as the previous studies, $Ra=5.7\times 10^3$ and 5×10^4 are selected. The size of the computational domain is set as $5R_g\times 5R_g$, where $R_g=R_i=0.5R_o$. As described in Fig. 31, the crescent-shaped eddies flow are observed. When the Rayleigh number is small ($Ra=5.7\times 10^3$), the thermal boundary layer on the bottom surface is

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press

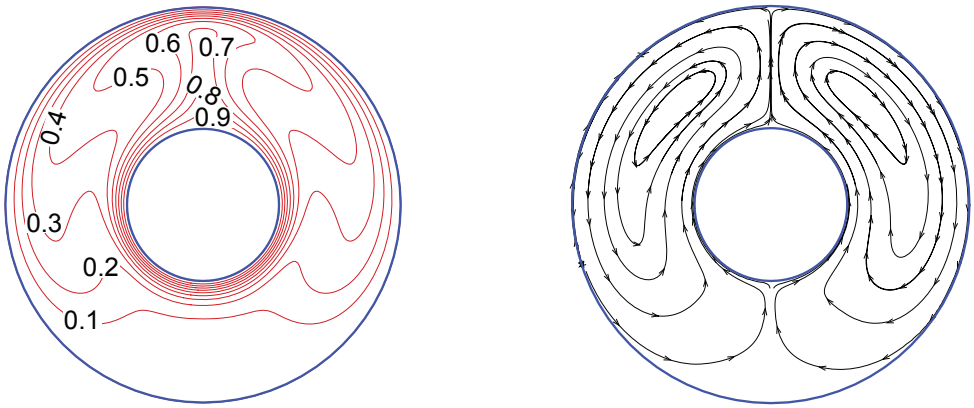


Figure 29: Isotherms (a) and streamlines (b) at $Ra=5\times10^4$.

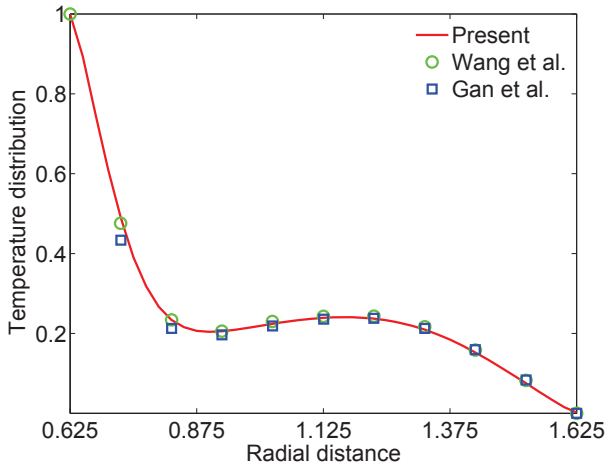
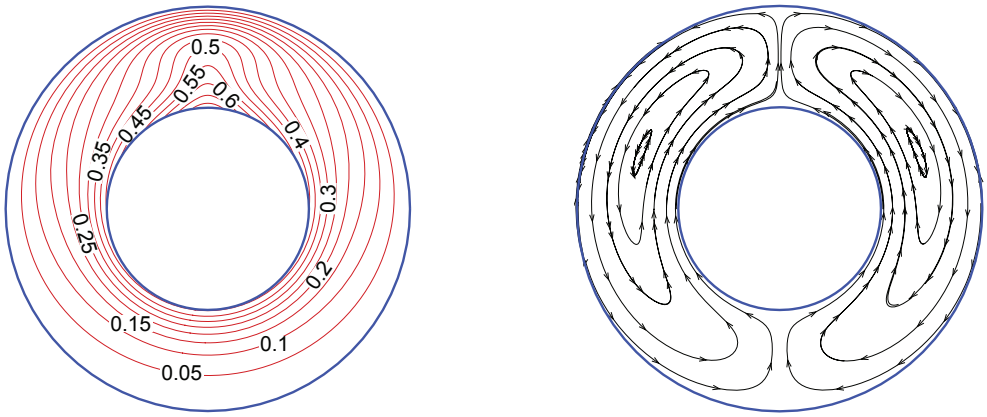


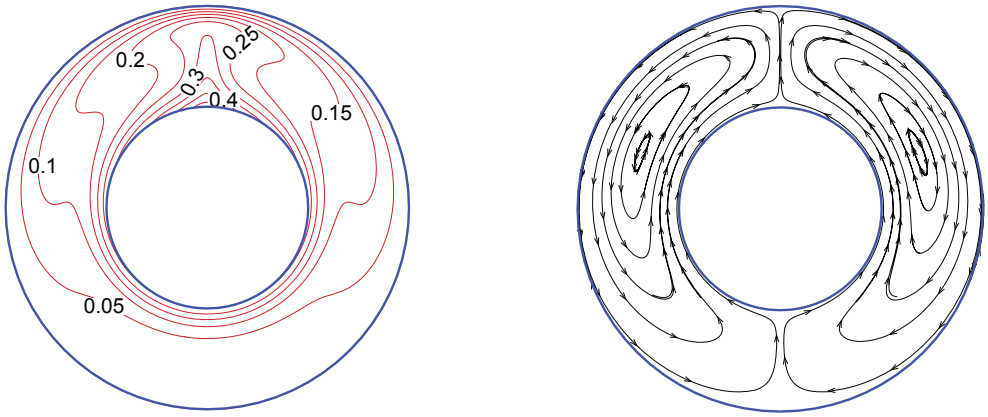
Figure 30: Comparison of the present results at $\theta=2\pi/3$.

