

# Lattice Boltzmann Method on Curvilinear Coordinates System: Flow around a Circular Cylinder

Xiaoyi He<sup>\*†</sup> and Gary Doolen<sup>†</sup>

*\*Center for Nonlinear Studies, MS-B258, †Complex Systems, Group T-13, MS-B213, Theoretical Division,  
Los Alamos National Laboratory, Los Alamos, New Mexico 87545  
E-mail: xyh@cnls.lanl.gov*

Received September 18, 1996; revised March 18, 1997

Using an interpolation-based strategy, the lattice Boltzmann method is extended to apply to general curvilinear coordinate systems. As an example, a cylindrical coordinate system is used to simulate two-dimensional flow around a circular cylinder. Numerical simulations are carried out for impulsive initial conditions with Reynolds numbers up to  $10^4$ . The agreement of our results with previous computational and experimental results is satisfactory. Compared with previous lattice Boltzmann simulations of the same problem, our new approach greatly enhances the computational efficiency. © 1997 Academic Press

## I. INTRODUCTION

The lattice Boltzmann (LB) method [1, 2] has been used as a computational fluid dynamics (CFD) approach for nearly a decade. Unlike its precursor, the lattice gas automaton (LGA) [3, 4], and other molecular dynamics (MD) approaches, the LB method simulates a flow system by tracking the evolution of particle distributions instead of tracking single particles. It also differs from the traditional CFD methods in that it does not directly solve for macroscopic variables which appear in the Navier–Stokes equations. In this regard, the lattice Boltzmann method is similar to the gas dynamics approaches based on the Boltzmann equation [5], although the latter approaches usually focus on Euler equation.

The first lattice Boltzmann model [1, 2] was a direct transcription from the lattice gas automaton. The Boolean variables in the LGA were replaced by a real-number single particle distribution function. This replacement eliminates the intrinsic stochastic noise in LGA and subsequently enhances the computational efficiency. The current lattice Boltzmann models adopt the simple relaxation BGK model [6] for the collisional operator, which further simplifies the algorithm, and more importantly, eliminates the lack of Galilean invariance and the dependence of pressure on velocity [7, 8]. It has been proved that the lattice Boltzmann equation recovers the Navier–Stokes

equation using the Chapman–Enskog expansion [7, 8]. Recent numerical experiments on complex flow systems [9–12] have shown that the lattice Boltzmann method is quantitatively accurate and computationally efficient.

In addition to the above successes, current lattice Boltzmann methods can be made more computationally efficient by generalizing them to apply to irregular grids. The grid in previous LB models has been limited to triangular, square, or cubic lattices. Needless to say, this limitation greatly hampers the broad application of the LB method because an irregular grid is much more efficient for many practical problems. With an irregular mesh, curved boundaries can be described more accurately and computational resources can be used more efficiently. This issue has been addressed before and a finite volume version of the lattice Boltzmann method (FVLBE) has been proposed by Succi and coworkers [13, 14]. The FVLBE has the disadvantages of loss of locality and numerical diffusion.

In an effort to implement the LB method on an irregular mesh, we recently proposed [12] an interpolation-supplemented lattice Boltzmann equation (ISLBE) model. This motivation was based on the fact that the density distribution is a continuous function in physical space and can be well-defined for any mesh system. Density distributions at non-grid points can always be calculated by interpolation from their surrounding grid nodes. Preliminary tests of ISLBE on a nonuniform rectangular grid yield very satisfactory results [12]. The goal of this study is to generalize this new scheme to an arbitrary grid. Specifically, we formulate the ISLBE method for a general curvilinear coordinate system. As an example, the ISLBE for a cylindrical grid system is used to study the two-dimensional flow around a circular cylinder. The abundant literature on this classical flow problem allows us to carry out extensive benchmarking for this new ISLBE model.

This paper is organized as follows. Section II briefly describes the interpolation-supplemented lattice Boltzmann model, the cylindrical grid system, and the boundary

and initial conditions. Section III presents the computational results and comparisons with previous studies. Section IV discusses the results and conclusions.

## II. NUMERICAL METHOD

### A. Interpolation-Supplemented Lattice Boltzmann Method

The method used in this study is based on the well-known fact that fluid motion can be described by the Boltzmann equation [15]. In fact, the Navier–Stokes equation is just the second-order approximation of the Boltzmann equation [16, 17]. The Boltzmann equation describes the time evolution of the density distribution function. Once the distribution function is known, the macroscopic velocity and pressure can be automatically calculated from its first two moments.

The lattice Boltzmann method uses a special discrete form of the Boltzmann equation, the lattice Boltzmann equation, which can be numerically solved on digital computers. Details of the LB method can be found in many previous publications (e.g., [1, 7, 8]), although all of them are based on regular lattices. In this paper, we will use the 9-bit incompressible LBGK model [18, 19] as an example to illustrate how to implement the LB method on an irregular grid.

The evolution of the original lattice Boltzmann BGK model consists of two steps: relaxation and streaming. In the relaxation step, distributions at grid nodes relax to the equilibrium state according to the BGK rule. In the streaming process, distributions advect freely at their characteristic velocities to the next site. The evolution equation for the 9-bit incompressible LBGK model can be described by

$$p_\alpha(\mathbf{x} + \mathbf{e}_\alpha \delta_t, t + \delta_t) - p_\alpha(\mathbf{x}, t) = \frac{1}{\tau} [g_\alpha(\mathbf{x}, t) - p_\alpha(\mathbf{x}, t)], \quad (1)$$

where  $p_\alpha$  is the pressure distribution and  $g_\alpha$  is its corresponding equilibrium state. The nine discrete velocities,  $\mathbf{e}_\alpha$ 's, are defined by

$$\mathbf{e}_\alpha = \begin{cases} \mathbf{0}, & \alpha = 0, \\ (\cos[(\alpha - 1)\pi/2], \sin[(\alpha - 1)\pi/2])c, & \alpha = 1, 2, 3, 4, \\ \sqrt{2}(\cos[(\alpha - 5)\pi/2 + \pi/4], \sin[(\alpha - 5)\pi/2 + \pi/4])c, & \alpha = 5, 6, 7, 8. \end{cases} \quad (2)$$

