Numerical Illustrations of the Coupling Between the Lattice Boltzmann Method and Finite-Type Macro-Numerical Methods

H. B. Luan [a]; H. Xu [a]; L. Chen [a]; D. L. Sun [b]; W. Q. Tao [a]

[a] State Key Laboratory of Multiphase Flow in Power Engineering, School of Energy & Power Engineering, Xi'an Jiaotong University, Xi'an, Shaanxi, People's Republic of China [b] Beijing Key Laboratory of New and Renewable Energy, North China Electric Power University, Beijing, People's Republic of China

Online publication date: 08 March 2010

PLEASE SCROLL DOWN FOR ARTICLE

Taylor & Francis
Taylor & Francis Group

NUMERICAL ILLUSTRATIONS OF THE COUPLING BETWEEN THE LATTICE BOLTZMANN METHOD AND FINITE-TYPE MACRO-NUMERICAL METHODS

H. B. Luan[1], H. Xu[1], L. Chen[1], D. L. Sun[2], and W. Q. Tao[1]

[1]*State Key Laboratory of Multiphase Flow in Power Engineering, School of Energy & Power Engineering, Xi'an Jiaotong University, Xi'an, Shaanxi, People's Republic of China*
[2]*Beijing Key Laboratory of New and Renewable Energy, North China Electric Power University, Beijing, People's Republic of China*

An analytic expression called a reconstruction operator is proposed for the exchange from velocity of finite-type methods to the single-particle distribution function of the lattice Boltzmann method (LBM). The combined finite-volume method and lattice Boltzmann method (called the CFVLBM) is adopted to solve three flow cases, backward-facing flow, flow around a circular cylinder, and lid-driven cavity flow. The results predicted by the CFVLBM agree with the available numerical solutions very well. It is shown that the vorticity contour distribution is a more appropriate parameter to ensure good smoothness and consistency at the coupling interface. At the same time, CPU time used by the CFVLBM(II), with more than one outer iteration before interface information exchange, is much less than that of the CFVLBM(I), where interface information exchanges are executed after each outer iteration.

1. INTRODUCTION

Challenging multiscale phenomena or processes exist widely in, for example, material science, fluid flows, and electrical and mechanical engineering [1–3]. Things are made up of atoms and electrons at the atomic scale, but they are usually several orders of magnitude larger when they are characterized by their own geometric dimensions. The study of multiscale problems has become one of the highlights of numerical simulation techniques, and several international journals have been created in the past 10 years. Examples of multiscale problems include turbulent fluid flow and heat transfer, transport phenomena in the proton exchange membranes of fuel cells, cooling processes in data centers, and so on. Multiscale problems can be divided into two categories: multiscale systems and multiscale processes. By multiscale system we refer to a system that is characterized by large variations in length

NOMENCLATURE

c	lattice speed	u_∞	far-field velocity
c_s	lattice sound speed	u_0	inlet velocity
C	compression operator	x, y	Cartesian coordinates
C_D	drag coefficient	Δx	space step
D2Q9	2-dimension, 9-velocity lattice	Δt	time step
f	lattice distribution function	ε	expanding parameter
H	height	θ	separation angle
I	identity operator	λ	relaxation time
Kn	Knudsen number	ν	kinematic viscosity
L	reattachment length	ρ	density
Ma	Mach number	τ	nondimensional relaxation time
p	pressure	ω	weight factor
r	cylinder radius		
R	reconstruction operator	**Subscripts**	
Re	Reynolds number	i	direction of the discretized velocity
S	stress tension, step height	α, β, γ	coordinate direction indices
t	time		
u, v	velocities along the x, y directions	**Superscript**	
u_{lid}	lid-driven velocity	eq	equilibrium

scales. The processes at different scales are not very closely related and can be studied separately with certain connections. Cooling in data centers is a typical multiscale system problem. The length scale of the cooling stream in the center as a whole is of the order of meters, while the cooling process of a chip is of the order of millimeters. By a multiscale process we mean that the overall behavior is governed by processes that occur at different length and/or time scales, and they are inherently connected by the process itself. Processes in PEMFC, launching a space rocket from the earth surface to outer space, and turbulent heat transfer are examples of multiscale process. From the viewpoint of simulation, study of the multiscale processes is more challenging and attractive. The focus of the present article is therefore on the multiscale processes.

It is usually accepted that different scale problems have different numerical methods which are most applicable to the corresponding scales. Broadly speaking, there are three levels of simulation methods for fluid flow and heat transfer: macro-scale, mesoscale, and microscale. The macroscale numerical methods include the finite-difference method (FDM), the finite-volume method (FVM), the finite-element method (FEM), and the finite analytic method (FAM) [4]. The basic feature of the four methods is that the smallest unit for computation is a cell with finite dimensions. Thus they can be called finite-type methods. The mesoscale numerical methods include the lattice Boltzmann method (LBM) and the direct-simulation Monte Carlo method (DSMC). These two methods adopt a concept of computational particles which are much larger than actual molecules but act as molecules (simulation mole-cules). The micro numerical methods include molecular dynamic simulation (MDS) and quantum molecular simulation (QMS). In MDS, every molecule is simulated according to Newton's law of motion. A quantum molecular dynamics simulation solves the coupled time-dependent Schrödinger equations for all particles in the system. This method is largely limited by present computer resources; hence, various approximations have to be used.

As indicated above, many engineering processes are multiscale in nature. However, mainly because of the limitation of computer resources; before the emergence of the new simulation field of "multiscale simulation," the inherently multiscale processes were usually simulated by single-scale numerical methods. One exception is direct numerical simulation (DNS) for turbulent flow. The numerical methods used in DNS, such as finite difference or finite volume, are of macroscale type. However, since a very tiny time scale and a very fine space scale have to be used in DNS, the time and space resolutions in DNS are fine enough to resolve eddies and fluctuations at different scales. Hence the numerical results obtained by DNS include enormous instant and local information and are essentially in multiscale. Averaged parameters, such as time-averaged velocity, can be obtained from the simulation results. Since DNS is not suitable for other multiscale processes, the terminology of "multiscale simulation" is usually not used in reference to this method.

The other traditional numerical approaches for inherently multiscale problems focus on the scale of interest and eliminate the effects of other scales. As indicated in [1], such approaches have limitations. If we adopt a macroscale method to simulate a multiscale process, three limitations may occur: accuracy, lack of detail for some special process, and the necessity of introducing empirical closures. On the other hand, if we use micro method—say, MDS—to simulate a multiscale process for the entire geometric domain, the required computer resource is not realistic.

This is where multiscale modeling technique comes in. Here, multiscale modeling implies a coupling technique in which some macro/meso/micro simulation methods are used in different local regimes for numerical simulations of the same engineering problem. The macro numerical methods usually are finite-type methods (FDM, FVM, FEM, or FAM). By mesoscale and microscale methods we usually mean the LBM, DSMC, MDS, and QMS, respectively. In such a multiscale simulation, the key issue is how to transfer information between two neighboring regimes in order to guarantee numerical stability, accuracy, and efficiency of computations.

A very typical multiscale simulation was provided recently by Abraham [5], who applied three levels of simulation schemes, FEM-MDS-QMS, to study crack dynamics. Nie et al. [6] adopted the MDS and FVM for the lid-driven cavity flow in which the two vertexes are single point in math. Dupuis et al. [7] proposed an LB-MD model for simulations of flows of liquid argon past and through a carbon nanotube. Wu et al. [8] proposed a scheme of coupled DSMC-NS using an unstructured mesh. The numerical approaches adopted in [5–8] have one thing in common: For the same problem, different regions are solved by different numerical methods and the results are coupled at their interfaces. In this article, this approach is adopted and the focus is on how to transfer information at the interface between the results of the LBM and those of the finite-type methods.

In the following, we first present a brief overview of the lattice Boltzmann model and the finite-volume model (Section 2). Then, we describe the basics and implement procedures of the couple strategy (Section 3). After that, we employ this couple strategy to solve the backward-facing step flow, flow around a circular cylinder, and lid-driven cavity flow (Section 4). Finally, some conclusions are given.

2. LATTICE BOLTZMANN MODEL AND FINITE-VOLUME MODEL

2.1. Lattice Boltzmann Model

The lattice Boltzmann method can be easily coupled to the finite-type methods for continuum partial differential equations, partly because of its small time steps and geometric flexibility [2]. The LBM combines the power of continuum methods with the geometric flexibility of the atomistic method, which is a bridge between macroscale and microscale numerical methods. To play this bridge role the LBM should be coupled downward with micro- and upward with macroscopic methods. Taking the upward coupling into consideration, the transfer of LBM results to macro results is very easy. However, the transfer of macro results to the particle distribution function of the LBM is not straightforward. This article offers the details of coupling the macroscale results with the LBM results.

A popular kinetic model of the LBM adopted in the literature is the single-relaxation-time (SRT) approximation, the so-called Bhatnagar-Gross-Krook (BGK) model [9],

$$\frac{\partial f}{\partial t} + \boldsymbol{c} \cdot \nabla f = -\frac{1}{\lambda}\left(f - f^{(\text{eq})}\right) \tag{1}$$

where f is the single-particle distribution function, ∇f is the gradient of the function f, $\boldsymbol{c}$ is the particle velocity vector, $f^{(\text{eq})}$ is the equilibrium distribution function (the Maxwell-Boltzmann distribution function), and λ is the relaxation time.