thinner than that on the top surface. Compared with the temperature gradient on the top of the cylinder, the one on the bottom surface is larger. As the Rayleigh number increases ($Ra=5\times10^4$), the buoyancy displays significant effects on the flow and temperature fields. We can clearly see the S-type isotherms appear. This indicates that the heat flow is convective-dominated in this case. Fig. 32 shows the comparison of the temperature distributions on the surface of inner cylinder. For the case at $Ra=5700$, it can be observed that the present results agree well with the reference data of Yoo and Ren *et al*. [45, 68]. However, as the Rayleigh number increases to $Ra=50000$, compared with the data of Ren *et al*. [45], the present temperature curve is more closer to the result of Yoo [68] which was obtained by using the body-fitted method. We can also find that both the maximum temperature on the inner cylinder appear at the uppermost point of

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press

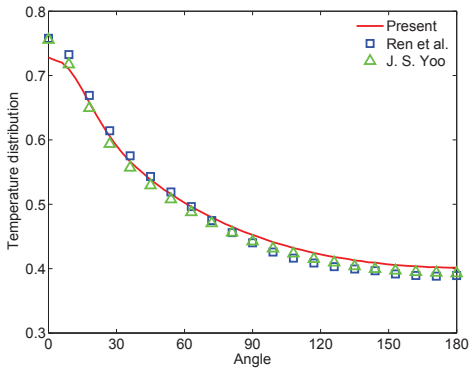


(a) $Ra=5700$

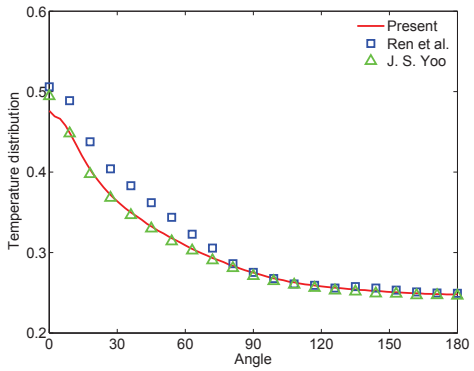


(b) $Ra=50000$

Figure 31: Isotherms (left) and streamlines (right) with the Neumann boundary condition on the inner cylinder surface.



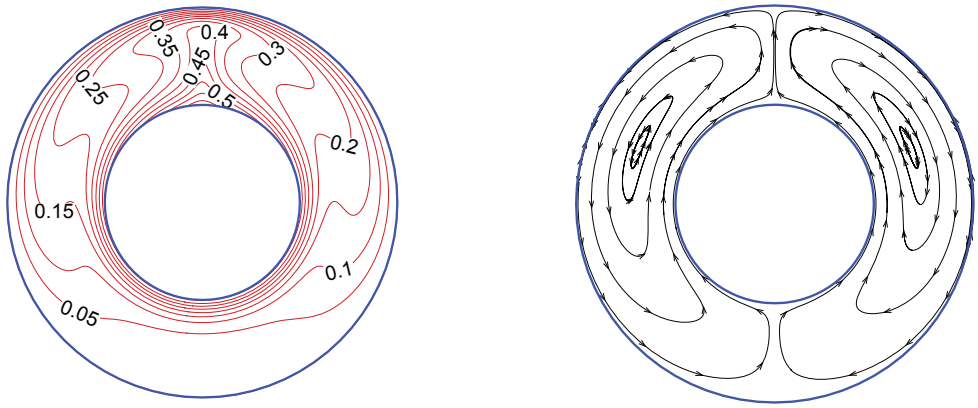
(a) $Ra=5700$



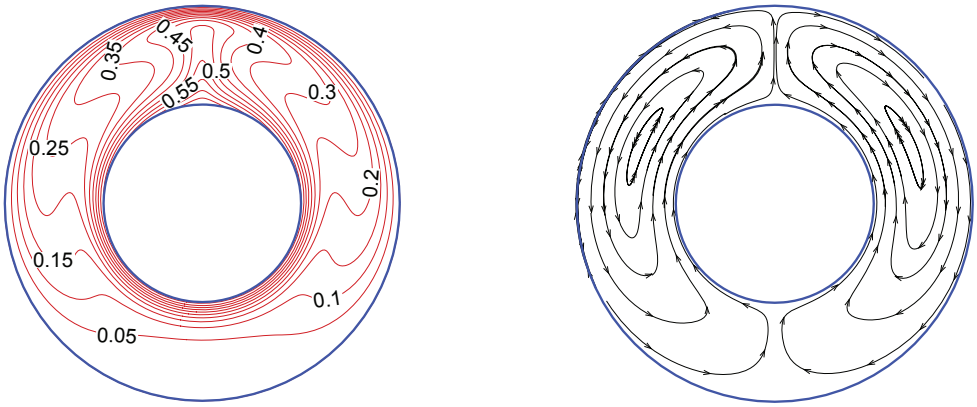
(b) $Ra=50000$

Figure 32: Comparison of the temperature distribution on the inner cylinder surface: (a) $Ra=5700$ and (b) $Ra=50000$.

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press



(a) $Bi=3$



(b) $Bi=5$

Figure 33: Isotherms (left) and streamlines (right) with the Robin boundary condition on the inner cylinder surface.

the annulus. Moreover, the temperatures on the surface with high Rayleigh number are larger than that with low Rayleigh number. Based on the Eq. (3.29), it means that the heat transfer is stronger as the Rayleigh number increases.

Lastly, the case of the inner cylinder with the Robin boundary is investigated. The Rayleigh number is also fixed at $Ra=5\times10^4$. The geometry configure is same as the case with Neumann boundary condition. Two kinds of $Bi=3,5$ are selected. Fig. 33 shows the streamlines and isotherms at the two different Bi. The local temperature distributions on the inner surface can be seen in Fig. 34. As same as the above numerical examples, the temperature on the inner boundary at high Bi is higher than the that at low Bi.