The streaming speed,  $c$ , is defined as  $\delta_x/\delta_t$ , where  $\delta_x$  and  $\delta_t$  are the streaming length and the time step, respectively. The dimensionless relaxation time  $\tau$  is related to the fluid viscosity

$$\nu = \frac{2\tau - 1}{6} c \delta_x.$$

The equilibrium pressure distribution,  $g_\alpha$ , for the 9-bit incompressible LBE model is

$$g_\alpha = w_\alpha \left[ p + \rho \left( (\mathbf{e}_\alpha \cdot \mathbf{u}) + \frac{3}{2} \frac{(\mathbf{e}_\alpha \cdot \mathbf{u})^2}{c^2} - \frac{1}{2} \mathbf{u}^2 \right) \right], \quad (3)$$

with  $w_0 = \frac{4}{9}$ ,  $w_\alpha = \frac{1}{9}$  for  $\alpha = 1, 2, 3, 4$ , and  $w_\alpha = \frac{1}{36}$  for  $\alpha = 5, 6, 7, 8$ . The density,  $\rho$ , is constant in incompressible flows. The macroscopic pressure,  $p$ , and the velocity,  $\mathbf{u}$ , are calculated using

$$p = \sum_\alpha p_\alpha, \quad (4)$$

$$\mathbf{u} = \frac{1}{\rho c_s^2} \sum_\alpha \mathbf{e}_\alpha p_\alpha, \quad (5)$$

where  $c_s = c/\sqrt{3}$  is the sound speed. Using the Chapman–Enskog expansion, it has been proven that the above model recovers the Navier–Stokes equations [7, 8].

Note that the time-step procedure in the lattice Boltzmann scheme is similar to that in a finite difference scheme, but they are not exactly the same. The time-step in a lattice Boltzmann scheme is related to the flow evolution at the microscopic level, while the time-step in a finite difference scheme is related to flow evolutions at the macroscopic level. In a finite difference scheme, to guarantee stability, the time-step must be less than the ratio of grid size to macroscopic velocity—the so-called “Courant condition.” In the lattice Boltzmann scheme, the Courant condition is automatically satisfied at the microscopic level. Consequently, the lattice Boltzmann scheme is much more stable than traditional CFD methods.

On a regular grid in the previous LB model, starting from one grid site  $\mathbf{x}$ , the distribution  $p_\alpha$  is advected to a neighboring grid site during each time step, since the lattice separation  $\delta_x$  equals to  $c\delta_t$ . Therefore, the distribution at all grid nodes is exactly known at the next time step, and the evolution continues. On an irregular grid, however, the end point of the streaming process,  $\mathbf{x} + \mathbf{e}_\alpha \delta_t$ , is generally no longer a grid node. This implies that the distribution at grid nodes is not EXACTLY known at the next time step. The determination of the poststreaming distribution at grid nodes is essential for the implementation of the LB method on an irregular grid.

The interpolation-supplemented lattice Boltzmann method takes advantage of the fact that the poststreaming distribution as a whole is a well-determined function in physical space, because its value is known at all the *shifted* grid nodes,  $\mathbf{x} + \mathbf{e}_\alpha \delta_t$ . The unknown poststreaming distribution at the original grid nodes can always be **approximately**

calculated by interpolation using their neighboring *shifted* grid nodes. This interpolation, of course, requires having sufficient accuracy.

In principle, the concept of interpolation works for any mesh system. For simplicity, however, we choose to formulate the interpolation procedure in a curvilinear coordinate system. Assume that the flow domain can be described by a curvilinear coordinate system,  $\xi = \xi(\mathbf{x})$  and  $\eta = \eta(\mathbf{x})$ , where  $\mathbf{x}$  is a Cartesian coordinate system. The computational domain in the curvilinear coordinate system can be projected onto a uniform grid. (This can always be done by choosing  $\xi(\mathbf{x})$  and  $\eta(\mathbf{x})$  appropriately). For each  $\mathbf{e}_\alpha$ , after the streaming process, the distribution  $p_\alpha$  is known at the *shifted* grid system. The same curvilinear coordinate system can be also used for the *shifted* domain, because the shift is a constant for the entire space. In the curvilinear coordinates system of the *shifted* domain, the relative position of an original grid node,  $\mathbf{x}$ , to its shifted correspondent,  $(\xi_i, \eta_j)$  can be written as

$$\begin{aligned} d\xi_i &= \xi(\mathbf{x} - \mathbf{e}_\alpha \delta_t) - \xi_i, \\ d\eta_j &= \eta(\mathbf{x} - \mathbf{e}_\alpha \delta_t) - \eta_j. \end{aligned}$$

The poststreaming distribution at the original grid node can then be calculated using a second-order upwind interpolation,

$$p_\alpha(\mathbf{x}, t + \delta_t) = \sum_{k=0}^2 \sum_{l=0}^2 a_{i,k} b_{j,l} p_\alpha(\xi_{i+k*id}, \eta_{j+l*jd}, t + \delta_t), \quad (6)$$

where  $id = \text{sign}(1, d\xi_i)$  and  $jd = \text{sign}(1, d\eta_j)$  determine the interpolation direction. The interpolation coefficients are calculated using

$$\begin{aligned} a_{i,0} &= \frac{(|d\xi_i| - \Delta\xi)(|d\xi_i| - 2\Delta\xi)}{2\Delta\xi^2}, \\ b_{j,0} &= \frac{(|d\eta_j| - \Delta\eta)(|d\eta_j| - 2\Delta\eta)}{2\Delta\eta^2}, \\ a_{i,1} &= -\frac{|d\xi_i|(|d\xi_i| - 2\Delta\xi)}{\Delta\xi^2}, \\ b_{j,1} &= -\frac{|d\eta_j|(|d\eta_j| - 2\Delta\eta)}{\Delta\eta^2}, \\ a_{i,2} &= \frac{|d\xi_i|(|d\xi_i| - \Delta\xi)}{2\Delta\xi^2}, \\ b_{j,2} &= \frac{|d\eta_j|(|d\eta_j| - \Delta\eta)}{2\Delta\eta^2}, \end{aligned} \quad (7)$$

where  $\Delta\xi$  and  $\Delta\eta$  are the grid spacings in the curvilinear coordinate system. Knowing the distribution at the original

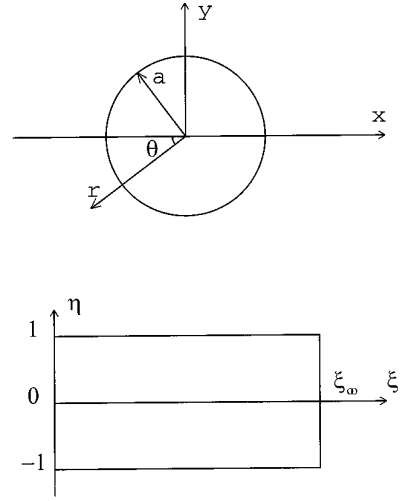


FIG. 1. The  $(x, y)$  and  $(\xi, \eta)$  planes.

grid points, the macroscopic velocity and pressure can be calculated using Eqs. (4) and (5).