To solve for f numerically, Eq. (1) is first discretized in the velocity space using a finite set of velocities $\{c_i\}$ without affecting the conservation laws [10, 11], giving

$$\frac{\partial f_i}{\partial t} + \boldsymbol{c}_i \cdot \nabla f_i = -\frac{1}{\lambda}\left(f_i - f_i^{(\text{eq})}\right) \tag{2}$$

In the above equation, $f_i(x, t) \equiv f(x, c_i, t)$ is the distribution function associated with the ith discrete velocity c_i, and $f_i^{(\text{eq})}$ is the ith equilibrium distribution function. The nine-velocity square lattice model D2Q9 [12] (Figure 1) has been used successfully for simulating 2-D flow. The nine velocities, denoted by c_i, are given by

$$\begin{aligned}
\boldsymbol{c}_0 &= 0 \\
\boldsymbol{c}_i &= c\{\cos[(i-1)\pi/4],\ \sin[(i-1)\pi/4]\} \qquad \text{for } i = 1, 2, 3, 4 \\
\boldsymbol{c}_i &= \sqrt{2}c\{\cos[(i-1)\pi/4],\ \sin[(i-1)\pi/4]\} \qquad \text{for } i = 5, 6, 7, 8
\end{aligned} \tag{3}$$

where $c = \Delta x/\Delta t$, Δx is the lattice spacing step size, and Δt is the time-step size. The equilibrium distribution function is given by

$$f_i^{(\text{eq})} = \omega_i \rho\left[1 + \frac{3}{c^2}\left(\boldsymbol{c}_i \cdot \boldsymbol{u}\right) + \frac{9}{2c^4}\left(\boldsymbol{c}_i \cdot \boldsymbol{u}\right)^2 - \frac{3}{2c^2}\boldsymbol{u}^2\right] \tag{4}$$

with the weights $\omega_0 = 4/9$, $\omega_1 = \omega_2 = \omega_3 = \omega_4 = 1/9$, $\omega_5 = \omega_6 = \omega_7 = \omega_8 = 1/36$.
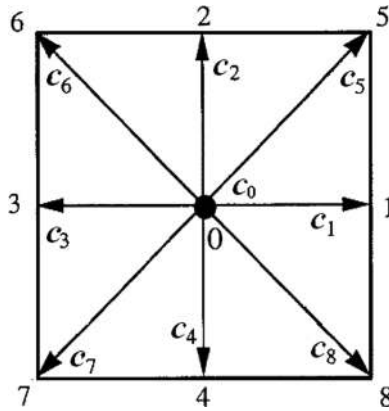
Figure 1. A 2-D, 9-velocity (D2Q9) lattice model.

The macroscopic density ρ and velocity vector $\boldsymbol{u}$ can be evaluated as

$$\rho = \sum_{i=0}^{8} f_i \tag{5a}$$

$$\rho \boldsymbol{u} = \sum_{i=0}^{8} \boldsymbol{c}_i f_i \tag{5b}$$

The pressure of an ideal gas can be calculated from $p = \rho c_s^2$, with the speed of sound being $c_s = c/\sqrt{3}$. In the LBM, Eq. (2) is discretized in both time and space, and the completely discretized equation (also the evolution equation) is

$$f_i(\mathbf{x} + \boldsymbol{c}_i\Delta t, t + \Delta t) = f_i(\mathbf{x}, t) - \frac{1}{\tau}\Big[f_i(\mathbf{x}, t) - f_i^{(\mathrm{eq})}(\mathbf{x}, t) \Big] \tag{6}$$

where $\tau = \lambda/\Delta t$.

Equation (6) can be divided into two substeps: (1) collision, which occurs when particles at a node interact with each other and then change velocity directions according to scattering rules; and (2) streaming, in which each particle moves to the nearest node in the velocity direction.

Collision step:

$$\tilde{f}_i(\boldsymbol{x}, t) = f_i(\boldsymbol{x}, t) - \frac{1}{\tau}\Big[f_i(\boldsymbol{x}, t) - f_i^{(\mathrm{eq})}(\boldsymbol{x}, t) \Big] \tag{7a}$$

Streaming step:

$$f_i(\boldsymbol{x} + \boldsymbol{c}_i\Delta t, t + \Delta t) = \tilde{f}_i(\boldsymbol{x}, t) \tag{7b}$$

where $\tilde{f}_i$ represents the postcollision state.

2.2. Finite-Volume Method

For multiscale simulation, a fast-converging algorithm of the continuum method is highly required. Among the finite-type methods, the FVM is the most widely adopted one in numerical heat transfer for its conservation properties of the discretized equation and the clear physical meaning of the coefficients. Generally speaking, the macroscale methods (called continuum methods hereafter) obey the fundamental laws of conservation of mass, momentum, and energy.

The corresponding differential equation of the conservation law is

$$\frac{\partial}{\partial t}(\rho \phi) + \operatorname{div}(\rho \boldsymbol{U} \phi) = \operatorname{div}\left(\Gamma_\phi \operatorname{grad} \phi\right) + S_\phi \tag{8}$$

where ϕ is the dependent variable (such as velocity, or temperature), U is the velocity vector, ρ is the fluid density, Γ is the nominal diffusion coefficient, and S_ϕ is the source term.

In 1972, Patankar and Spalding proposed a solution procedure called SIMPLE, which is the most widely adopted algorithm for dealing with the coupling between velocity and pressure. There are two major assumptions in the simple algorithm: (1) the initial pressure and initial velocity are independently assumed, leading to some inconsistency between p and u, v; and (2) when the velocity correction equation is derived, the effects of the neighboring grids' velocity corrections are totally neglected. These two assumptions do not affect the final solution but do affect the convergence rate. The first assumption was overcome by SIMPLER of Patankar (1980) [13]. Researchers have made many efforts to overcome the second assumption, such as SIMPLEC by van Doormaal and Raithby (1984) [14], PISO by Issa (1986) [15], the explicit correction-step method by Yen and Liu (1993) [16], and MSIMPLER by Yu et al. (2001) [17]. None of the above revised versions could successfully overcome the second assumption. In recent years, our group developed CLEAR [18, 19], and IDEAL [20, 21]. They completely delete the second assumption, making the algorithm fully implicit. In both the CLEAR and IDEAL algorithms, the improved pressure and velocity are solved directly, rather than by adding a correction term to the intermediate solution. A further improvement of the solution procedure is developed in the IDEAL algorithm, making its convergence rate and robustness better than that of CLEAR.

In this article, the 2-D IDEAL collocated grid algorithm is adopted [20, 21], and the SGSD scheme is used for the discretization of the convective term [22].

3. PRINCIPLE AND PROCEDURE OF THE CFVLBM

3.1. Coupling Principle

First we present a general framework for designing a numerical method that couples marco methods and meso/micro methods. Assume that a macroscale process is described by a state variable U and a microscopic process is described by a state variable u. The two processes and state variables are related to each other at the interface of the macro and micro models by compression and reconstruction operations, denoted by C (compression operator) and R (reconstruction operator),

respectively, as follows [23]:

$$U(\boldsymbol{x}, t) = C[\boldsymbol{u}(\boldsymbol{x}, t)] \tag{9}$$

$$\boldsymbol{u}(\boldsymbol{x}, t) = R[\boldsymbol{U}(\boldsymbol{x}, t)] \tag{10}$$

The two operators have the property $CR = I$, where I is the identity operator.

Generally speaking, the reconstruction operator is not unique. In fact, the reconstruction procedure leads to a one-to-many mapping, because the microscopic simulator contains more information than that of the macroscopic simulator. On the contrary, the compression operation is usually a local/ensemble average through which a unique parameter can be obtained from lots of micro/meso-scale information. Thus, for the coupling between the results from macroscopic and mesoscopic methods, the major difficulty is how to transform the macroscopic results, such as velocity, into the dependent variables adopted in the micro/meso-scale methods, for example, from velocity of a finite-type method to the single-particle distribution function of the LBM.

An analytic expression has been derived to solve the above problem [24]. For the readers' convenience, the derivation process is presented below. In the following derivation the fluid is assumed to be incompressible, hence the fluctuation of the density is neglected.

According to the Chapman-Enskog method [25], we can introduce the following time and space scale expansion:

$$\partial_t = \varepsilon \partial_t^{(1)} + \varepsilon^2 \partial_t^{(2)} \tag{11a}$$

$$\partial_{x_\alpha} = \varepsilon \partial_{x_\alpha}^{(1)} \tag{11b}$$

The small expansion parameter ε can be viewed as the Knudsen number, Kn, which is the ratio between the mean free path and the characteristic length scale of the flow, and α represents the two coordinate directions.

The distribution f_i is expanded around the distributions $f_i^{(0)}$ as follows:

$$f_i = f_i^{(0)} + \varepsilon f_i^{(1)} + \varepsilon^2 f_i^{(2)} \tag{12}$$

with

$$\sum_i f_i^{(1)} = 0 \qquad \sum_i \boldsymbol{c}_i f_i^{(1)} = 0 \qquad \sum_i f_i^{(2)} = 0 \qquad \sum_i \boldsymbol{c}_i f_i^{(2)} = 0 \tag{13a-d}$$

Then, the $f_i(\boldsymbol{x} + \boldsymbol{c}_i \Delta t, t + \Delta t)$ in Eq. (6) is expanded about $\boldsymbol{x}$ and t, which gives

$$f_i(\boldsymbol{x} + \boldsymbol{c}_i \Delta t, t + \Delta t) = f_i(\boldsymbol{x}, t) + \Delta t D_{i\alpha} f_i(\boldsymbol{x}, t) + \frac{(\Delta t)^2}{2} D_{i\alpha}^2 f_i(\boldsymbol{x}, t) + \mathrm{O}[(\Delta t)^3] \tag{14}$$

where $D_{i\alpha} = \partial_t + \boldsymbol{c}_i \partial_{x_\alpha}$ for concise expression.