3.5 Sedimentation of a single isothermal cold particle in a vertical channel

To verify the capacity of the present method to simulate moving boundaries problem, the sedimentation of particle in vertical channel with heat convection is studied in the present

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press

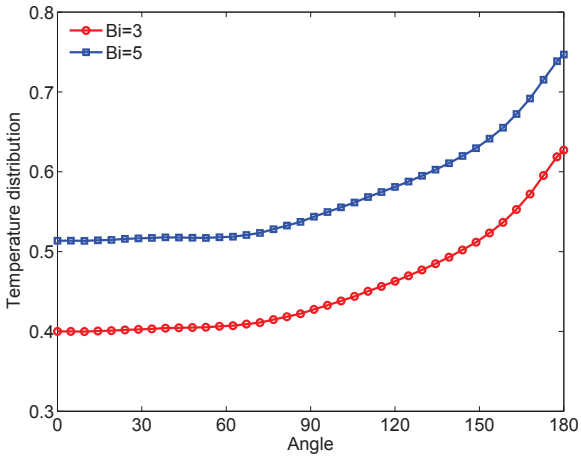


Figure 34: Comparison of the temperature distribution on the inner cylinder surface with different Bi.

work. This problem involves two types of thermal convection: natural convection and forced convection. This example was firstly investigated using an arbitrary Eulerian-Lagrangian method by Gan *et al*. [67]. Later Yu *et al*. [3] studied this example using fictitious domain method. In contrast to the work of Gan *et al*., the particle is located at a lateral position being one particle radius away from the centerline in Yu *et al*.'s research. Then many other researchers considered this example, such as Feng *et al*. (Direct numerical simulation) [41], Wachs (Parallel DNS method) [69], Kang *et al*. (Direct-forcing IB-LBM) [50], Haeri *et al*. (Implicit fictitious domain method) [4].

The bifurcation phenomena can be observed in this problem which depends on the Grashof number Gr. Gr is defined as

$$Gr=\frac{\rho^2 g\beta D^3(T_h-T_c)}{\mu^2}, \tag{3.36}$$

where D is the diameter of the circular particle and g is the gravitational acceleration.

Gan *et al*. have confirmed five different motion patterns of the particle. When $Gr\in(0,500)$, the particle is located at the centerlines with steady and symmetric wake. For $Gr\in(500,810)$, the particle achieves a lateral oscillation with small amplitude and vortex shedding occurs. When $Gr\in(800,2150)$, the final vertical position of the particle is located close to one of the walls and the vortex shedding disappear. For $Gr\in(2150,4500)$, a steady state of the particle which settles along the centerline can be observed again. When Gr is over 4500, the flow characteristic including large amplitude oscillation and turbulent-like vortex shedding can be observed which is caused by the Kelvin-Helmholtz instability.

In this study, following the previous work, the Prandtl number is set as $Pr=0.7$. The density ratio is chosen as $\rho_s/\rho_f=1.00232$, where ρ_s,ρ_f denote the densities of solid particle and fluid, respectively. We notice the density ratio is very close to 1. In order to ensure numerical stability, the particle motion algorithm in [21] is used. The characteristic

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press

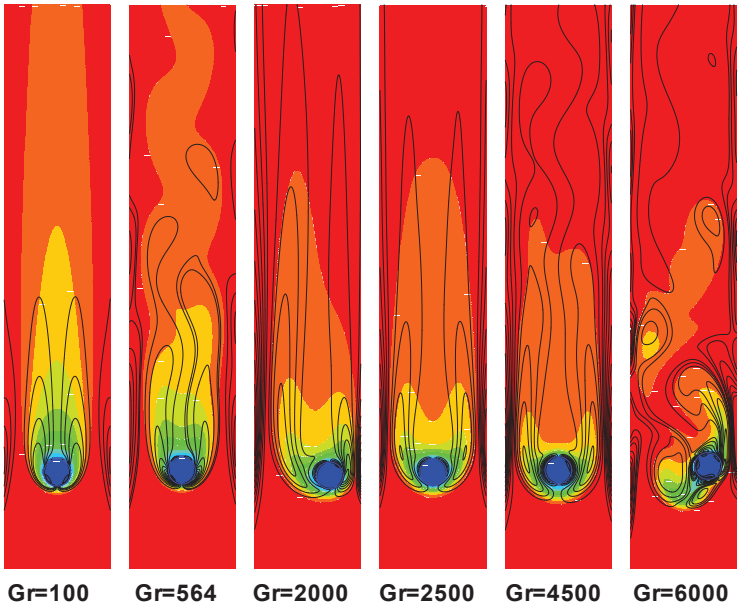


Figure 35: Vorticity contours and isotherms for different Grashof numbers.

velocity is defined by

$$U=\sqrt{\frac{\pi D}{2}(\frac{\rho_s}{\rho_f}-1)g}. \tag{3.37}$$

The Reynolds number Re is set as 40.5 which is consistent with the previous works. We choose a rectangle with the size of $4D\times 128D$ as the computational domain. And in all cases a 200×6400 grid is used.

As shown in Fig. 35, the vorticity contours and the isotherms are plotted. Fig. 36 illustrates the time history of the lateral position of the center of the particle in horizontal direction. The flow patterns at different Grashof numbers are basically consistent with the previous results except at the high Grashof numbers. It can be observed that the flow field situation at $Gr=4500$ are same as the results of Gan *et al*. [67], Wachs [69] but in contrast to the results of Yu *et al*. [3], Feng *et al*. [41] and Kang *et al*. [50]. In fact, the large amplitude transverse oscillation which generates at the early stage decays rapidly at $Gr=4500$. However, the large transverse oscillation is found in the case with higher Grashof number ($Gr=6000$). As explained in [69], the numerical schemes, grid and time steps have great influence on simulation solutions.