As pointed out in [12], a second-order interpolation scheme is necessary to avoid spurious numerical velocity. In addition, to minimize the numerical interpolation error, it is recommended to have  $\Delta\xi$  and  $\Delta\eta$  comparable to  $d\xi$  or  $d\eta$  in regions where large velocity gradient occurs.

### B. Mesh Setup

As shown in Fig. 1, the flow region outside a circular cylinder can be transformed conformally to a semi-infinite strip  $\xi \geq 0$ ,  $-1 \leq \eta \leq 1$  using

$$\xi + i\eta = \frac{1}{\pi} \ln \left( \frac{x + iy}{a} \right), \quad (8)$$

where  $a$  is the radius of the cylinder. The variables  $\xi$  and  $\eta$  are related to the usual polar coordinates  $r$  and  $\theta$  by  $r = ae^{\pi\xi}$  and  $\theta = \pi\eta$ . In numerical simulations, the computational domain must be finite. In this study, we assume the outer boundary is located at  $\xi = \xi_\infty$  or  $r = ae^{\pi\xi_\infty}$ . The truncated computational domain in  $(\xi, \eta)$  plane,  $0 \leq \xi \leq \xi_\infty$  and  $-1 \leq \eta \leq 1$ , can be then divided into a uniform mesh,

$$\begin{aligned} \xi &= \xi_\infty \frac{i-1}{NX-1}, \quad i = 1, 2, \dots, NX, \\ \eta &= 2 \frac{j-1}{NY-1}, \quad j = 1, 2, \dots, NY. \end{aligned} \quad (9)$$

In the physical plane, this mesh is constructed from a series of concentric circles uniformly spaced in the circumferen-

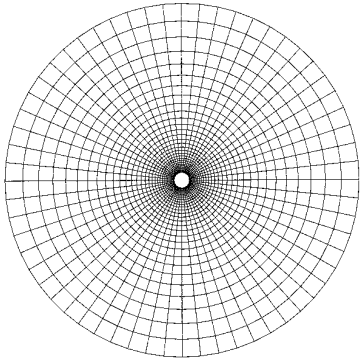


FIG. 2. Computational mesh for flow around a circular cylinder.

tial direction. A typical mesh is shown in Fig. 2 for  $NX = 31$  and  $NY = 61$  with  $\xi_\infty = 1$ . Note that no symmetry requirement is imposed in above setup. Therefore, this setup is capable of simulating symmetry-breaking phenomena, although no symmetry-breaking will physically arise in all cases simulated in this study.

This type of mesh has two obvious advantages over the regular lattice grid used in the conventional LB models. First, the boundary of the cylinder is described much more accurately. Second, the mesh is placed more densely around the cylinder and the numerical efficiency of the mesh use is greatly increased. Although commonly used in conventional CFD method (e.g., [20, 21]), this type of mesh has not been used in previous lattice Boltzmann studies.

### C. Boundary and Initial Condition

Unlike the traditional CFD methods, the lattice Boltzmann method requires a boundary condition for density distribution instead of those for velocity and pressure. In other words, the density distributions coming from outside the domain need to be given at every boundary node. Three boundary conditions are required in our simulation: one at  $\eta = \pm 1$ , one at  $\xi = 0$ , and the other at  $\xi = \xi_\infty$ . The boundary  $\eta = \pm 1$  is a virtual boundary in physical space. It corresponds to the upstream axis and can be treated as a periodic boundary in the cylindrical coordinate system.

The boundary,  $\xi = 0$ , corresponds to the cylinder wall, where the no-slip boundary condition holds. Although many schemes have been proposed for implementation of the no-slip boundary condition at a stationary wall, we feel that the simple bounce-back rule is the most convenient choice considering circular shape of the cylinder wall. With the bounce-back rule, particles colliding with a wall simply reverse their velocities. It should be born in mind that the bounce-back rule yields an effective wall half-way between the bounce-back row and its adjacent row in the fluid [22]. This will become very important in calculations of the

viscous stress on the cylinder and the length of the recirculation zone.

The boundary  $\xi = \xi_\infty$  approximates the infinite flow field where the density distribution is always at its equilibrium state. It is natural and computationally easy to assume the flow field at  $\xi = \xi_\infty$  to be a potential one with the distribution function in its equilibrium state. The far-field velocity,  $U$ , is set to 0.1, and the far-field pressure,  $p_\infty$ , is set to  $\frac{1}{3}$ .

Initially, the numerical simulation starts from an irrotational potential flow field. The same initial condition has been used in previous studies [23, 24].