The expansion of Eq. (14) is substituted into Eq. (6), which gives

$$\Delta t D_{i\alpha} f_i + \frac{(\Delta t)^2}{2} D_{i\alpha}^2 f_i = -\frac{1}{\tau}(f_i - f_i^{(\text{eq})}) + \mathrm{O}[(\Delta t)^3] \qquad (15)$$

After substituting Eqs. (11a), (11b), and (12) into Eq. (15), the following equation can be obtained:

$$\varepsilon D_{i\alpha}^{(1)} f^{(0)} + \varepsilon^2 \left(D_{i\alpha}^{(1)} f_i^{(1)} + \partial_t^{(2)} f_i^{(0)} \right) + \varepsilon^2 \frac{\Delta t}{2} \left(D_{i\alpha}^{(1)} \right)^2 f_i^{(0)}$$
$$= -\frac{1}{\Delta t \tau} \left(f_i^{(0)} + \varepsilon f_i^{(1)} + \varepsilon^2 f_i^{(2)} - f_i^{(eq)} \right) + \mathrm{O}[(\Delta t)^3] \qquad (16)$$

Then, by matching the scales of ε^0, ε^1, and ε^2, we have

$$\varepsilon^0: \qquad f_i^{(0)} = f_i^{(\text{eq})} \qquad (17)$$

$$\varepsilon^1: \qquad f_i^{(1)} = -\Delta t \tau D_{i\alpha}^{(1)} f_i^{(0)} + \mathrm{O}[(\Delta t)^2] \qquad (18)$$

$$\varepsilon^2: \qquad f_i^{(2)} = -\Delta t \tau \left[D_{i\alpha}^{(1)} f_i^{(1)} + \partial_t^{(2)} f_i^{(0)} \right] - \tau \frac{(\Delta t)^2}{2} \left(D_{i\alpha}^{(1)} \right)^2 f_i^{(0)} + \mathrm{O}[(\Delta t)^3] \qquad (19)$$

Considering Eqs. (5a) and (5b), we can sum Eq. (18) over the phase space. Then the first order of the continuity equation and momentum equation can be derived [26] as

$$\varepsilon^1: \qquad \partial_t^{(1)} \rho + \partial_{x_\alpha}^{(1)} (\rho u_\alpha) + \mathrm{O}[(\Delta t)^2] = 0 \qquad (20a)$$

$$\partial_t^{(1)} (\rho u_\alpha) + \partial_{x_\beta}^{(1)} \left(\rho u_\alpha u_\beta + p \delta_{\alpha\beta} \right) + \mathrm{O}[(\Delta t)^2] = 0 \qquad (20b)$$

In the same way, we can obtain the second order of the continuity equation and momentum equation according to Eq. (19):

$$\varepsilon^2: \qquad \partial_t^{(2)} \rho + \mathrm{O}[(\Delta t)^3] = 0 \qquad (21a)$$

$$\partial_t^{(2)} (\rho u_\alpha) - \nu \partial_{x_\beta}^{(1)} \left[\rho \left(\partial_{x_\alpha}^{(1)} u_\beta + \partial_{x_\beta}^{(1)} u_\alpha \right) \right] + \mathrm{O}[(\Delta t)^3] = 0 \qquad (21b)$$

In the following, we introduce formulas according to the chain rule of derivatives:

$$\partial_t f_i^{(\text{eq})} = \partial_\rho f_i^{(\text{eq})} \partial_t \rho + \partial_{u_\beta} f_i^{(\text{eq})} \partial_t u_\beta \qquad (22a)$$

$$\partial_{x_\alpha} f_i^{(\text{eq})} = \partial_\rho f_i^{(\text{eq})} \partial_{x_\alpha} \rho + \partial_{u_\beta} f_i^{(\text{eq})} \partial_{x_\alpha} u_\beta \qquad (22b)$$

From Eq. (4), we can get that

$$\partial_{u_\beta} f_i^{(eq)} = \omega_i \rho \left[\frac{1}{c_s^2}(c_{i\beta} - u_\beta) + \frac{1}{c_s^4} c_{i\alpha} c_{i\beta} u_\alpha \right] \tag{23}$$

$$\partial_\rho f_i^{(eq)} = \frac{1}{\rho} f_i^{(eq)} \tag{24}$$

Furthermore, substituting Eqs. (20)–(24) into Eq. (18) gives the first-order expression of the distribution function f:

$$
\begin{aligned}
f_i^{(1)} &= -\tau \Delta t \left(\partial_t^{(1)} f_i^{(0)} + c_i \partial_{x_\alpha}^{(1)} f_i^{(0)} \right) \\
&= -\tau \Delta t \left[\partial_\rho f_i^{(0)} \partial_t^{(1)} \rho + \partial_{u_\beta} f_i^{(0)} \partial_t^{(1)} u_\beta + c_i \left(\partial_\rho f_i^{(0)} \partial_{x_\alpha}^{(1)} \rho + \partial_{u_\beta} f_i^{(0)} \partial_{x_\alpha}^{(1)} u_\beta \right) \right] \\
&= -\tau \Delta t \left[-\partial_\rho f_i^{(0)} \partial_{x_\alpha}^{(1)} (\rho u_\alpha) - \frac{1}{\rho} \partial_{u_\beta} f_i^{(0)} \partial_{x_\alpha}^{(1)} (\rho u_\alpha u_\beta + p \delta_{\alpha\beta}) \right. \\
&\quad \left. + c_i \left(\partial_\rho f_i^{(0)} \partial_{x_\alpha}^{(1)} \rho + \partial_{u_\beta} f_i^{(0)} \partial_{x_\alpha}^{(1)} u_\beta \right) \right] \\
&= -\tau \Delta t \left(U_{i\alpha} \partial_{x_\alpha}^{(1)} \rho \partial_\rho f_i^{(0)} + U_{i\alpha} \partial_{x_\alpha}^{(1)} u_\beta \partial_{u_\beta} f_i^{(0)} - \rho \partial_\rho f_i^{(0)} \partial_{x_\alpha}^{(1)} u_\alpha - \frac{1}{\rho} \partial_{x_\alpha}^{(1)} p \partial_{u_\alpha} f_i^{(0)} \right) \\
&= -\tau \Delta t \left[U_{i\alpha} f_i^{(0)} \frac{1}{\rho} \partial_{x_\alpha}^{(1)} \rho + U_{i\alpha} \omega_i \rho \left(\frac{1}{c_s^2} U_{i\beta} + \frac{1}{c_s^4} c_{i\beta} c_{i\gamma} u_\gamma \right) \partial_{x_\alpha}^{(1)} u_\beta - f_i^{(0)} \partial_{x_\alpha}^{(1)} u_\alpha \right. \\
&\quad \left. - \omega_i \left(\frac{1}{c_s^2} U_{i\alpha} + \frac{1}{c_s^4} c_{i\alpha} c_{i\gamma} u_\gamma \right) \partial_{x_\alpha}^{(1)} p \right] \tag{25}
\end{aligned}
$$

where $U_{i\alpha} = c_{i\alpha} - u_\alpha$.

The second-order expression of f in Eq. (19) is calculated as follows:

$$
\begin{aligned}
f_i^{(2)} &= -\Delta t \tau \left(D_{i\alpha}^{(1)} f_i^{(1)} + \partial_t^{(2)} f_i^{(0)} \right) - \frac{(\Delta t)^2}{2} \tau \left(D_{i\alpha}^{(1)} \right)^2 f_i^{(0)} \\
&= -\Delta t \tau \left[D_{i\alpha}^{(1)} \left(-\tau \Delta t D_{i\alpha}^{(1)} f_i^0 \right) + \partial_t^{(2)} f_i^{(0)} \right] - \frac{(\Delta t)^2}{2} \tau \left(D_{i\alpha}^{(1)} \right)^2 f_i^{(0)} \\
&= -\Delta t \tau \partial_t^{(2)} f_i^{(0)} + (\Delta t)^2 \tau \left(\tau - \frac{1}{2} \right) \left(D_{i\alpha}^{(1)} \right)^2 f_i^{(0)} \tag{26}
\end{aligned}
$$

We can ignore the second-order derivative of f_i^0; then

$$f_i^{(2)} = -\Delta t \tau \partial_t^{(2)} f_i^{(0)} \tag{27}$$

By the chain rule of derivatives,

$$
\begin{aligned}
\partial_t^{(2)} f_i^{(0)} &= \partial_\rho f_i^{(0)} \partial_{t_2} \rho + \partial_{u_\beta} f_i^{(0)} \partial_{t_2} u_\beta \\
&= \partial_{u_\beta} f_i^{(0)} \partial_{t_2} u_\beta \tag{28}
\end{aligned}
$$

Using Eqs. (21b) and (23), we get

$$
\begin{aligned}
\partial_t^{(2)} f_i^{(0)} &= \partial_{u_\beta} f_i^{(0)} \partial_{t_2} u_\beta \\
&= \frac{1}{\rho} \partial_{u_\beta} f_i^{(0)} \partial_{t_2} (\rho u_\beta) \\
&= \nu \omega_i \left[\frac{1}{c_s^2} (c_{i\beta} - u_\beta) + \frac{1}{c_s^4} c_{i\alpha} c_{i\beta} u_\alpha \right] \partial_{x_\alpha}^{(1)} \left[\rho \left(\partial_{x_\alpha}^{(1)} u_\beta + \partial_{x_\beta}^{(1)} u_\alpha \right) \right] \\
&= \nu \omega_i \rho \left[\frac{1}{c_s^2} (c_{i\beta} - u_\beta) + \frac{1}{c_s^4} c_{i\alpha} c_{i\beta} u_\alpha \right] \\
&\quad \times \left[\frac{1}{\rho} \partial_{x_\alpha}^{(1)} \rho (\partial_{x_\alpha}^{(1)} u_\beta + \partial_{x_\beta}^{(1)} u_\alpha) + \partial_{x_\alpha}^{(1)} (\partial_{x_\alpha}^{(1)} u_\beta + \partial_{x_\beta}^{(1)} u_\alpha) \right]
\end{aligned} \tag{29}
$$

So we can obtain

$$
\begin{aligned}
f_i^{(2)} &= -\Delta t \tau \nu \omega_i \rho \left(\frac{1}{c_s^2} U_{i\beta} + \frac{1}{c_s^4} c_{i\alpha} c_{i\beta} u_\alpha \right) \\
&\quad \times \left\{ \frac{1}{\rho} \partial_{x_\alpha}^{(1)} \rho \left(\partial_{x_\alpha}^{(1)} u_\beta + \partial_{x_\beta}^{(1)} u_\alpha \right) + \partial_{x_\alpha}^{(1)} \left[\left(\partial_{x_\alpha}^{(1)} u_\beta + \partial_{x_\beta}^{(1)} u_\alpha \right) \right] \right\}
\end{aligned} \tag{30}
$$