In Fig. 37, we compare the values of Re_T for different values of Gr. Re_T which named terminal velocity based Reynolds number can be calculated by

$$Re_T=Re\frac{U_T}{U}, \tag{3.38}$$

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press

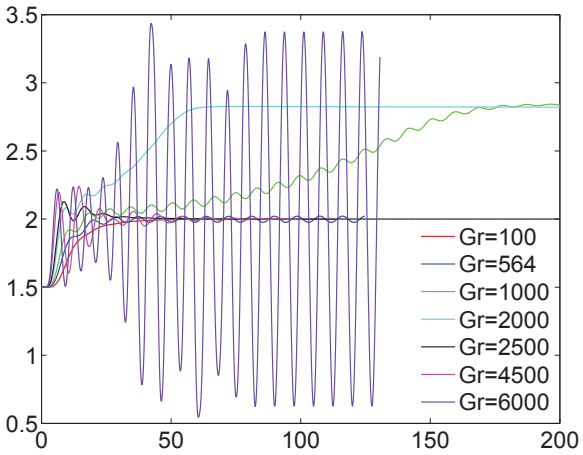


Figure 36: Time history of the lateral position of the center of the particle in horizontal direction.

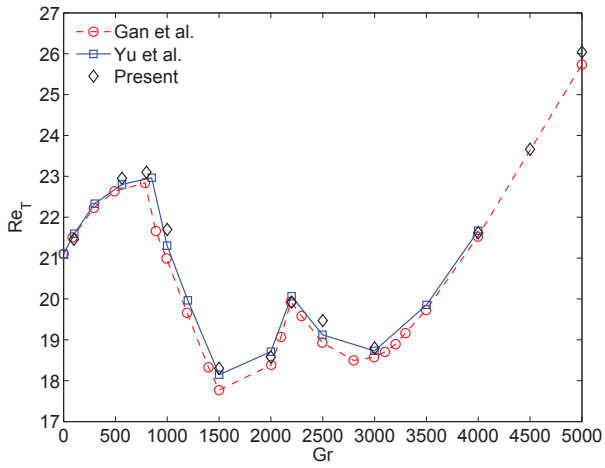


Figure 37: Comparison of the terminal velocity based Reynolds numbers for different Grashof numbers.

where U_T is the terminal velocity of the particle. Although there exists difference of the flow pattern at the high Grashof number, it is worth mentioning that the present values of Re_T are good agreement with the results of Gan *et al*. [67] and Yu *et al*. [3].

4 Conclusion

An efficient IBM which deals with three different temperature boundary conditions is developed in this paper. The LBM is used as flow and temperature fields solver. For different types of boundary conditions, the different strategies are applied. An iterative method is proposed to treat the Dirichlet boundary condition. The advantage of the itera-

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press

tive method is to enforce the given temperature on the boundary accurately. For the Neumann and Robin boundary conditions, based on the jump conditions on the boundary, two simple algebraic equations are constructed to solve the heat flux at the Lagrangian points. Then the temperature field is corrected by the heat flux distributing to the Eulerian points near the boundary. Some thermal flows are simulated to test the performance of the present IBM. A briefly discussion is given to explain that the Taylor Green vortices problem which is usually chosen as a test example of the accuracy verification can not get the correct conclusion. And the spatial accuracy of the present method is validated by the pure heat diffusion problem. The algorithm is also tested by some natural and forced convective problems. Good agreements are achieved between the results in this work and those in the previous literatures.

Acknowledgments

This work is supported by the Fundamental Research Funds for the Central Universities (Grant No. 2016YJS134). Y. Hu likes to thank Dr. Leping Qian for his help in image processing. Y. Hu also appreciates Dr. Haizhuan Yuan for helpful discussion.

Appendix A

The jump condition (2.54) is derived as follows. Here the properties of the Dirac function and Green's theorem are used (original reference, see [5]). As shown in Fig. 38, firstly we introduce a narrow region Ω_ε which contains the immersed boundary Γ. Then the Eq. (2.10) multiply by an arbitrary twice-differentiable function φ and integrate, we have

$$\begin{aligned}&\int_{\Omega_\varepsilon}\rho c_p\left(\frac{\partial T}{\partial t}+\mathbf{u}\cdot\nabla T\right)\varphi d\mathbf{x}=\int_{\Omega_\varepsilon}(\kappa\nabla^2 T+q)\varphi d\mathbf{x}\\ =&\int_{\Omega_\varepsilon}\kappa\nabla^2 T\varphi d\mathbf{x}+\int_{\Omega_\varepsilon}\int_\Gamma\delta Q(\mathbf{X})\delta(\mathbf{x}-\mathbf{X})ds\varphi d\mathbf{x}.\end{aligned}\tag{A.1}$$

By using the Green's theorem, the following formula is obtained

$$\begin{aligned}&\int_{\Omega_\varepsilon}\rho c_p\frac{\partial T}{\partial t}\varphi d\mathbf{x}+\int_{\partial\Omega_\varepsilon}\rho c_p\mathbf{u}\cdot\mathbf{n}T\varphi ds-\int_{\Omega_\varepsilon}\rho c_p\nabla\varphi\cdot\mathbf{u}Td\mathbf{x}\\ =&\int_{\partial\Omega_\varepsilon}\kappa\nabla T\cdot\mathbf{n}\varphi ds-\int_{\Omega_\varepsilon}\kappa\nabla T\cdot\nabla\varphi d\mathbf{x}+\int_\Gamma\delta Q(\mathbf{X})\varphi ds\\ =&\int_{\partial\Omega_\varepsilon}\kappa\nabla T\cdot\mathbf{n}\varphi ds-\int_{\partial\Omega_\varepsilon}\kappa T\nabla\varphi\cdot\mathbf{n}ds\\ &+\int_{\Omega_\varepsilon}\kappa T\nabla^2\varphi d\mathbf{x}+\int_\Gamma\delta Q(\mathbf{X})\varphi ds.\end{aligned}\tag{A.2}$$

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press

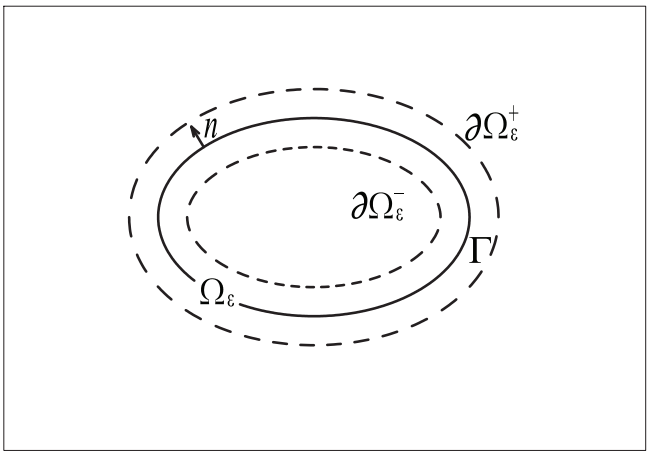


Figure 38: A schematic view used to prove the interface relations.