## III. RESULTS

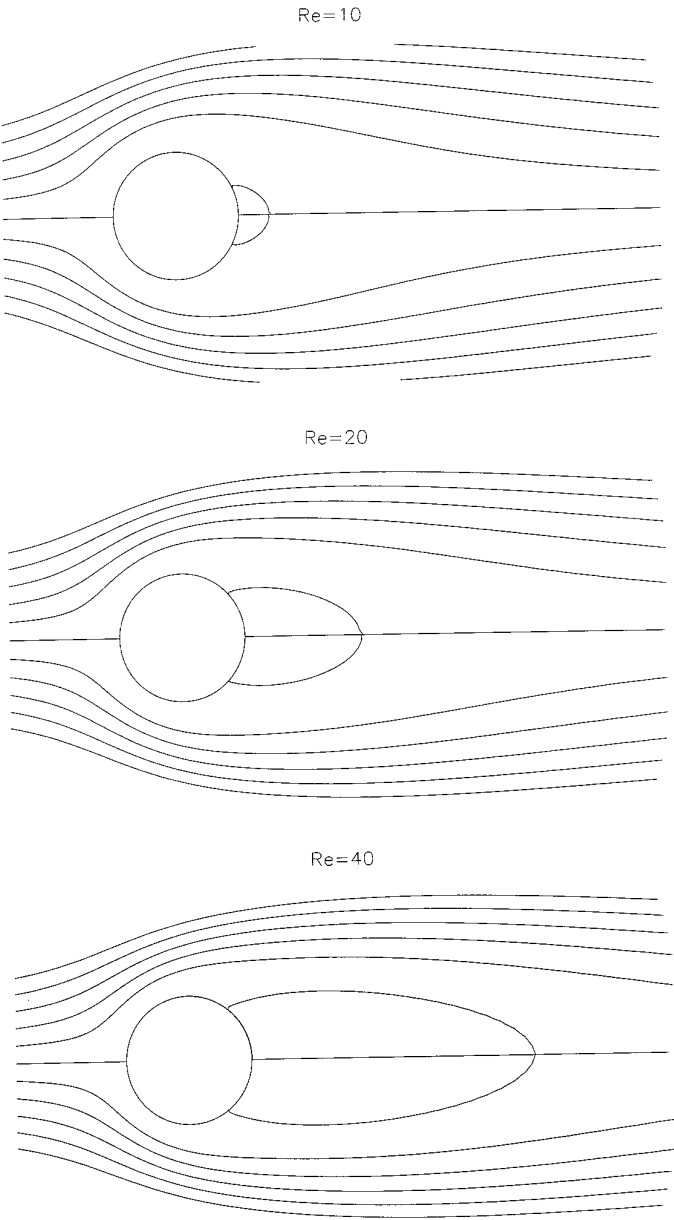
We apply the interpolation-supplemented lattice Boltzmann method to simulate flow induced by an impulsively started circular cylinder. The Reynolds number ( $Re = 2Ua/\nu$ ), based on the upstream velocity and the diameter of the cylinder, ranges from 10 to  $10^4$ . Simulations of small Reynolds number flow were carried out to evaluate the accuracy of the ISLBE method by comparing quantitatively with previous experimental and numerical studies. Simulations with moderate and higher Reynolds numbers demonstrate the ability of the ISLBE method to simulate high Reynolds number flows. Simulations of flow at a Reynolds number less than 10 is a straightforward extension; while simulations of flow at a Reynolds number higher than  $10^4$  are omitted due to the lack of experimental data.

### A. Low Reynolds Numbers

Numerical simulations were carried out for three small Reynolds numbers, 10, 20, and 40. Unless otherwise mentioned, the reported data are obtained from simulations on an  $181 \times 241$  grid for Reynolds numbers of 10 and 20 and on a  $226 \times 301$  grid for Reynolds number of 40. The outer boundary is located at  $\xi_\infty = 1.5$ , or  $r_\infty/a = 111.3$ . The ratio of the streaming length to the radius of a cylinder,  $\delta_x/a$ , is 0.025. The time step, in units of  $a/U$ , is equal to 0.0025. The CPU time for each time step is 0.2 on an IBM-RISC 6000 workstation with a peak performance of 267 Mflops.

Figure 3 shows the streamlines when flow reaches its final steady state. In all cases, a pair of stationary recirculating eddies develops behind the cylinder. The length of the recirculating region,  $L$ , from the rearmost point of the cylinder to the end of the wake, increases linearly with the Reynolds number. The quantitative geometrical parameters for the recirculating region, as well as comparisons with previous computational and experimental data, are listed in Table I. Both the wake length and separation angle,  $\theta_s$ , agree well with the results of previous studies for all three Reynolds numbers studied.

Since the lattice Boltzmann equation simulates the unsteady Navier–Stokes equations, it is interesting to check



**FIG. 3.** Streamlines at the final steady state for different Reynolds numbers.

that this model can correctly predict the time evolution of the recirculating region. Figures 4 and 5 shows the time dependences of the wake length and separation angle at Reynolds numbers of 20 and 40. The wake length and time are expressed in units of  $a$  and  $a/U$ , respectively. Experimental measurements by Coutanceau and Bouard [26] and computational simulations by Collins and Dennis [27, 28] are included for comparison. The evolution of the wake length agrees well with both the experimental and numerical results. The separation angles agree well with

**TABLE I**

Comparison of Geometrical Parameters with Previous Studies

Re	Authors	$L/a$	$\theta_s$
10	Dennis and Chang (1970)	0.53	29.6
	Nieuwstadt and Keller (1973)	0.434	27.96
	Coutanceau and Bouard (1977a)	0.68	32.5
	Fornberg (1980)	—	—
	Present	0.474	26.89
20	Dennis and Chang (1970)	1.88	43.7
	Nieuwstadt and Keller (1973)	1.786	43.37
	Coutanceau and Bouard (1977a)	1.86	44.8
	Fornberg (1980)	1.82	—
	Present	1.842	42.96
40	Dennis and Chang (1970)	4.69	53.8
	Nieuwstadt and Keller (1973)	4.357	53.34
	Coutanceau and Bouard (1977a)	4.26	53.5
	Fornberg (1980)	4.48	—
	Present	4.490	52.84

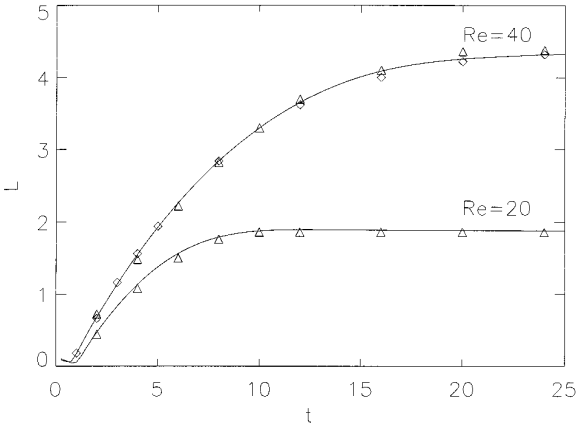
Note.  $L$ : wake length.  $\theta_s$ : separation angle.

the numerical simulation, but they evolve faster than the experimental measurements.