Here, we introduce an approximation of $\partial_{u_\beta} f_i^{(0)}$ by dropping terms of a higher order than u^2 as follows:

$$
\partial_{u_\beta} f_i^{(0)} = \omega_i \rho \left(\frac{1}{C_s^2} U_{i\beta} + \frac{1}{C_s^4} c_{i\beta} c_{i\gamma} u_\gamma \right) \approx \frac{U_{i\beta}}{C_s^2} f_i^{(0)} \tag{31}
$$

Assume the velocity fields is divergence-free, written as

$$
\partial_{x_\alpha} u_\alpha = 0 \tag{32}
$$

According to Eqs. (31) and (32), we can rewrite the expressions of $f_i^{(1)}$ and $f_i^{(2)}$, as

$$
\begin{aligned}
f_i^{(1)} &= -\tau \Delta t \left(U_{i\alpha} f_i^{(0)} \frac{1}{\rho} \partial_{x_\alpha}^{(1)} \rho + U_{i\alpha} U_{i\beta} f_i^{(0)} \frac{1}{c_s^2} \partial_{x_\alpha}^{(1)} u_\beta - U_{i\alpha} f_i^{(0)} \frac{1}{\rho c_s^2} \partial_{x_\alpha}^{(1)} p \right) \\
&= -\tau \Delta t U_{i\alpha} U_{i\beta} f_i^{(0)} c_s^{-2} \partial_{x_\alpha}^{(1)} u_\beta
\end{aligned} \tag{33}
$$

$$
\begin{aligned}
f_i^{(2)} &= -\Delta t \tau \nu U_{i\beta} f_i^{(0)} c_s^{-2} \left[\frac{1}{\rho} \partial_{x_\alpha}^{(1)} \rho \left(\partial_{x_\alpha}^{(1)} u_\beta + \partial_{x_\beta}^{(1)} u_\alpha \right) + \left(\partial_{x_\alpha}^{(1)} \right)^2 u_\beta \right] \\
&= -\Delta t \tau \nu U_{i\beta} f_i^{(0)} c_s^{-2} \left[\frac{1}{\rho} S_{\alpha\beta}^{(1)} \partial_{x_\alpha}^{(1)} \rho + \left(\partial_{x_\alpha}^{(1)} \right)^2 u_\beta \right]
\end{aligned} \tag{34}
$$

where $S_{\alpha\beta} = \partial_{x_\beta} u_\alpha + \partial_{x_\alpha} u_\beta$.

At last, we can obtain the expression of f_i:

$$
\begin{aligned}
f_i &= f_i^{(0)} + \varepsilon f_i^{(1)} + \varepsilon^2 f_i^{(2)} \\
&= f_i^{(0)} - \tau \Delta t U_{i\alpha} U_{i\beta} f_i^{(0)} c_s^{-2} \partial_{x_\alpha} u_\beta - \tau \Delta t \nu U_{i\beta} f_i^{(0)} c_s^{-2} \left(\frac{1}{\rho} S_{\alpha\beta} \partial_{x_\alpha} \rho + \partial_{x_\alpha}^2 u_\beta \right) \\
&= f_i^{(eq)} \left[1 - \tau \Delta t U_{i\beta} c_s^{-2} \left(U_{i\alpha} \partial_{x_\alpha} u_\beta + \nu \partial_{x_\alpha}^2 u_\beta + \nu \rho^{-1} S_{\alpha\beta} \partial_{x_\alpha} \rho \right) \right]
\end{aligned} \tag{35}
$$

Equation (35) is an analytic expression for reconstructing the micro variables from the macro variables. The reconstruction operation is essential to establish the information exchange from the macro solver to the micro solver and is very useful to construct a reasonable initial field for accelerating the microscopic computation.

3.2. Computational Procedure

To illustrate the basic idea of the CFVLBM, the computational domain is decomposed into two regions in which the LBM and FVM are used separately. In Figure 2, the decomposition of the computational domain for the lid-driven cavity flow is shown schematically. We can easily control the coarseness and fineness of grids according to the zone spatial scale in each region. When the grid systems at the interface of the subregions are not identical, space interpolation at the interface is required when transferring the information at the interface. In this article, for convenience we choose the FVM grid size equal to the lattice size, to avoid the spatial interpolation. Line $\overline{MN}$ is the FVM region boundary located in the LBM subregion, while line $\overline{AB}$ is the LBM region boundary located in the FVM subregion. Hence the subdomain between the two lines is the overlapped region (often called the hand-shaking region) where both the LBM and FVM methods are adopted. This arrangement of the interface is convenient for the information exchange between the two neighboring regions [27].

The CFVLBM computation procedures are conducted as follows.

Step 1. With some arbitrary assumed velocity at the line $\overline{MN}$, the FVM simulation is performed in the lower region.
Step 2. After a temporary solution is obtained, the information at the line $\overline{AB}$ is transformed into the single-particle distribution function.
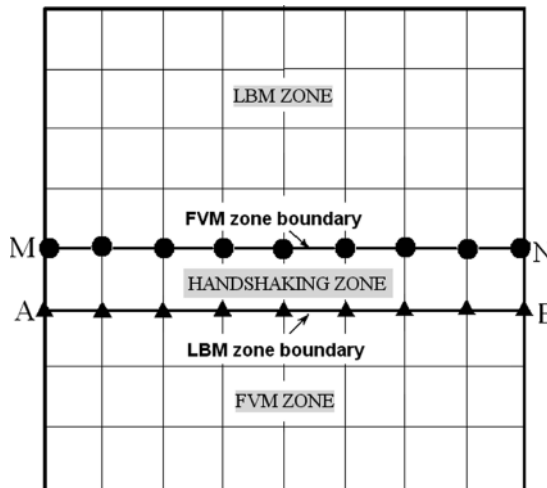Step 3. The LBM simulation is carried out in the upper region.

Figure 2. Interface structure between two regions of FVM and LBM.

Step 4. The temporary solution of the LBM at the line $\overline{MN}$ is transported into the macro velocity and the FVM simulation is repeated.

Step 5. Computation is repeated until the results at the two lines remain the same within an allowed tolerance.

4. RESULTS AND DISCUSSION

In this section, numerical results for three examples are presented, which are solved by the CFVLBM with the reconstruction operator shown in Eq. (35).

4.1. Backward-Facing Step

The problem of a viscous flow over an isothermal, two-dimensional, backward-facing step is a standard test problem, in which the dependence of the reattachment length x_r on the Reynolds number is usually taken as the criterion for comparison. The geometry and boundary conditions for this flow are shown in Figure 3, where a 2-D Cartesian coordinate system is also presented. The downstream channel is defined to have unit height $H = 1$ with a step height S equally to $H/2$. The downstream outflow boundary is located at $x = 15H$ for $\mathrm{Re} = 50$, and at $x = 20H$ for $\mathrm{Re} = 100$. No-slip condition is specified for all solid surfaces. The outlet flow is assumed to be fully developed. At the inflow boundary, located at the step, a parabolic profile is prescribed by $u_0(y) = 1.2(y - H/2)(H - y)$ for $H/2 \leq y \leq H$. This produces a maximum inflow velocity of $u_{\max} = 0.075$ and an average inflow velocity of $u_{\mathrm{avg}} = 0.05$. In this case the maximum Mach number is 0.129. The computational domain is divided into two zones, the LBM zone and the FVM zone, which are partly overlapped (see Figure 3). Uniform grids are adopted with grid size $\Delta x = \Delta y = 0.01H$.

For $\mathrm{Re} = 50$ and 100, the predicted reattachment lengths are $x_r/S = 1.76$ and 3.0, respectively, and the corresponding results reported by Armaly et al. [28] are $x_r/S = 1.8$ and 3.1, respectively. The agreement is reasonably good because the flow predicted in the LBM zone is not completely incompressible.
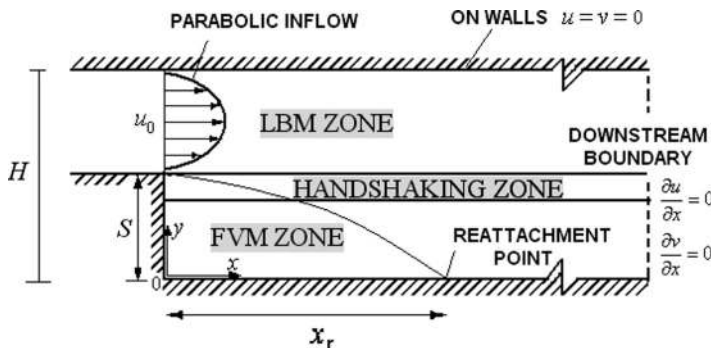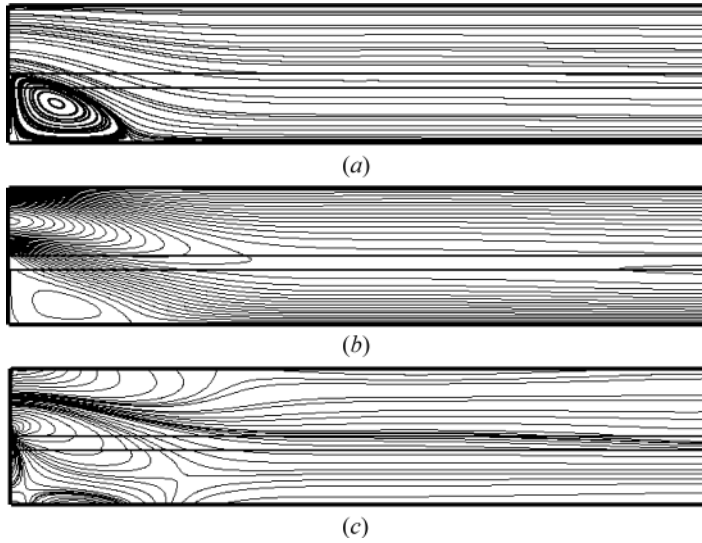

Figure 3. Backward-facing step flow.

