# Simulation of fluid flow in a body-fitted grid system using the lattice Boltzmann method

Masoud Mirzaei* and Amin Poozesh†

*Center of Excellence for Design and Simulation of Space Systems, Faculty of Aerospace Engineering, K.N. Toosi University of Technology, Tehran, Iran*

(Received 16 September 2012; revised manuscript received 15 March 2013; published 28 June 2013)

The ability of the interpolation supplemented lattice Boltzmann method (ISLBM) diminishes in simulation of fluid flow around complex geometries and it is nearly impossible to use this method in body-filled grid systems. In this paper, a developed version of the interpolation supplemented lattice Boltzmann method is proposed to remove the limitations of the original ISLBM. Combination of the ISLBM and the Joukowski transformation is the basis of the method. In fact, using the Joukowski transformation, the physical domain with a body-fitted grid system is mapped to the computational domain with a uniform Cartesian grid system such that the conventional ISLBM can be easily applied. The results are compared with those of a Navier-Stokes solver and there is good agreement between these two results.

DOI: 10.1103/PhysRevE.87.063312 PACS number(s): 47.11.Qr, 47.11.St

## I. INTRODUCTION

There has been rapid progress in developing and employing the lattice Boltzmann method (LBM) as an alternative for the computational fluid dynamics (CFD) approach for simulation of fluid flows [1–3]. However, because of the essential restriction of the standard LBM in using grid systems other than the Cartesian uniform grid system, the broad application of the LBM in engineering problems has been greatly hampered. It is clear that an irregular grid system is always preferable because curved boundaries can be described in a simple way and computational resources can be used more efficiently. The lattice-uniformity requirement of the standard LBM comes from its ancestor, the lattice-gas automata (LGA) method [4]. In the LGA, all the particles with the same mass must move to their neighboring points at one time step. This requirement leads to the lattice uniformity. The LBM is developed from the LGA, in which the mass of particles is replaced by the density distribution function. In general, the collision operator is linearized by the Bhatnagar-Gross-Krook (BGK) approximation in the LBM [5]. The LBM enhances the computational efficiency of the LGA method, but inherits its feature of the lattice uniformity. Theoretically, this feature is not necessarily to be kept for the LBM because the distribution functions are continuous in physical space. Currently, there are three ways to improve the standard LBM so that it can be applied to complex problems. The first is the interpolation supplemented LBM (ISLBM) proposed by He and co-workers [6–8]. In this method, in order to obtain the density distribution function at a lattice point, interpolation is applied at every time step. So the computational effort by this method is large in comparison with the standard LBM. The second is based on the solution of a partial differential equation. For simplicity, this scheme is called the differential LBM [9]. For complex problems, the differential LBM can be solved by conventional finite difference schemes with the aid of a coordinate transformation, by finite volume method, or by finite element method [10–12].

The third one is the grid refinement technique that was first presented by Fillipova and Hänelin [13] and improved by Yu *et al.* [14]. The grid refinement techniques were extended for multiple-relaxation-time (MRT) and cascaded lattice Boltzmann (CLB) models by Tölke and Krafczyk [15] and also by Schönher *et al.* [16]. This technique works based on the conventional adaptive mesh refinement method (i.e., using a coarse grid in the whole domain and a finer grid in critical regions).

In this study, in order to simulated fluid flows in a nonuniform body-fitted grid system using the standard LBM, an alternative method is developed. In fact, we have used mapping from a physical domain onto a computational domain and an intermediate grid system is used to correlate between the physical and computational grid systems. Using this method makes it possible to simulate fluid flow around complex geometries in a body-fitted grid system.

## II. NUMERICAL MODEL

### A. Lattice Boltzmann method

The LBM is a simulation technique in which the Boltzmann equation is solved for the particle distribution function on a regular, uniform Cartesian grid. The Boltzmann equation used to model particles motion is

$$\frac{\partial f_\alpha}{\partial t} + \vec{\vartheta}_\alpha \cdot \vec{\nabla} f_\alpha = -\frac{1}{\lambda}\left(f_\alpha - f_\alpha^{eq}\right). \tag{1}$$

The single relaxation time form of the BGK approximation [5] is used for a collision operator that converts the Boltzmann equation into a linear differential equation. In Eq. (1), $f$ is the distribution function, $f_\alpha^{eq}$ is the equilibrium distribution function (Maxwell-Boltzmann distribution), $\vec{\vartheta}$ is the particle velocity vector, and $\lambda$ is the relaxation time.