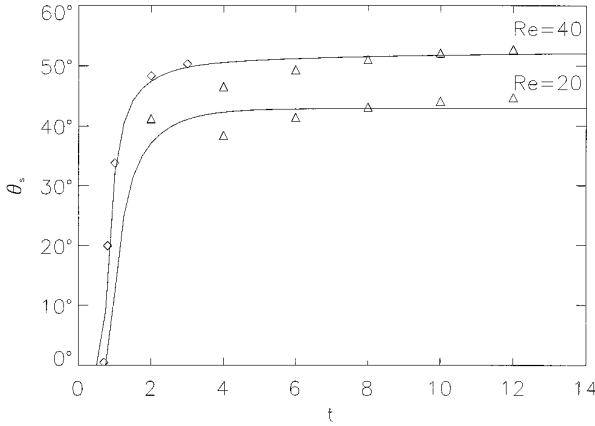
Another important test for the ISLBE is its ability to calculate the drag coefficient and the pressure distribution on the cylinder. Figure 6 shows the pressure distribution on the cylinder. The pressure is expressed in its nondimensional form,

$$C_p = (p - p_\infty)/\frac{1}{2}\rho U^2. \quad (10)$$

The highest pressure is observed at the front stagnation point ( $\theta = 0^\circ$ ). The location of the lowest pressure does not occur at the rearmost point of the cylinder. It moves



**FIG. 4.** Time variations of the wake lengths for low Reynolds numbers:  $\Delta$ , experimental data by Coutanceau and Bouard [26] with the diameter ratio of 0;  $\diamond$ , numerical results by Collins and Dennis [27].



**FIG. 5.** Time development of the separation angle:  $\Delta$ , experimental data by Coutanceau and Bouard [26] with the diameter ratio of 0;  $\diamond$ , numerical results by Collins and Dennis [27].

upstream as the Reynolds number increases. This phenomenon is consistent with previous observations (e.g., [20, 29, 21]).

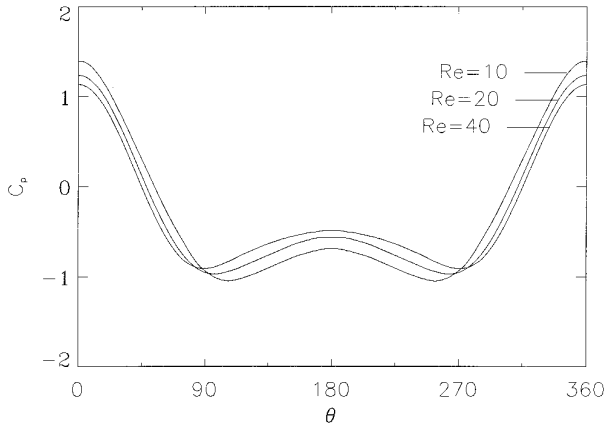
Table II lists the quantitative comparisons for the stagnation pressures at the front and end of the cylinder, as well as the drag coefficient. The drag coefficient,  $C_D$ , is calculated using

$$C_D = \frac{1}{\rho U^2 a} \int S \cdot \mathbf{n} dl,$$

where  $\mathbf{n}$  is the normal direction of the cylinder wall and

$$S = -pI + \rho\nu(\nabla\mathbf{u} + \mathbf{u}\nabla)$$

is the stress tensor. In the above equation, the calculation of the pressure is straight forward, but the evaluation of



**FIG. 6.** Pressure distribution on the cylinder wall.

**TABLE II**

Comparison of Dynamical Parameters with Previous Studies

Re	Authors	$C_D$	$C_p(\text{gp})$	$-C_p(0)$
10	Dennis and Chang (1970)	2.846	1.489	0.742
	Nieuwstadt and Keller (1973)	2.828	1.500	0.692
	Fornberg (1980)	—	—	—
	Present	3.170	1.393	0.687
20	Dennis and Chang (1970)	2.045	1.269	0.589
	Nieuwstadt and Keller (1973)	2.053	1.274	0.582
	Fornberg (1980)	2.000	1.28	0.54
	Present	2.152	1.233	0.567
40	Dennis and Chang (1970)	1.522	1.144	0.509
	Nieuwstadt and Keller (1973)	1.550	1.117	0.554
	Fornberg (1980)	1.498	1.14	0.46
	Present	1.499	1.133	0.487

Note.  $C_D$ : drag coefficient.  $C_p(\pi)$ : pressure coefficient at the front stagnation point.  $C_p(0)$ : pressure coefficient at the rear stagnation point.

the shear rate, or the derivative of the velocity, requires special consideration. To calculate the derivative of the velocity using a finite-difference equation, we need to know the exact location of the wall. As point out in [22], the bounce-back rule yields an effective wall half-way between the bounce-back row and its adjacent row in fluid. In our calculations, placing the wall at the bounce-back row indeed yields a significant error. Placing the wall half-way between the bounce-back row and its adjacent row gives satisfactory results. As shown in Table II, all the dynamical parameters compare well with results of the previous studies.

### B. Moderate and High Reynolds Numbers

To demonstrate the ability of the ISLBE method to simulate high Reynolds number flows, we also extended our study into flow regimes with moderate and high Reynolds numbers. Three Reynolds numbers, 550, 3000, 9500 were chosen to compare with the existing experimental observations and numerical simulations. Flow around a cylinder at these Reynolds numbers eventually becomes three-dimensional and turbulent, and we do not intend to cover that regime. We only want to focus on the early stage of the flow development in which the two-dimensional laminar assumption has been justified by experiments.

All three simulations were carried out on a computational grid with a size of  $241 \times 241$ . For Reynolds numbers of 550 and 3000, the outer boundary is placed at  $\xi_\infty = 1.0$  or  $r_\infty/a = 23.1$ , the ratio of the streaming length to the radius of cylinder,  $\delta_\infty/a$ , is  $\frac{1}{80}$ , and the time step, in units of  $a/U$ , is equal to  $\frac{1}{800}$ . For the Reynolds number of 9500, the corresponding parameters are  $\xi_\infty = 0.5$  ( $r_\infty/a = 4.8$ ),

$\delta_x/a = \frac{1}{160}$ , and  $a\delta_t/U = \frac{1}{1600}$ . The CPU time for each time step is 0.26 on an IBM-RISCH 6000 workstation.

The flow patterns at different times for  $Re = 550$  are shown in Fig. 7. The time is expressed in units of  $a/U$ . A pair of recirculating eddies have been formed by  $t = 1.0$ . A secondary vortex appears at  $t = 2.75$  and becomes visible at  $t = 3$ . This appearance time of the secondary vortex compares quite well with the prediction of Ta Phuoc Loc ( $t = 2.85$ ) [31]. The secondary vortex develops with time and becomes more obvious at  $t = 5$ . This secondary vortex is always isolated in the main vortex. The flow structures compare well with the flow visualization of Bouard and Coutanceau [30] and the computational simulations of Ta Phuoc Loc [31] and Koumoutsakos and Leonard [24]. The secondary vortex in our simulation is better revealed than that in Chang and Chern [23].