Figure 4. Contour lines for Re = 50: (*a*) streamline; (*b*) *u* velocity; and (*c*) vorticity.

For Re = 50 and 100, Figures 4*a* and 5*a* illustrate the main features of the separated flow by streamline contour. The contours of the streamwise velocity *u* are shown in Figures 4*b* and 5*b*. The corresponding voticity contours ($\omega = \partial v/\partial x - \partial u/\partial y$) are shown in Figures 4*c* and 5*c*. The reasonably good agreement with the reference and the smoothness of contours in the hand-shaking region illustrate the feasibility of the reconstruction operator presented by Eq. (35).
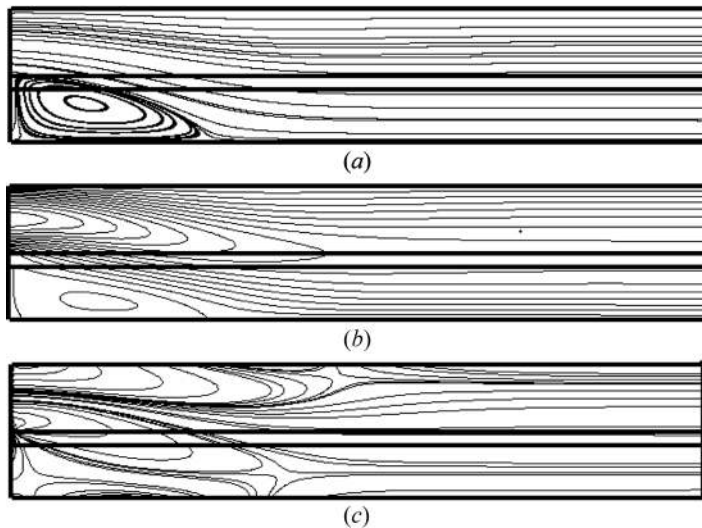

Figure 5. Contour lines for Re = 100: (*a*) streamline; (*b*) *u* velocity; and (*c*) vorticity.

4.2. Flow Around a Circular Cylinder

The second numerical simulation is two-dimensional flow around a circular cylinder for low Reynolds numbers [29–32]. The geometry and boundary conditions for this flow are shown in Figure 6. A uniform velocity $\boldsymbol{u}_0 = (u_\infty, 0)$ is specified along the domain perimeter as physical boundary, and zero velocities are imposed at the cylinder surface. The parameters are defined as height $H = 1.8$, cylinder radius $r = 0.005$, density $\rho = 1.0$, velocity $u_\infty = 0.01$, and grid length $\Delta x = \Delta y = 2 \times 10^{-4}$. The Reynolds number is defined by $\mathrm{Re} = 2u_\infty r/\nu$.

Here three small Reynolds numbers 10, 20, and 40 are chosen to validate the proposed method. Figure 7 shows the streamlines when flow reaches its final steady state. A pair of vortices is observed behind the circular cylinder. For flow around a circular cylinder there are three characteristic parameters: the length of the recirculation region L, the separation angle θ, and the drag coefficient C_D. C_D is defined as

$$C_D = \frac{1}{\rho u_\infty^2 r} \int \boldsymbol{S} \cdot \boldsymbol{n} \, dl \tag{36}$$

where $\boldsymbol{n}$ is the normal direction of the cylinder wall and $\boldsymbol{S}$ is the stress tensor,

$$\boldsymbol{S} = -p\boldsymbol{I} + \rho\upsilon(\nabla\boldsymbol{u} + \boldsymbol{u}\nabla) \tag{37}$$

The drag coefficient C_D and the geometry parameters L and θ are listed in Table 1. All of the parameters predicted by the CFVLBM agree well with the results of previous studies for each Re.
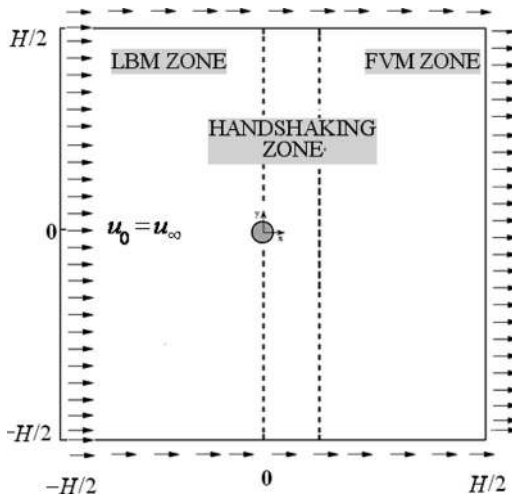
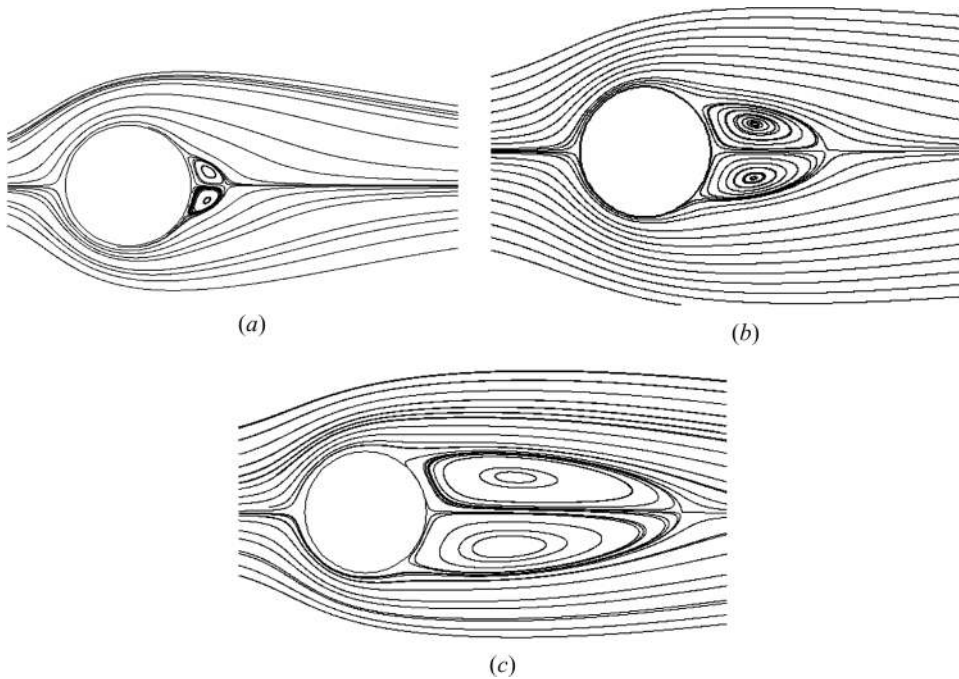

Figure 6. Flow around a circular cylinder.

Figure 7. Streamlines: (*a*) Re = 10; (*b*) Re = 20; and (*c*) Re = 40.

4.3. Lid-Driven Cavity Flow

In this section, a detailed analysis of the capabilities of the reconstruction operator is presented by using the CFVLBM on another popular numerical example,

Table 1. Comparison of previous studies of drag coefficients and geometric parameters for cylinder flow at low Reynolds number

| Re	Authors	Method	C_D	L/r	θ (deg)
10	Dennis and Chang (1970) [29]	N.S.	2.846	0.53	29.6
	He and Doolen (1997) [30]	ISLBM	3.170	0.474	26.89
	Guo and Zhao (2003) [31]	FDLBM	3.049	0.486	28.13
	Imamura et al. (2005) [32]	GILBM	2.848	0.478	26.0
	Present work	CFVLBM	2.810	0.51	29.2
20	Dennis and Chang (1970)	N.S.	2.045	1.88	43.7
	He and Doolen (1997)	ISLBM	2.152	1.842	42.96
	Guo and Zhao (2003)	FDLBM	2.048	1.824	43.59
	Imamura et al. (2005)	GILBM	2.051	1.852	43.3
	Present work	CFVLBM	2.010	1.85	43.2
40	Dennis and Chang (1970)	N.S.	1.522	4.69	53.8
	He and Doolen (1997)	ISLBM	1.499	4.49	52.84
	Guo and Zhao (2003)	FDLBM	1.475	4.168	53.44
	Imamura et al. (2005)	GILBM	1.538	4.454	52.4
	Present work	CFVLBM	1.511	4.44	53.5
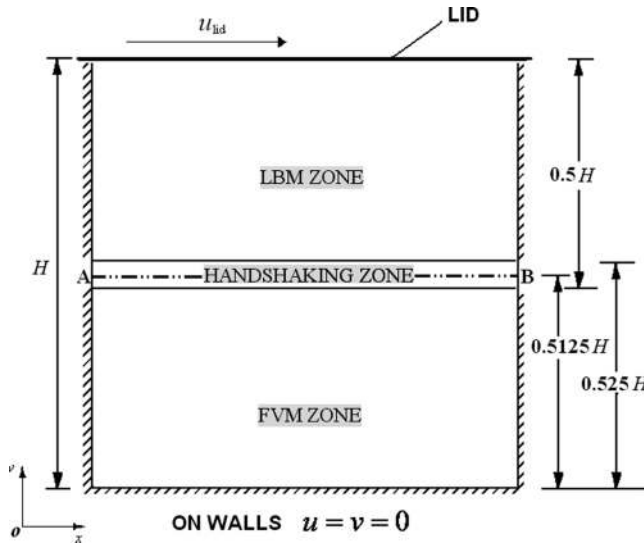
Figure 8. Lid-driven cavity flow.

lid-driven cavity flow. The geometry and boundary conditions are shown in Figure 8. Three numerical simulations were carried out for $Re = 100$, 400, and 1000 on a grid of 400×400. The characteristic length in Reynolds numbers is the height of the squared cavity, $H = 1$. The boundaries of the cavity are still walls, except the upper boundary, for which a uniform tangential velocity is prescribed as $u_{Re=100} = 3.33 \times 10^{-3}$, $u_{Re=400} = 1.33 \times 10^{-3}$, $u_{Re=1000} = 3.33 \times 10^{-2}$ for the three Reynolds numbers, respectively. Corresponding to each case, the Mach numbers are $Ma_{Re=100} = 5.77 \times 10^{-3}$, $Ma_{Re=400} = 2.31 \times 10^{-2}$, and $Ma_{Re=1000} = 5.77 \times 10^{-2}$.