In order to confine the motion of the particles in specified directions, various models are proposed. The most popular of them for two-dimensional flow fields is the D2Q9 model. Figure 1 describes this model. It can be seen there are nine directions $\vec{e}_\alpha$ and every particle is allowed to move along only these directions. The equilibrium distribution function for the

*Mirzaei@kntu.ac.ir
†A.Poozesh@sina.kntu.ac.ir

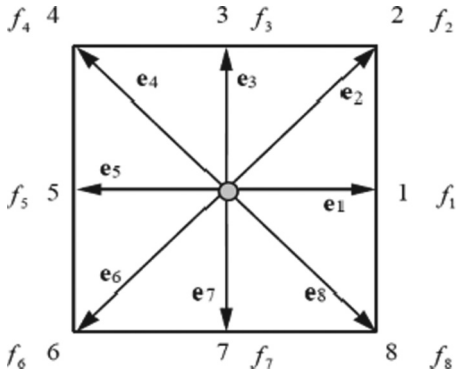



FIG. 1. The D2Q9 model.

D2Q9 model can be expressed as

$$f_\alpha^{eq} = \rho \cdot w_\alpha \left[1 + \frac{3}{c^2}\vec{e}_\alpha \cdot \vec{u} + \frac{9}{2c^4}(\vec{e}_\alpha \cdot \vec{u})^2 - \frac{3}{2c^2}\vec{u} \cdot \vec{u}\right], \tag{2}$$

where $w_\alpha$ is the weighting factor for discretization of the equilibrium distribution function. Equation (1) is discretized in time and space domains. Using the BGK model for the collision term at time step $\delta t$ and space step $\delta x = \vec{e}_\alpha \delta t$, the equation is discretized as

$$f_\alpha(\vec{x}_i + \vec{e}_\alpha \delta t, t + \delta t) - f_\alpha(\vec{x}_i, t) = -\frac{1}{\tau}\left[f_\alpha(\vec{x}_i, t) - f_\alpha^{eq}(\vec{x}_i, t)\right], \tag{3}$$

where $\tau = \frac{\lambda}{\delta t}$ is the nondimensional relaxation time and indicates the time it takes a particle to reach equilibrium after collision with another particle. The equation is solved in two steps: (i) a collision step

$$\tilde{f}_\alpha(\vec{x}_i, t + \delta t) - f_\alpha(\vec{x}_i, t) = -\frac{1}{\tau}\left[f_\alpha(\vec{x}_i, t) - f_\alpha^{eq}(\vec{x}_i, t)\right] \tag{4}$$

and (ii) a streaming step

$$f_\alpha(\vec{x}_i + \vec{e}_\alpha \delta t, t + \delta t) = \tilde{f}_\alpha(\vec{x}_i, t + \delta t), \tag{5}$$

where $\tilde{f}_\alpha$ represents distribution after collision. After solving the equation for $f_\alpha$, the macroscopic properties of the flow field can be calculated as

$$\rho = \sum_{\alpha=0}^{8} f_\alpha = \sum_{\alpha=0}^{8} f_\alpha^{eq}, \tag{6}$$

$$\rho \cdot \vec{u} = \sum_{\alpha=1}^{8} \vec{e}_\alpha f_\alpha = \sum_{\alpha=1}^{8} \vec{e}_\alpha f_\alpha^{eq}. \tag{7}$$

### B. Joukowski transformation

One of the most common usage conformal mappings in aerodynamics is the Joukowski mapping. Using this mapping, it is possible to transform geometries such as a circular cylinder to a straight-line segment and a symmetric or a nonsymmetric airfoil in another complex plane [17]. For instance, let us consider a circular cylinder in the $w$ plane as shown in Fig. 2. The cylinder can be expressed as $w = ae^{i\theta}$ ($a$ is the radius of the cylinder). The Joukowski transformation is a correlation that transforms the cylinder in general to an asymmetric airfoil.

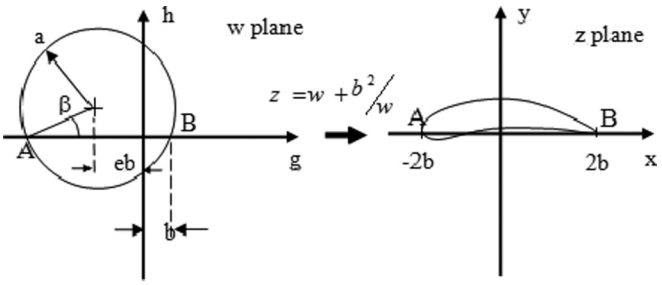



FIG. 2. Joukowski transformation.