Let $\varepsilon\to 0$, we have $\partial\Omega_\varepsilon^+\to\Gamma$, $\partial\Omega_\varepsilon^-\to\Gamma$. The following formulas hold

$$\int_{\Omega_\varepsilon}\rho c_p\frac{\partial T}{\partial t}\varphi d\mathbf{x}\to 0, \tag{A.3}$$

$$\int_{\Omega_\varepsilon}\kappa T\nabla^2\varphi d\mathbf{x}\to 0, \tag{A.4}$$

$$\int_{\Omega_\varepsilon}\rho c_p\nabla\varphi\cdot\mathbf{u}T d\mathbf{x}\to 0. \tag{A.5}$$

On the immersed boundary, $[T]_\Gamma=0$, $[\mathbf{u}]_\Gamma=\mathbf{0}$, we get

$$\begin{aligned}\int_{\partial\Omega_\varepsilon}\rho c_p\mathbf{u}\cdot\mathbf{n}T\varphi ds&=\int_{\partial\Omega_\varepsilon^+}\rho c_p\mathbf{u}\cdot\mathbf{n}T\varphi ds+\int_{\partial\Omega_\varepsilon^-}\rho c_p\mathbf{u}\cdot(-\mathbf{n})T\varphi ds\\&=\int_{\partial\Omega_\varepsilon^+}\rho c_p\mathbf{u}\cdot\mathbf{n}T\varphi ds-\int_{\partial\Omega_\varepsilon^-}\rho c_p\mathbf{u}\cdot\mathbf{n}T\varphi ds\\&\xrightarrow{\varepsilon\to 0}\int_\Gamma\rho c_p[\mathbf{u}]\cdot\mathbf{n}T\varphi ds=0,\end{aligned} \tag{A.6}$$

$$\begin{aligned}\int_{\partial\Omega_\varepsilon}\kappa T\nabla\varphi\cdot\mathbf{n}ds&=\int_{\partial\Omega_\varepsilon^+}\kappa T\nabla\varphi\cdot\mathbf{n}+\int_{\partial\Omega_\varepsilon^-}\kappa T\nabla\varphi\cdot(-\mathbf{n})\\&=\int_{\partial\Omega_\varepsilon^+}\kappa T\nabla\varphi\cdot\mathbf{n}-\int_{\partial\Omega_\varepsilon^-}\kappa T\nabla\varphi\cdot\mathbf{n}\\&\xrightarrow{\varepsilon\to 0}\int_\Gamma\kappa[T]\nabla\varphi\cdot\mathbf{n}=0,\end{aligned} \tag{A.7}$$

$$\int_\Gamma\left(\kappa\left[\frac{\partial T}{\partial n}\right]+\delta Q\right)\cdot\varphi ds=0. \tag{A.8}$$

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press

Based the above discussion, we get

$$-\kappa\left[\frac{\partial T}{\partial n}\right]=\delta Q. \tag{A.9}$$

For Eq. (2.55), the proof is straightforward. Differentiating the following equation along the interface

$$[T(\mathbf{X}(s))]=0, \tag{A.10}$$

where s is the arc-length parameter of the interface Γ, wet get

$$\left[\frac{dT}{ds}\right]=\left[\frac{\partial T}{\partial \tau}\right]=0. \tag{A.11}$$

Appendix B

For Lemma 2.2, the Taylor series expansion method is used. As shown in Fig. 3, $T(x_{i,j},y_{i,j})$ is expanded at $T(X_l,Y_l)$ as follow

$$\begin{aligned} T(x_{i,j},y_{i,j})=&T(X_l,Y_l)+T_x^+(X_l,Y_l)(x_{i,j}-X_l)\\ &+T_y^+(X_l,Y_l)(y_{i,j}-Y_l)+\mathcal{O}(h^2), \quad (x_{i,j},y_{i,j})\in\Omega^+, \end{aligned} \tag{B.1}$$

$$\begin{aligned} T(x_{i,j},y_{i,j})=&T(X_l,Y_l)+T_x^-(X_l,Y_l)(x_{i,j}-X_l)\\ &+T_y^-(X_l,Y_l)(y_{i,j}-Y_l)+\mathcal{O}(h^2), \quad (x_{i,j},y_{i,j})\in\Omega^-. \end{aligned} \tag{B.2}$$

Firstly, we have

$$\begin{aligned} &\frac{T_{i+1,j}-T_{i-1,j}}{2h}=\\ &=a_{i,j}^l T_x^+(\mathbf{X}_l)+b_{i,j}^l T_x^-(\mathbf{X}_l)+c_{i,j}^l T_y^+(\mathbf{X}_l)+d_{i,j}^l T_y^-(\mathbf{X}_l)+\mathcal{O}(h). \end{aligned} \tag{B.3}$$

If $\mathbf{x}_{i+1,j},\mathbf{x}_{i-1,j}\in\Omega^+$,

$$a_{i,j}^l=\frac{x_{i+1,j}-x_{i-1,j}}{2h}, \quad b_{i,j}^l=0, \quad c_{i,j}^l=0, \quad d_{i,j}^l=0. \tag{B.4}$$