Figure 9 shows the streamlines plots for the Reynolds number considered. These plots give a clear picture of the overall flow pattern and the effect of Reynolds number on the structure of the recirculating eddies in the cavity. The smoothness of the streamline, especially around the hand-shaking region, further confirms the correctness of the information transfer at the interface. In addition to the primary center vortex, a pair of counterrotating eddies of much smaller strength are developed in the lower corners of the cavity. To quantify these results, the center locations of the primary vortices, bottom left vortices, and bottom right vortices are listed in Table 2. The results are in close agreement with the benchmark solution [33]. As Re increases, the primary vortex center moves toward the right and increasingly becomes circular.

Figures 10 and 11 show the contours of u velocity and v velocity. It is seen that these physical quantities are all smooth across the interface. The velocity profiles along the vertical and horizontal centerlines of the cavity are shown in Figures 12 and 13. It is seen that the maximum errors occur at the peak points. This is because the absolute values of velocity are larger at those points. Errors grow as Re increases. This may be partially caused by the weak compressibility of the LBM results. In the LBM zone, the density is not a constant inherently related to the LBM. However, the FVM model adopted here is incompressible. To illustrate this argument more
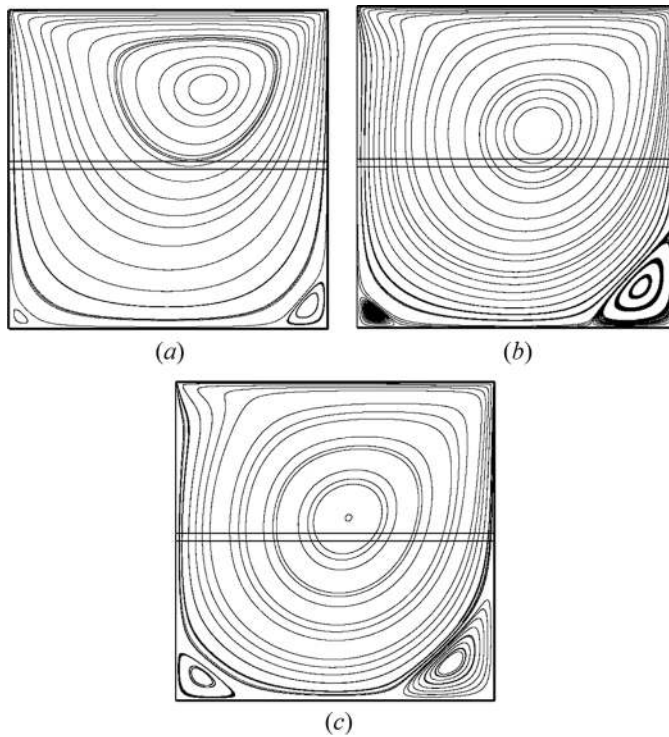
Figure 9. Streamlines: (*a*) Re = 100; (*b*) Re = 400; and (*c*) Re = 1,000.

clearly, Figure 14 shows the density contour lines of the predicted results. We can
see that the density is changing in the LBM zones. When the LBM and incompress-
ible FVM are coupled, flow velocity should not be large, to decrease the com-
pressibility influence by the LBM. All things considered, we conclude that the
results obtained by the CFVLBM are in good agreement with the benchmark work
by Ghia et al. [33].

When the Navier-Stokes equations are used to solve the incompressible flow, it
is crucial to maintain the mass conservation of the entire flow domain. This issue is
especially importance when the coupled method is used. The hand-shaking region is

Table 2. Comparison of vortices location between present results and [33]

| Re/vortices		Primary vortices location (x, y)	Bottom left vortices location (x, y)	Bottom right vortices location (x, y)
Re = 100	Present	(0.615, 0.738)	(0.0325, 0.038)	(0.936, 0.062)
	Ref. (33)	(0.6172, 0.7344)	(0.0313, 0.0391)	(0.9453, 0.0625)
Re = 400	Present	(0.565, 0.609)	(0.051, 0.046)	(0.888, 0.124)
	Ref. (33)	(0.5547, 0.6055)	(0.0508, 0.0469)	(0.8906, 0.1250)
Re = 1,000	Present	(0.537, 0.568)	(0.083, 0.076)	(0.861, 0.111)
	Ref. (33)	(0.5313, 0.5625)	(0.0859, 0.0781)	(0.8594, 0.1094)
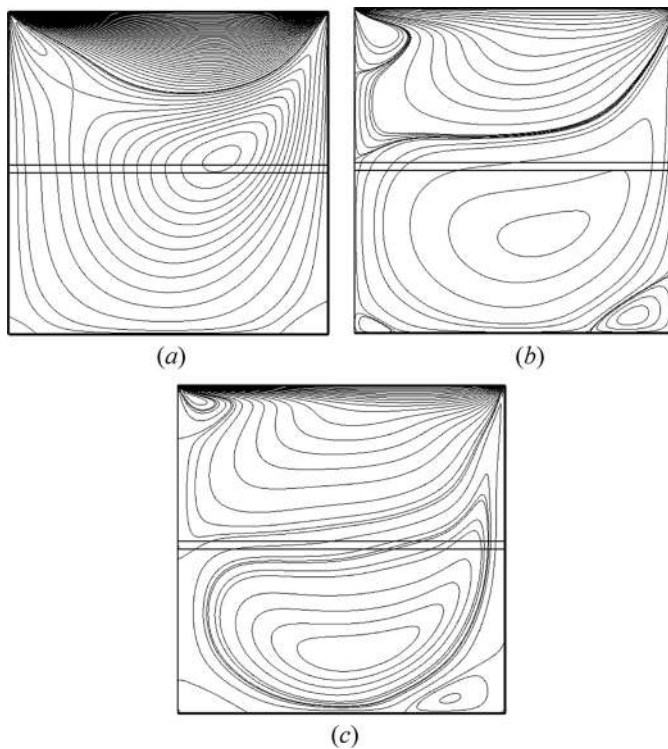
Figure 10. The u-velocity contour lines: (a) Re $=100$; (b) Re $=400$; and (c) Re $=1,000$.

best to examine the conservation of the mass flow rate. For this purpose, Figures 15 and 16 are provide, where the mass fluxes at the interface (i.e., line A–B in Figure 8) at Re $=1,000$ are shown. We can see that the mass fluxes from the LBM and FVM match very well.

The contours of vorticity distribution are now examined, and the results are presented in Figure 17. The plots of vorticity with viscous effects are confined to thin shear layers near the wall. It can be observed that the vorticity contours predicted from the FVM and LBM are in agreement. However, there is some nonsmoothness at the interface. We have tried several numerical methods (refining the grids, improving convergence criteria), but this nonsmoothness could not be fully removed. Thus it is expected to be caused by the weak compressibility and unsteady nature of the LBM. This observation gives us a hint: to confirm good smoothness and consistency when coupling of numerical solutions at the interface is investigated, the quality of the vorticity contour plot is more sensitive to that of velocity. This is because vorticity is the first derivative of velocity; its smoothness requirement is more strict than that of velocity. The stream function is an integral of the velocity; hence its smoothness is the easiest to obtain.

Finally, CPU time to obtain a steady-state solution of the lid-driven cavity flow is discussed. For the FVM, the steady-state solution can be obtained from a steady governing equation with an assumed initial field. This iterative process is in some
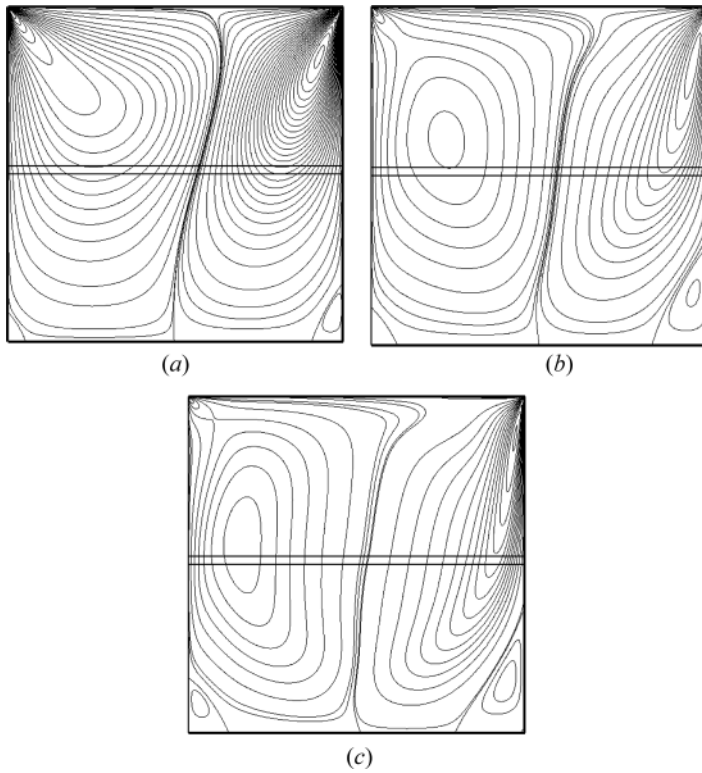
Figure 11. The *v*-velocity contour lines: (*a*) Re = 100; (*b*) Re = 400; and (*c*) Re = 1,000.
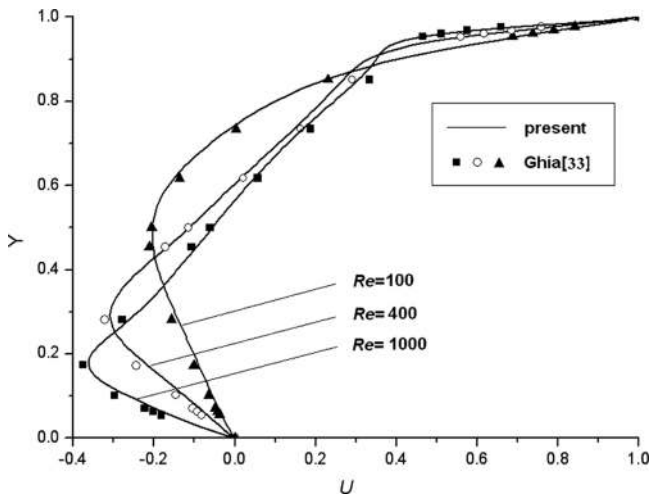