Figure 8 shows the flow pattern at different times for  $Re = 3000$ . Comparing with flow at  $Re = 550$ , the secondary vortex appears at an earlier time ( $t = 2.0$ ). The secondary vortex grows so large at later time ( $t = 3$  and  $t = 5$ ) that it bulges out into the main stream. As a result, a part of the main vortex is pinched off and joins the original

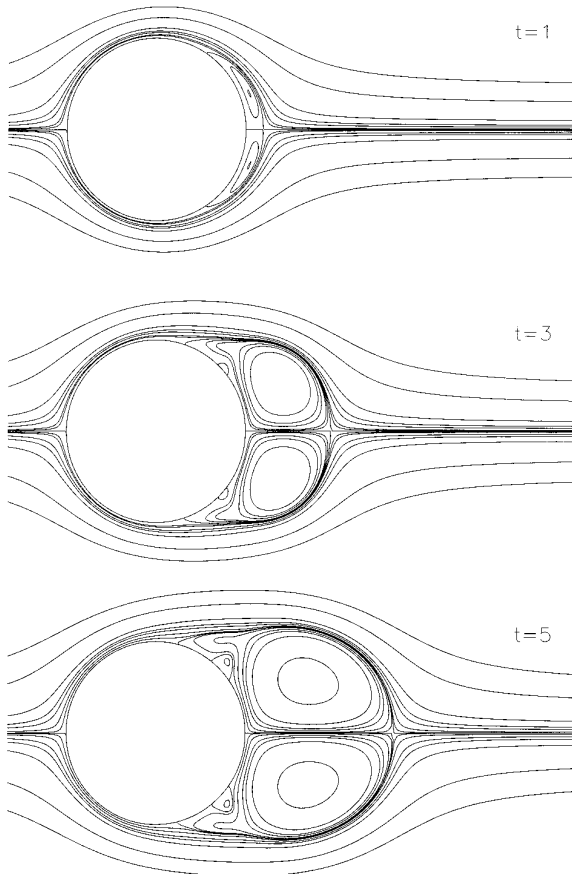


FIG. 7. Evolution with time of streamlines for  $Re = 550$ .

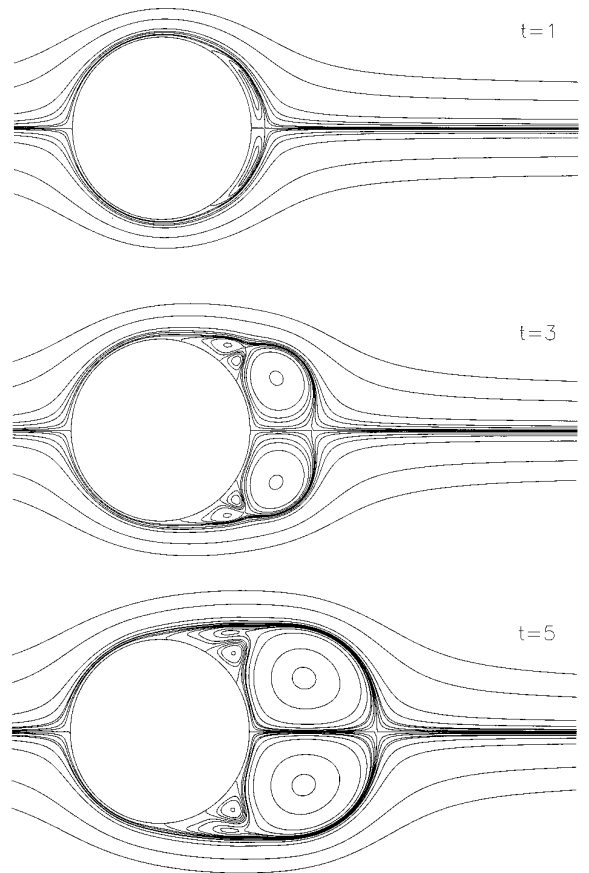


FIG. 8. Evolution with time of streamlines for  $Re = 3000$ .

secondary vortex to form a pair of counterrotating secondary vortices. This so-called  $\alpha$ -phenomenon has been observed experimentally by Bouard and Coutanceau [30] and numerically by Ta Phuoc Loc and Bouard [32]. The detailed flow structure in our simulation matches very well with their observations.

Figure 9 shows the flow pattern at different times for  $Re = 9500$ . Flow at this Reynolds number has the feature of the so-called  $\beta$ -phenomenon [30], which our simulation captures quite well. A thin recirculating wake—“forewake” after Bouard and Coutanceau [30]—is formed at early stage ( $t = 1.2$ ). The core of this “forewake” gradually develops into a rapidly rotating vortex which separates the initial “forewake” into two parts at a later time ( $t = 2$ ). The rapidly rotating vortex eventually absorbs the “forewake” to form the main wake ( $t = 2.8$  and  $t = 4.0$ ). All the flow structures in our simulation compare well with the previous flow visualization and numerical simulations [30, 32, 33, 24].

A quantitative comparison of the time variation of the wake length with the previous experimental measurements [30] is shown in Fig. 10 and good agreement is obtained.

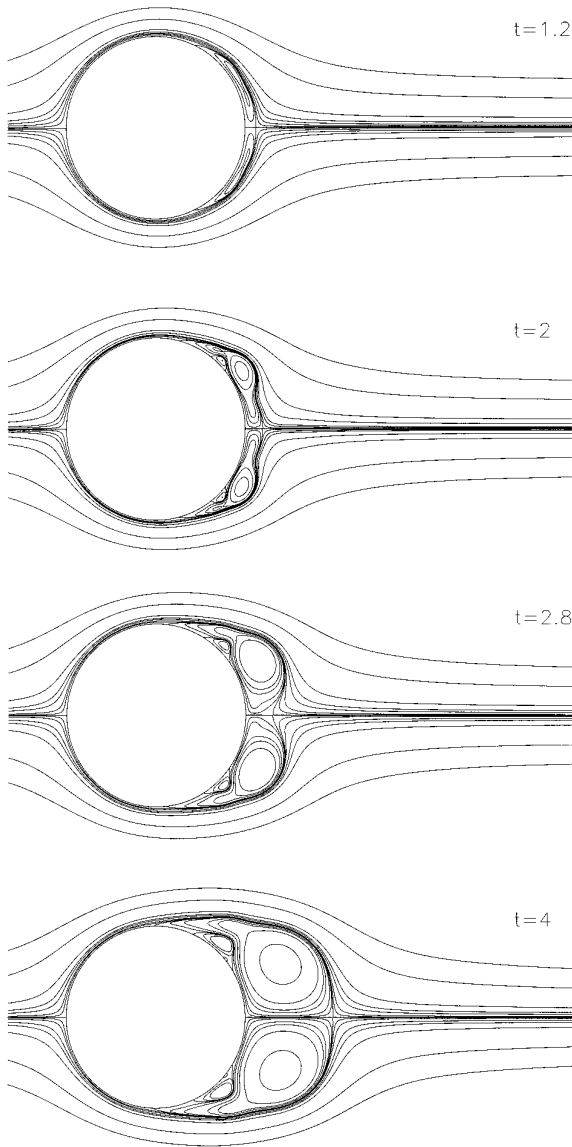


FIG. 9. Evolution with time of streamlines for  $Re = 9500$ .