If $\mathbf{x}_{i+1,j},\mathbf{x}_{i-1,j}\in\Omega^-$,

$$a_{i,j}^l=0, \quad b_{i,j}^l=\frac{x_{i+1,j}-x_{i-1,j}}{2h}, \quad c_{i,j}^l=0, \quad d_{i,j}^l=0. \tag{B.5}$$

If $\mathbf{x}_{i+1,j}\in\Omega^+,\mathbf{x}_{i-1,j}\in\Omega^-$,

$$a_{i,j}^l=\frac{x_{i+1,j}-X_l}{2h}, \quad b_{i,j}^l=\frac{x_{i-1,j}-X_l}{2h}, \quad c_{i,j}^l=\frac{y_{i,j}-Y_l}{2h}, \quad d_{i,j}^l=\frac{y_{i,j}-Y_l}{2h}. \tag{B.6}$$

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press

If $\mathbf{x}_{i+1,j}\in\Omega^-, \mathbf{x}_{i-1,j}\in\Omega^+$,

$$a^l_{i,j}=\frac{x_{i-1,j}-X_l}{2h},\quad b^l_{i,j}=\frac{x_{i+1,j}-X_l}{2h},\quad c^l_{i,j}=\frac{y_{i,j}-Y_l}{2h},\quad d^l_{i,j}=\frac{y_{i,j}-Y_l}{2h}. \tag{B.7}$$

Then

$$\begin{aligned}&\sum_{(i,j)\in S_l} A^l_{i,j}\frac{T_{i+1,j}-T_{i-1,j}}{2h}\\ &=\sum_{(i,j)\in S_l} A^l_{i,j}a^l_{i,j}T^+_x(\mathbf{X}_l)+\sum_{(i,j)\in S_l} A^l_{i,j}b^l_{i,j}T^-_x(\mathbf{X}_l)\\ &\quad+\sum_{(i,j)\in S_l} A^l_{i,j}c^l_{i,j}T^+_y(\mathbf{X}_l)+\sum_{(i,j)\in S_l} A^l_{i,j}d^l_{i,j}T^-_y(\mathbf{X}_l).\end{aligned} \tag{B.8}$$

Obviously, for a second-order interpolation method, we have

$$\sum_{(i,j)\in S_l} A^l_{i,j}a^l_{i,j}+\sum_{(i,j)\in S_l} A^l_{i,j}b^l_{i,j}=1, \tag{B.9}$$

$$\sum_{(i,j)\in S_l} A^l_{i,j}c^l_{i,j}+\sum_{(i,j)\in S_l} A^l_{i,j}d^l_{i,j}=0. \tag{B.10}$$

If we note

$$\alpha_1=\sum_{(i,j)\in S_l} A^l_{i,j}a^l_{i,j}, \tag{B.11}$$

$$\beta_1=\sum_{(i,j)\in S_l} A^l_{i,j}c^l_{i,j}, \tag{B.12}$$

the Eq. (2.57) follows. Similarly, we can prove Eq. (2.58).

References

[1] R. Glowinski, T. W. Pan, T. Hesla, D. Joseph, A distributed Lagrange multiplier/fictitious domain method for particulate flows, Int. J. Multiphase Flow 25 (1999) 755-794.
[2] R. Glowinski, T. Pan, T. Hesla, D. Joseph, J. Périaux, A fictitious domain approach to the direct numerical simulation of incompressible viscous flow past moving rigid bodies: Application to particulate flow, J. Comput. Phys. 169 (2001) 363-426.
[3] Z. Yu, X. Shao, A. Wachs, A fictitious domain method for particulate flows with heat transfer, J. Comput. Phys. 217 (2006) 424-452.
[4] S. Haeri, J. Shrimpton, A new implicit fictitious domain method for the simulation of flow in complex geometries with heat transfer, J. Comput. Phys. 237 (2013) 21-45.
[5] R. LeVeque, Z. L. Li, Immersed interface methods for Stokes flow with elastic boundaries or surface tension, SIAM J. Sci. Comput. 18 (1997) 709-735.
[6] Z. Li, M. C. Lai, The immersed interface method for the Navier-Stokes equations with singular forces, J. Comput. Phys. 171 (2001) 822-842.