Figure 12. Comparison of *u* velocity along vertical line through geometric center.
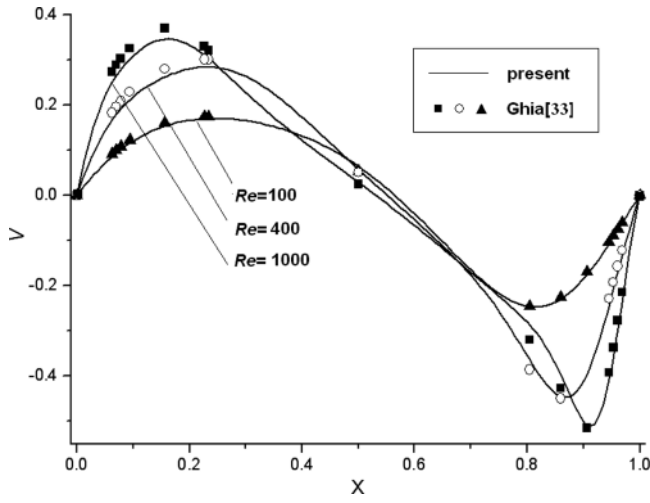
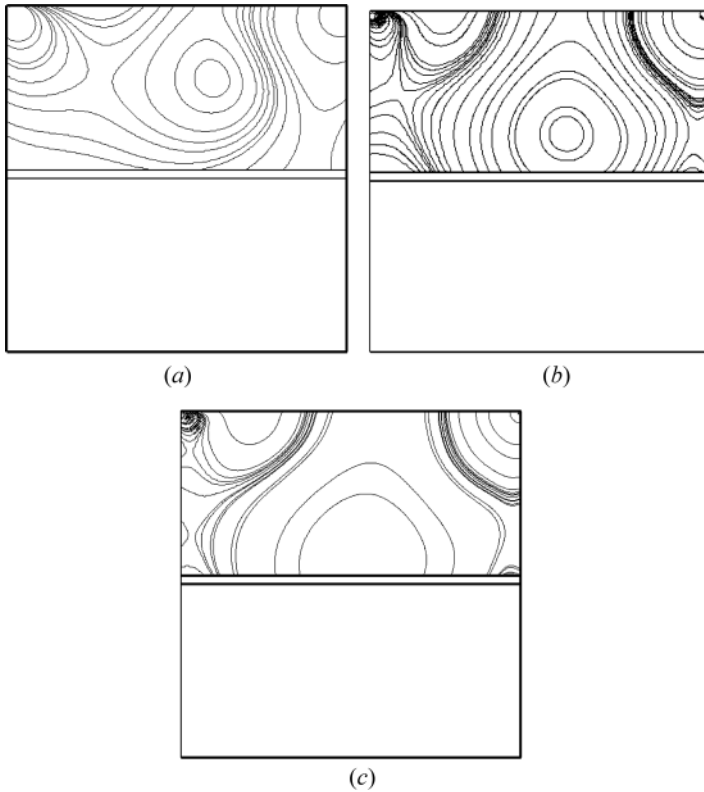**Figure 13.** Comparison of *v* velocity along horizontal line through geometric center.

Figure 14. Density contour lines: (*a*) Re = 100; (*b*) Re = 400; and (*c*) Re = 1,000.
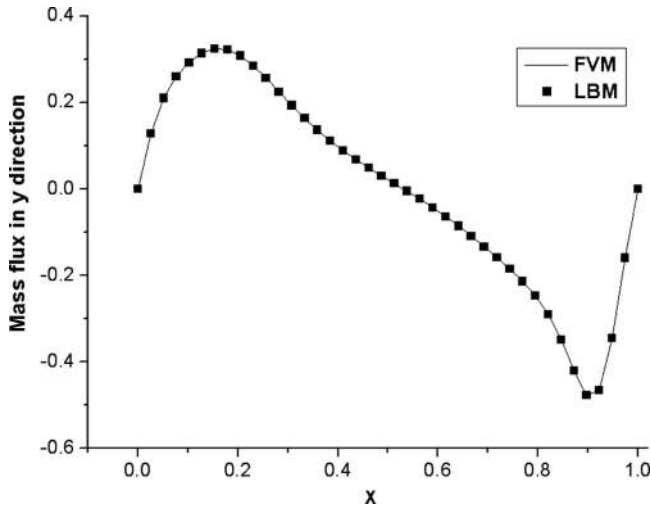
Figure 15. The y component of the mass flux $\rho v/(\rho_0 u_{\text{lid}})$ on the interface AB defined in Figure 8.

sense similar to an unsteady solution process, with one outer iteration being equivalent to marching one time step forward. The LBM is an essentially unsteady approach in which the solution gradually evolves into steady state. In this article the iteration convergence criterion for reaching the steady state is defined by Eq. (38) with err $= 10^{-6}$.

$$\text{Residual} = \frac{\sum\sum |u(i,j,t+\Delta t) - u(i,j,t)| + \sum\sum |v(i,j,t+\Delta t) - v(i,j,t)|}{\sum\sum(|u(i,j,t)| + |v(i,j,t)|)} \leq \text{err} \tag{38}$$

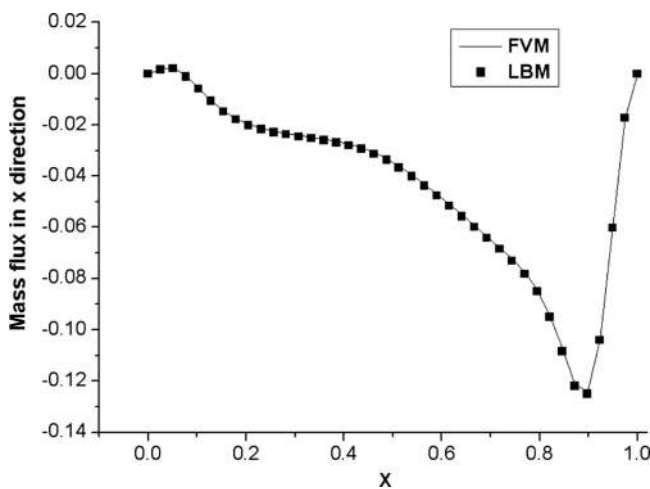**Figure 16.** The x component of the mass flux $\rho u/(\rho_0 u_{\text{lid}})$ on the interface AB.
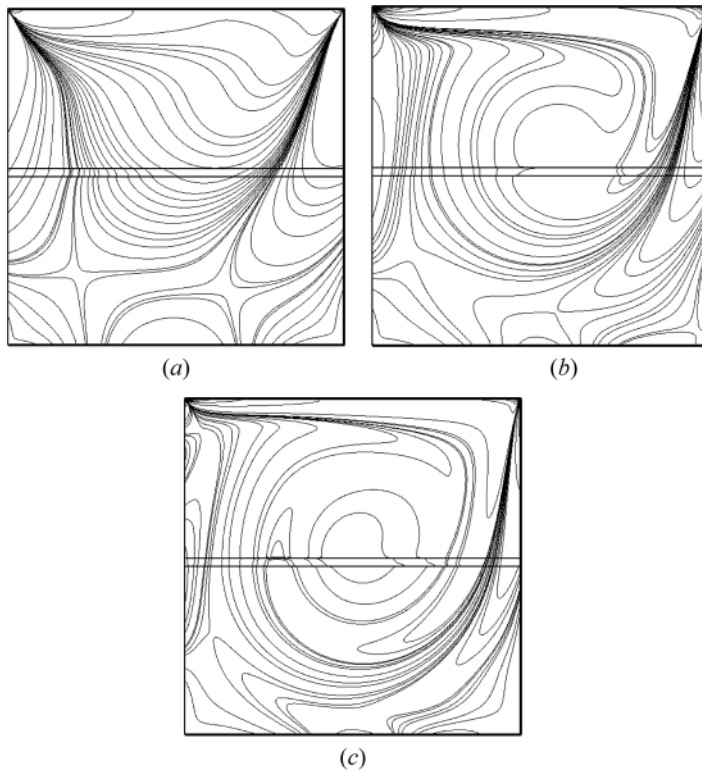
Figure 17. Vorticity contour lines: (*a*) Re = 100; (*b*) Re = 400; and (*c*) Re = 1,000.

The case of Re = 1,000 is selected to compare the residual history for four methods. The first method and the second method are the single FVM method and the single LBM in the whole computation domain, respectively. The third method is denoted by CFVLBM(I), in which the interface information exchange is executed after one outer iteration in the FVM region and one time step forward in the LBM region. In the fourth method, denoted by CFVLBM(II), the interface information exchange is not executed until the interface mass residual defined by Eq. (38) is almost equal to or even less than the mass residual of the entire subregion. Here, by one outer iteration in the FVM we mean that the coefficients of the discretization equation are updated once. The inner iteration means the solution process of the algebraic equation for the given set of coefficients in the discretization equations. For both the CFVLBM(I) and CFVLBM(II), the inner iteration is the same.