The wake length develops in a linear fashion for  $Re = 550$ . For higher Reynolds numbers, the wake length increases slowly in the very beginning ( $t < 3$ ), corresponding to the development of the “forewake.” The faster increase of  $L$  at later times ( $t > 3$ ) is associated with the destruction of the “forewake” and the formation of the main wake.

### C. Effects of the Grid Size and Outer Boundary Location

To demonstrate that the results are grid-independent, we carried out numerical simulations for several grids with different sizes. The length of the recirculating zone at steady state was used as an indicator for the effect of the

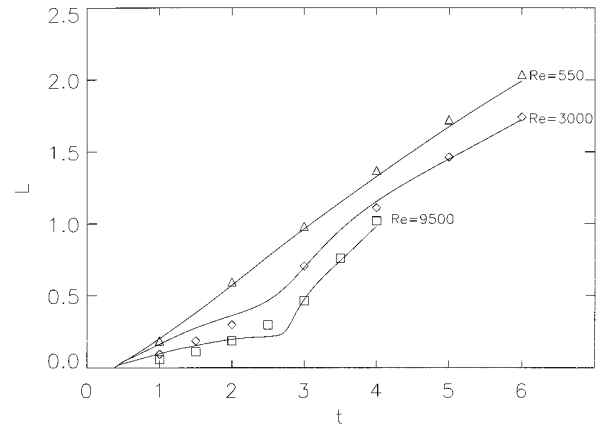


FIG. 10. Time variations of the wake lengths for moderate and high Reynolds numbers. Data points are the experimental measurements by Bouard and Coutanceau [30].

grid size. As shown in Table III for the three small Reynolds numbers studied,  $L/a$  converges when the grid spacing decreases. A grid with size of  $181 \times 241$  is fine enough to obtain grid-independent results for Reynolds numbers of 10 and 20, while a  $226 \times 301$  grid is necessary for Reynolds number of 40.

We studied the effect of the outer boundary location using  $Re = 40$  and  $Re = 9500$  as examples. The geometrical and dynamical parameters at the final steady state are compared in Table IV for flow at  $Re = 40$  with different outer boundaries. As shown, moving the outer boundary from  $23.1a$  to  $111.3a$  decreases the drag coefficient by 6%, increases the wake length by 5%, and has little effect on the separation angle. Further moving outer boundary away barely has any effect on results.

The effect of the outer boundary location seems to be negligible at early stage of flow development. Figure 11 compares the streamlines of flow at  $Re = 9500$  at  $t = 2$  for  $r_\infty = 4.8$  and  $r_\infty = 10.6$ . The flow structures are almost the same. Measurements show that  $L$  only changes 0.3% when  $r_\infty$  increases from 4.8 to 10.6.

TABLE III  
Effect of the Grid Size

Mesh	$Re = 10$	$Re = 20$	$Re = 40$
$46 \times 61$	0.351	1.448	2.489
$91 \times 121$	0.470	1.810	3.872
$136 \times 181$	0.474	1.842	4.306
$181 \times 241$	0.474	1.843	4.442
$226 \times 301$	—	1.842	4.490
$271 \times 361$	—	—	4.510



TABLE IV

Effect of the Outer Boundary Location for  $Re = 40$ 

$r_\infty/a$	$\xi_\infty$	$C_f$	$L/a$	$\theta_s$
	1.0	1.598	4.292	52.22
23.1				
111.3	1.5	1.499	4.490	52.84
535.5	2.0	1.494	4.516	52.86

## IV. CONCLUSIONS

The lattice Boltzmann method has been successfully extended to general curvilinear coordinate systems by adding an interpolation step. Numerical results for flow around a circular cylinder using a cylindric coordinate system are found to be satisfactory. The results agree well with previous computational and experimental data. Compared with previous LBE studies on the same problem [33], the required number of grid points is reduced at least by a factor of a few hundred. The computational efficiency is greatly enhanced. This enhancement will be even greater for simulations of three-dimensional flows.

The only difference between the interpolation-supplemented lattice Boltzmann equation (ISLBE) and previous LBE model is the addition of a new interpolation step between the streaming and the relaxation steps. The beauty

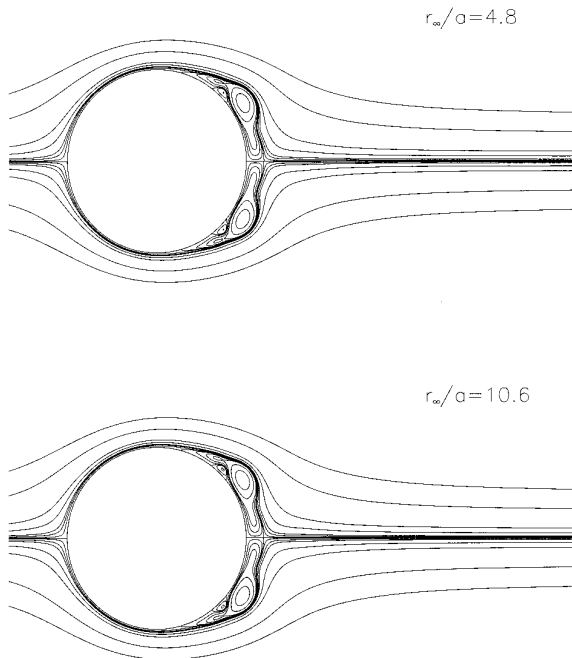


FIG. 11. Effect of the location of the outer boundary on flow pattern.  $Re = 9500$  and  $t = 2$ .

of the ISLBE is that it does not change the locality of the LB method. Therefore, it is still able to exploit fully the power of parallel computing. Because the interpolation coefficients are flow field independent and can be calculated initially, the interpolation step does not increase the computational time significantly. In the current code, we only measured a factor of two increase in CPU time compared with LBE simulations without interpolation on the same size of grids.