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press

[7] D. Le, B. Khoo, J. Peraire, An immersed interface method for viscous incompressible flows involving rigid and flexible boundaries, J. Comput. Phys. 220 (2006) 109-138.
[8] Z. Li, S. R. Lubkin, Numerical analysis of interfacial two-dimensional Stokes flow with discontinuous viscosity and variable surface tension, Int. J. Numer. Meth. Fl. 37 (2001) 525-540.
[9] C. S. Peskin, Numerical analysis of blood flow in the heart, J. Comput. Phys. 25 (1977) 220-252.
[10] X. He, L. S. Luo, Theory of the lattice Boltzmann method: From the Boltzmann equation to the lattice Boltzmann equation, Phys. Rev. E 56 (1997) 6811-6817.
[11] S. Chen, G. D. Doolen, Lattice Boltzmann method for fluid flows, Annu. Rev. Fluid Mech. 30 (1998) 329-364.
[12] X. W. Shan, H. D. Chen, Lattice Boltzmann model for simulating flows with multiple phases and components, Phys. Rev. E 47 (1993) 1815-1819.
[13] X. Y. He, S. Y. Chen, R. Y. Zhang, A lattice boltzmann scheme for incompressible multiphase flow and its application in simulation of Rayleigh-Taylor instability, J. Comput. Phys. 152 (1999) 642-663.
[14] C. M. Teixeira, Incorporating turbulence models into the lattice-Boltzmann method, Int. J. Mod. Phys. C 09 (1998) 1159-1175.
[15] H. D. Yu, S. S. Girimaji, L. S. Luo, DNS and LES of decaying isotropic turbulence with and without frame rotation using lattice boltzmann method, J. Comput. Phys. 209 (2005) 599-616.
[16] X. Nie, G. D. Doolen, S. Chen, Lattice-Boltzmann simulations of fluid flows in MEMS, Phys. Fluids 107 (2002) 279-289.
[17] C. Y. Lim, C. Shu, X. D. Niu, Y. T. Chew, Application of lattice Boltzmann method to simulate microchannel flows, Phys. Fluids 14 (2002) 2299-2308.
[18] H. Z. Yuan, X. D. Niu, S. Shu, M. J. Li, H. Yamaguchi, A momentum exchange-based immersed boundary-lattice Boltzmann method for simulating a flexible filament in an incompressible flow, Comput. Math. Appl. 67 (2014) 1039-1056.
[19] H. Z. Yuan, X. D. Niu, S. Shu, M. J. Li, Y. Hu, A numerical study of jet propulsion of an oblate jellyfish using a momentum exchange-based immersed boundary-lattice Boltzmann method, Adv. Appl. Math. Mech. 6 (2014) 307-326.
[20] H. Zhang, Y. Q. Tan, S. Shu, X. D. Niu, F. X. Trias, D. M. Yang, H. Li, Y. Sheng, Numerical investigation on the role of discrete element method in combined LBM-IBM-DEM modeling, Comput. Fluids 94 (2014) 37-48.
[21] Y. Hu, D. C. Li, S. Shu, X. D. Niu, Modified momentum exchange method for fluid-particle interactions in the lattice Boltzmann method, Phys. Rev. E 91 (2015) 033301.
[22] X. Wang, C. Shu, J. Wu, L. Yang, An efficient boundary condition-implemented immersed boundary-lattice Boltzmann method for simulation of 3D incompressible viscous flows, Comput. Fluids 100 (2014) 165-175.
[23] X. He, S. Chen, G. D. Doolen, A novel thermal model for the lattice Boltzmann method in incompressible limit, J. Comput. Phys. 146 (1998) 282-300.
[24] Y. Hu, X. D. Niu, S. Shu, H. Z. Yuan, M. J. Li, Natural convection in a concentric annulus: A lattice Boltzmann method study with boundary condition-enforced immersed boundary method, Adv. Appl. Math. Mech. 5 (2013) 321-336.
[25] Y. Hu, D. Li, S. Shu, X. Niu, An efficient smoothed profile-lattice Boltzmann method for the simulation of forced and natural convection flows in complex geometries, Int. Commun. Heat Mass Transfer 68 (2015) 188-199.
[26] Y. Hu, D. Li, S. Shu, X. Niu, Simulation of steady fluid-solid conjugate heat transfer problems via immersed boundary-lattice Boltzmann method, Comput. Math. Appl. 70 (2015) 2227-

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press

2237.
[27] Y. Hu, D. Li, S. Shu, X. Niu, Full Eulerian lattice Boltzmann model for conjugate heat transfer, Phys. Rev. E 92 (2015) 063305.
[28] Z. Guo, T. Zhao, Lattice Boltzmann model for incompressible flows through porous media, Phys. Rev. E 66 (2002) 036304.
[29] Y. Hu, D. Li, S. Shu, X. Niu, Immersed boundary-lattice Boltzmann simulation of natural convection in a square enclosure with a cylinder covered by porous layer, Int. Commun. Heat Mass Transfer 92 (2016) 1166-1170.
[30] Z. G. Feng, E. E. Michaelides, The immersed boundary-lattice Boltzmann method for solving fluid-particles interaction problems, J. Comput. Phys. 195 (2004) 602-628.
[31] Z. G. Feng, E. E. Michaelides, Proteus: a direct forcing method in the simulations of particulate flows, J. Comput. Phys. 202 (2005) 20-51.
[32] X. Niu, C. Shu, Y. Chew, Y. Peng, A momentum exchange-based immersed boundary-lattice Boltzmann method for simulating incompressible viscous flows, Phys. Lett. A 354 (2006) 173-182.
[33] J. Zhang, P. C. Johnson, A. S. Popel, An immersed boundary lattice Boltzmann approach to simulate deformable liquid capsules and its application to microscopic blood flows, Phys. Biol. 4 (2007) 285.
[34] F.-B. Tian, H. Luo, L. Zhu, J. C. Liao, X.-Y. Lu, An efficient immersed boundary-lattice Boltzmann method for the hydrodynamic interaction of elastic filaments, J. Comput. Phys. 230 (2011) 7266-7283.
[35] J. Hao, L. Zhu, A lattice Boltzmann based implicit immersed boundary method for fluid-structure interaction, Comput. Math. Appl. 59 (2010) 185-193.
[36] A. Dupuis, P. Chatelain, P. Koumoutsakos, An immersed boundary-lattice-Boltzmann method for the simulation of the flow past an impul- sively started cylinder, J. Comput. Phys. 227 (2008) 4486-4498.
[37] J. Kim, H. Choi, An immersed-boundary finite-volume method for simulation of heat transfer in complex geometries, KSME Int. J. 18 (2004) 1026-1035.
[38] J. R. Pacheco, V. A. Pacheco, T. Rodi c, R. E. Peck, Numerical simulations of heat transfer and fluid flow problems using an immersed- boundary finite-volume method on nonstaggered grids, Numer. Heat Tr. B-Fund 48 (2005) 1-24.
[39] V. A. Pacheco, J. R. Pacheco, T. Rodi c, A general scheme for the boundary conditions in convective and diffusive heat transfer with immersed boundary methods, ASME J. Heat Transfer 129 (2007) 1506-1516.
[40] N. Zhang, Z. Zheng, S. Eckels, Study of heat-transfer on the surface of a circular cylinder in flow using an immersed-boundary method, Int. J. Heat Fluid Fl. 29 (2008) 1558-1566.
[41] Z.-G. Feng, E. E. Michaelides, Heat transfer in particulate flows with direct numerical simulation (DNS), Int. J. Heat Mass Transfer 52 (2009) 777-786.
[42] Z. Wang, J. Fan, K. Luo, K. Cen, Immersed boundary method for the simulation of flows with heat transfer, Int. J. Heat Mass Transfer 52 (2009) 4510-4518.
[43] H. Jeong, H. Yoon, M. Ha, M. Tsutahara, 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. 229 (2010) 2526-2543.
[44] W. Ren, C. Shu, J. Wu, W. Yang, Boundary condition-enforced immersed boundary method for thermal flow problems with Dirichlet temperature condition and its applications, Comput. Fluids 57 (2012) 40-51.
[45] W. Ren, C. Shu, W. Yang, An efficient immersed boundary method for thermal flow prob-