When the flow reaches steady state, the flow fields obtained from the four methods are almost identical. However, the residual histories are quite different. Figure 18 shows the residual history of the four methods. It can be seen that the FVM method has the fastest convergence rate, the LBM shows the worst convergence speed, and the CFVLBM(I) and CFVLBM(II) are in between, with the CFVLBM(II) being much better than the CFVLBM(I).
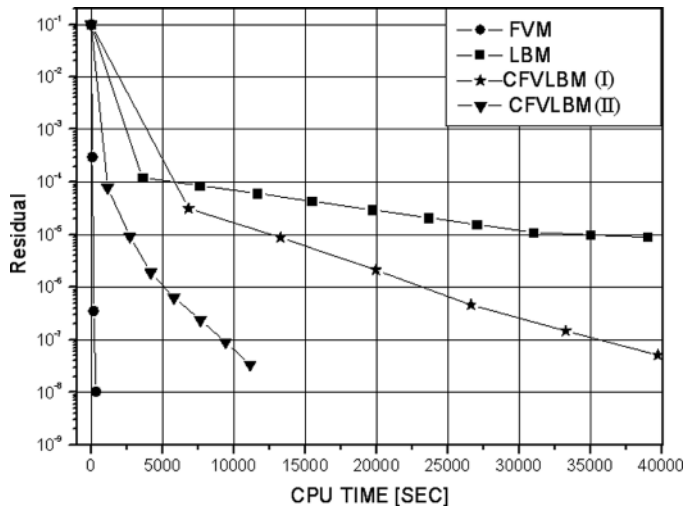
Figure 18. Residual history for the lid-driven cavity flow for Re = 1,000.

5. CONCLUSIONS

In this article, a coupling approach is used to combine the LBM and FVM for simulating fluid flow problems. A reconstruction operator is adopted to lift the macroscopic velocity fields of the FVM to mesoscopic distribution functions of the LBM. Three numerical examples have validated the feasibility and reliability of the proposed reconstruction operator.

The stream function contours are the easiest to smooth. To confirm good smoothness and consistency of solutions coupling at the interface, the vorticity contours are suggested separately from the velocity distributions. The convergence to steady-state flow is expected to be accelerated by the proposed CFVLBM(II).

Extension to higher Mach numbers and 3-D computations is now underway in the authors' group and will be reported later.

REFERENCES

1. E. Weinan, B. Engquist, X. Li, W. Ren, and E. Vanden-Eijnden, Heterogeneous Multi-scale Methods: A Review, *Commun. Comput. Phys.*, vol. 44, pp. 367–450, 2007.
2. S. Succi, O. Filippova, G. Smith, and E. Kaxiras, Applying the Lattice Boltzmann Equation to Multiscale Fluid Problems, *Comput. Sci. Eng.*, vol. 3, pp. 26–37, 2001.
3. Q. H. Zeng, A. B. Yu, and G. Q. Lu, Multiscale Modeling and Simulation of Polymer Nanocomposites, *Prog. Polymer. Sci.*, vol. 33, pp. 191–269, 2008.
4. W. Q. Tao, *Numerical Heat Transfer*, 2nd ed., pp. 1–26, Xi'an Jiaotong University Press, Xi'an, China, 2000.
5. F. F. Abraham, Dynamically Spanning the Length Scales from the Quantum to the Continuum, *Int. J. Mod. Phys. C*, vol. 11, pp. 1135–1148, 2000.
6. X. B. Nie, S. Y. Chen, W. N. E., and M. O. Robbins, A Continuum and Molecular Dynamics Hybrid Method for Micro- and Nano-Fluid Flow, *J. Fluid Mech.*, vol. 500, pp. 55–64, 2004.

7. A. Dupuis, E. M. Kotsalis, and P. Koumoutsakos, Coupling Lattice Boltzmann and Molecular Dynamics Models for Dense Fluids, *Phys. Rev. E.*, vol. 75, pp. 046704, 2007.

8. J. S. Wu, X. Y. Lian, G. Cheng, P. R. Koomullil, and K. C. Tseng, Development and Verification of a Coupled DSMC–NS Scheme Using Unstructured Mesh, *J. Comput. Phys.*, vol. 219, pp. 579–607, 2006.

9. P. L. Bhatnagar, E. P. Gross, and M. Krook, A Model for Collision Processes in Gases, Part I. Small Amplitude Processes in Charged and Neutral One-Component System, *Phys. Rev.*, vol. 94, pp. 511–525, 1954.

10. X. He and L. S. Luo, A Priori Derivation of the Lattice Boltzmann Equation, *Phys. Rev. E*, vol. 55, pp. R6333-6, 1997.

11. X. He and L. S. Luo, Theory of the Lattice Boltzmann Equation: From Boltzmann Equation to Lattice Boltzmann Equation, *Phys. Rev. E.*, vol. 56, pp. 6811–6817, 1997.

12. Y. H. Qian, D. d'Humiéres, and P. Lallemand, Lattice BGK Models for Navier Stokes Equation, *Eur. Phys. Lett.*, vol. 15, pp. 603–607, 1991.

13. S. V. Patankar, A Calculation Procedure for Two-Dimensional Elliptic Situation, *Numer. Heat Transfer*, vol. 4, pp. 409–425, 1981.

14. J. P. van Doormaal and G. D. Raithby, Enhancement of the SIMPLE Method for Predicting Incompressible Fluid Flow, *Numer. Heat Transfer*, vol. 7, pp. 147–163, 1984.

15. R. I. Issa, Solution of the Implicit Discretized Fluid-Flow Equations by Operator Splitting, *J. Comput. Phys.*, vol. 62, pp. 40–65, 1986.

16. R. H. Yen and C. H. Liu, Enhancement of the SIMPLE Algorithm by an Additional Explicit Correction Step, *Numer. Heat Transfer B*, vol. 24, pp. 127–141, 1993.

17. B. Yu, H. Ozoe, and W. Q. Tao, A Modified Pressure-Correction Scheme for the SIMPLER Method, MSIMPLER, *Numer. Heat Transfer B*, vol. 39, pp. 435–449, 2001.

18. W. Q. Tao, Z. G. Qu, and Y. L. He, A Novel Segregated Algorithm for Incompressible Fluid Flow and Heat Transfer Problems—Clear (Coupled and Linked Equations Algorithm Revised) Part I: Mathematical Formulation and Solution Procedure, *Numer. Heat Transfer B*, vol. 45, pp. 1–17, 2004.

19. W. Q. Tao, Z. G. Qu, and Y. L. He, A Novel Segregated Algorithm for Incompressible Fluid Flow and Heat Transfer Problems—Clear (Coupled and Linked Equations Algorithm Revised) Part II: Application Examples, *Numer. Heat Transfer B*, vol. 45, pp. 19–48, 2004.

20. D. L. Sun, Z. G. Qu, Y. L. He, and W. Q. Tao, An Efficient Segregated Algorithm for Incompressible Fluid Flow and Heat Transfer Problems—IDEAL (Inner Doubly Iterative Efficient Algorithm for Linked Equations) Part I: Mathematical Formulation and Solution Procedure, *Numer. Heat Transfer B*, vol. 53, pp. 1–17, 2008.

21. D. L. Sun, Z. G. Qu, Y. L. He, and W. Q. Tao, An Efficient Segregated Algorithm for Incompressible Fluid Flow and Heat Transfer Problems—IDEAL (Inner Doubly Iterative Efficient Algorithm for Linked Equations) Part II: Application Examples, *Numer. Heat Transfer B*, vol. 53, pp. 18–38, 2008.

22. Z. Y. Li and W. Q. Tao, A New Stability-Guaranteed Second-Order Difference Scheme, *Numer. Heat Transfer B*, vol. 42, pp. 349–365, 2002.

23. W. Q. Tao and Y. L. He, Recent Advances in Multiscale Simulations of Heat Transfer and Fluid Flow Problems, *Prog. Comput. Fluid Dynam.*, vol. 9, pp. 150–157, 2009.

24. H. Xu, H. B. Luan, and W. Q. Tao, A Reconstruction Operator for the Interface Coupling between LBM and Macro-numerical Methods of Finite-Family, *J. Xi'an Jiaotong Univ.*, vol. 43, no. 11, pp. 6–10, 2009 (in Chinese).

25. U. Frisch, D. d'Humiéres, B. Hasslacher, P. Lallemand, Y. Pomeau, and J. P. Rivet, Lattice Gas Hydrodynamics in Two and Three Dimensions, *Complex Syst.*, vol. 1, pp. 679–707, 1987.

26. D. A. Wolf-Gladrow, *Lattice Gas Cellular Automata and Lattice Boltzmann Models: An Introduction*, pp. 175–190, Springer-Verlag, Berlin, 2000.

27. W. Q. Tao, *Recent Advances in Computational Heat Transfer*, pp. 337–352, Science Press, Beijing, 2005.

28. B. F. Armaly, F. Durst, J. C. F. Pereira, and B. Schonung, Experimental and Theoretical Investigation of Backward-Facing Step Flow, *J. Fluid Mech.*, vol. 127, pp. 473–496, 1983.

29. S. Dennis and G. Chang, Numerical Solutions for Steady Flow past a Circular Cylinder at Reynolds Number up to 100, *J. Fluid Mech.*, vol. 42, pp. 471–489, 1970.

30. X. He and G. Doolen, Lattice Boltzmann Method on Curvilinear System: Flow around a Circular Cylinder, *J. Comput. Phys.*, vol. 134, pp. 306–315, 1997.

31. Z. Guo and T. Zhao, Explicit Finite-Difference Lattice Boltzmann Method for Curvilinear Coordinates, *Phys. Rev. E*, vol. 67, pp. 066709-1–066709-12, 2003.

32. T. Imamura, K. Suzuki, T. Nakamura, and M. Yoshida, Acceleration of Steady State Lattice Boltzmann Simulation on Non-uniform Mesh Using Local Time Step Method, *J. Comput. Phys.*, vol. 202, pp. 645–663, 2005.

33. U. Ghia, K. N. Ghia, and C. T. Shin, High-Re Solutions for Incompressible Flow Using the Navier-Stokes Equations and a Multigrid Method, *J. Comput. Phys.*, vol. 48, pp. 387–411, 1982.