The cost of the interpolation is the introduction of a small numerical viscosity, which is also observed in the finite volume lattice Boltzmann method [14]. A previous study [12] has shown that the linear interpolation schemes do introduce a numerical viscosity. Using a second-order quadratic interpolation scheme, however, can reduce the viscosity to the same level as the truncation error of the lattice Boltzmann equation itself. A theoretical analysis of the numerical viscosity is desirable in the future.

## ACKNOWLEDGMENTS

The authors thank Dr. Shuling Hou and Dr. Lishi Luo for their helpful discussions during this work.

## REFERENCES

1. G. McNamara and G. Zanetti, Use of the Boltzmann equation to simulate lattice-gas automata, *Phys. Rev. Lett.* **61**, 2332 (1988).
2. F. J. Higuera and J. Jeménez, Boltzmann approach to lattice gas simulations, *Europhys. Lett.* **9**, 663 (1989).
3. U. Frisch, B. Hasslacher, and Y. Pomeau, Lattice-gas automata for the Navier–Stokes equation, *Phys. Rev. Lett.* **56**, 1505 (1986).
4. S. Wolfram, Cellular automaton fluid 1: Basic theory, *J. Stat. Phys.* **45**, 471 (1986).
5. K. Xu *et al.*, On the construction of BGK-type schemes for compressible flow simulations, AIAA Paper 96-0525, 1996.
6. P. L. Bhatnagar, E. P. Gross, and M. Krook, A model for collision processes in gases. I. Small amplitude processes in charged and neutral one-component system, *Phys. Rev.* **94**, 511 (1954).
7. H. Chen, S. Chen, and W. H. Matthaeus, Recovery of the Navier–Stokes equations using a lattice Boltzmann method, *Phys. Rev. A* **45**, R5339 (1991).
8. Y. H. Qian, D. d’Humières, and P. Lallemand, Lattice BGK models for the Navier–Stokes equation, *Europhys. Lett.* **17**, 479 (1992).
9. A. J. C. Ladd, Numerical simulations of particulate suspensions via a discretized Boltzmann equation. Part I. Theoretical foundation, *J. Fluid Mech.* **271**, 285 (1994).
10. A. J. C. Ladd, Numerical simulations of particulate suspensions via a discretized Boltzmann equation. Part II. Numerical results, *J. Fluid Mech.* **271**, 311 (1994).
11. S. Hou, Q. Zou, S. Chen, G. D. Doolen, and A. Cogley, Simulation of cavity flow by the lattice Boltzmann method, *J. Comput. Phys.* **118**, 329 (1995).
12. X. He, L. S. Luo, and M. Dembo, Some progress in lattice Boltzmann method: Part I. Nonuniform mesh grids, *J. Comput. Phys.* **129**, 357 (1996).
13. F. Nannelli and S. Succi, The lattice Boltzmann equation on irregular lattices, *J. Stat. Phys.* **68**, 401 (1992).

14. S. Succi, G. Amati, and R. Benzi, Challenges in lattice Boltzmann computing, *J. Stat. Phys.* **81**, 5 (1995).
15. S. Chapman and T. G. Cowling, *The Mathematical Theory of Non-Uniform Gases* (Cambridge Univ. Press, Cambridge, 1970).
16. H. Grad, On the kinetic theory of rarefied gases, *Commun. Pure Appl. Math.* **2**, 331 (1949).
17. C. L., Pekeris, Solution of the Boltzmann–Hilbert integral equation, *Proc. Nat. Acad. Soc.* **41**, 661 (1955).
18. Q. Zou, S. Hou, S. Chen, and G. Doolen, An improved incompressible lattice Boltzmann model for time-independent flows, *J. Stat. Phys.* **81**, 35 (1995).
19. X. He and L. S. Luo, Lattice Boltzmann model for the incompressible Navier–Stokes equation, *J. Stat. Phys.* [in press].
20. S. C. R. Dennis and G. Z. Chang, Numerical solutions for steady flow past a circular cylinder at Reynolds number up to 100, *J. Fluid Mech.* **42**, 471 (1980).
21. B. Fornberg, A numerical study of steady viscous flow past a circular cylinder, *J. Fluid Mech.* **98**, 819 (1980).
22. I. Ginzbourg and P. M. Alder, Boundary condition analysis for the three-dimensional lattice Boltzmann model, *J. Phys. II France* **4**, 191 (1994).
23. C. C. Chang and R. L. Chern, A numerical study of flow around an impulsively started circular cylinder by a deterministic vortex method, *J. Fluid Mech.* **233**, 243 (1991).
24. P. Koumoutsakos and A. Leonard, High-resolution simulations of the flow around an impulsively started cylinder using vortex method, *J. Fluid Mech.* **296**, 1 (1995).
25. M. Coutanceau and R. Bouard, Experimental determination of the main features of the viscous flow in the wake of a circular cylinder in uniform translation. Part 1. Steady flow, *J. Fluid Mech.* **79**, 231 (1977a).
26. M. Coutanceau and R. Bouard, Experimental determination of the main features of the viscous flow in the wake of a circular cylinder in uniform translation. Part 2. Unsteady flow, *J. Fluid Mech.* **79**, 257 (1977b).
27. W. M. Collins and S. C. R. Dennis, Flow past an impulsively started circular cylinder, *J. Fluid Mech.* **60**, 105 (1973).
28. W. M. Collins and S. C. R. Dennis, The initial flow past an impulsively started circular cylinder, *Q. J. Mech. Appl. Math.* **26**, 53 (1973).
29. F. Nieuwstadt and H. B. Keller, Viscous flow past circular cylinders, *Comput. & Fluids* **1**, 59 (1973).
30. R. Bouard and M. Coutanceau, The early stage of development of the wake behind an impulsively started cylinder for  $40 < Re < 10^4$ , *J. Fluid Mech.* **101**, 583 (1980).
31. Ta Phuoc Loc, Numerical analysis of unsteady secondary vortices generated by an impulsively started circular cylinder, *J. Fluid Mech.* **100**, 111 (1980).
32. T. P. Loc and R. Bouard, Numerical solution of the early stage of the unsteady viscous flow around a cylinder: A comparison with experimental visualization and measurement, *J. Fluid Mech.* **160**, 93 (1985).
33. F. J. Higuera and S. Succi, Simulating the flow around a circular cylinder with a lattice Boltzmann equation, *Europhys. Lett.* **8**, 517 (1989).