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press

lems with heat flux boundary conditions, Int. J. Heat Mass Transfer 64 (2013) 694-705.
[46] M. Uhlmann, An immersed boundary method with direct forcing for the simulation of particulate flows, J. Comput. Phys. 209 (2005) 448-476.
[47] D. J. Chen, K. H. Lin, C. A. Lin, Immersed boundary method based lattice Boltzmann method to simulate 2D and 3D complex geometry flows, Int. J. Mod. Phys. C 18 (2007) 585-594.
[48] J. Wu, C. Shu, Implicit velocity correction-based immersed boundary-lattice Boltzmann method and its applications, J. Comput. Phys. 228 (2009) 1963-1979.
[49] S. Wang, X. Zhang, An immersed boundary method based on discrete stream function formulation for two- and three-dimensional incom- pressible flows, J. Comput. Phys. 230 (2011) 3479-3499.
[50] S. K. Kang, Y. A. Hassan, A comparative study of direct-forcing immersed boundary-lattice Boltzmann methods for stationary complex boundaries, Int. J. Numer. Meth. Fl. 66 (2011) 1132-1158.
[51] Y. Hu, H. Z. Yuan, S. Shu, X. D. Niu, M. J. Li, An improved momentum exchaged-based immersed boundary-lattice Boltzmann method by using iterative technique, Comput. Math. Appl. 68 (2014) 140-155.
[52] S. W. Son, H. S. Yoon, H. K. Jeong, M. Y. Ha, S. Balachandar, Discrete lattice effect of various forcing methods of body force on immersed boundary-lattice boltzmann method, J. Mech. Sci. Technol. 27 (2013) 429-441.
[53] N. Thai-Quang, N. Mai-Duy, C. D.Tran, T. Tran-Cong, A direct forcing immersed boundary method employed with compact integrated RBF approximations for heat transfer and fluid flow problems, CMES-Comp. Model. Eng. 96 (2013) 49-90.
[54] S. Dash, T. Lee, T. Lim, H. Huang, A flexible forcing three dimension IB-LBM scheme for flow past stationary and moving spheres, Comput. Fluids 95 (2014) 159-170.
[55] C. S. Peskin, The immersed boundary method, Acta Numer. 11 (2002) 479-517.
[56] B. E. Griffith, C. S. Peskin, On the order of accuracy of the immersed boundary method: Higher order convergence rates for sufficiently smooth problems, J. Comput. Phys. 208 (2005) 75-105.
[57] R. P. Beyer, R. J. Leveque, Analysis of a one-dimensional model for the immersed boundary method, SIAM J. Numer. Anal. 29 (1992) 332-364.
[58] R. D. Guy, D. A. Hartenstine, On the accuracy of direct forcing immersed boundary methods with projection methods, J. Comput. Phys. 229 (2010) 2479-2496.
[59] C. Shu, W. Ren, W. Yang, Novel immersed boundary methods for thermal flow problems, Int. J. Numer. Method H. 23 (2013) 124-142.
[60] H. Ding, C. Shu, K. S. Yeo, Simulation of natural convection in eccentric annuli between a square outer cylinder and a circular inner cylinder using local MQ-DQ method, Numer. Heat Tr. A-App. 47 (2005) 291-313.
[61] R. P. Bharti, R. P. Chhabra, V. Eswaran, A numerical study of the steady forced convection heat transfer from an unconfined circular cylinder, Heat Mass Transfer 43 (2007) 639-648.
[62] E. M. Sparrow, J. P. Abraham, J. C. Tong, Archival correlations for average heat transfer coefficients for non-circular and circular cylinders and for spheres in cross-flow, Int. J. Heat Mass Transfer 47 (2004) 5285-5296.
[63] R. A. Ahmad, Z. H. Qureshi, Laminar mixed convection from a uniform heat flux horizontal cylinder in a crossflow, J. Thermophys. Heat Tr. 6 (1992) 277-287.
[64] D. Pan, A general boundary condition treatment in immersed boundary methods for incompressible Navier-Stokes equations with heat transfer, Numer. Heat Tr. B-Fund. 61 (2012)

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press

279-297.
[65] N. Zhang, Z. Zheng, An improved direct-forcing immersed-boundary method for finite difference applications, J. Comput. Phys. 221 (2007) 250-268.
[66] O. M. Griffin, The unsteady wake of an oscillating cylinder at low Reynolds number, J. Appl. Mech. 38 (1971) 729-738.
[67] H. Gan, J. Z. Chang, J. J. Feng, H. H. Hu, Direct numerical simulation of the sedimentation of solid particles with thermal convection, J. Fluid Mech. 481 (2003) 385-411.
[68] J. S. Yoo, Dual free-convective flows in a horizontal annulus with a constant heat flux wall, Int. J. Heat Mass Transfer 46 (2003) 2499-2503.
[69] A. Wachs, Rising of 3D catalyst particles in a natural convection dominated flow by a parallel DNS method, Comput. Chem. Eng. 35 (2011) 2169-2185.

https://doi.org/10.4208/cicp.090815.170316a Published online by Cambridge University Press