This transformation is expressed as

$$z = w + b^2/w, \tag{8}$$

where $b$ is the parameter by which the chord of the airfoil can be controlled in the transformed plane. By changing the position of the center of the cylinder in the $w$ plane, other parameters of the airfoil such as mean camber and thickness can be controlled. In fact, the position $g$ of the center determines the thickness distribution and position $h$ of the center controls the camber distribution along the chord of the airfoil in the transformed plane.

## III. GRID GENERATION

The Joukowski transformation is used to generate a body-fitted grid system around a desired geometry. First, a nonuniform grid around a circular cylinder is generated using a mapping that transfers the uniform Cartesian grid system to a nonuniform polar grid system. The uniform grid is shown in Fig. 3, where the Cartesian domain is limited within $[0,\xi_\infty]$ along the $\xi$ direction and within $[-1, 1]$ along the $\eta$ direction. In this domain a Cartesian uniform grid with dimension N $\times$ M is generated and the coordinates of each grid are

$$\xi = \xi_\infty \frac{i-1}{N-1}, \quad i = 1,2,\ldots,N,$$
$$\eta = -1 + 2\frac{j-1}{M-1}, \quad j = 1,2,\ldots,M. \tag{9}$$

With determination of the coordinates of grid points on the uniform Cartesian grid points, their corresponding points in the polar grid system can be expressed as

$$r = ae^{\pi\xi}, \quad \theta = \pi\eta, \tag{10}$$

where $a$ is radius of the cylinder.

This nonuniform grid around the cylinder consists of concentric circles ($r$ constant) and rays ($\theta$ constant). As stated before, the aim of this research is to resolve the fluid flow around an airfoil in a body-fitted lattice system

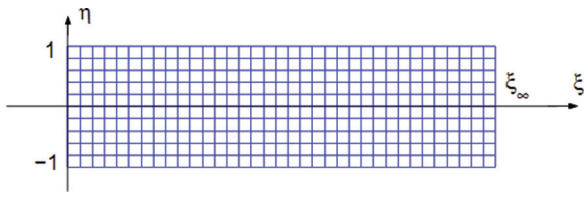



FIG. 3. (Color online) Uniform rectangular grid.

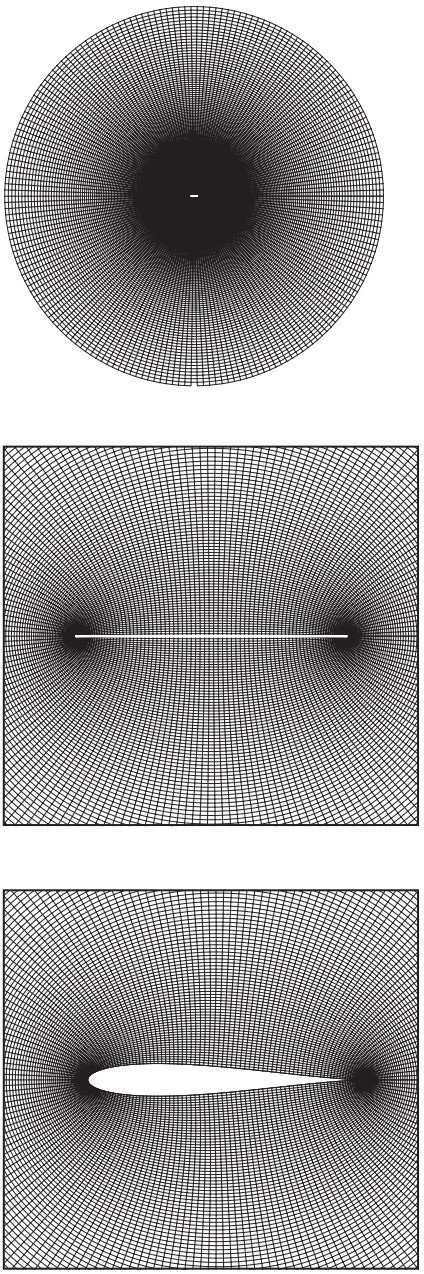


FIG. 4. Body-fitted grid generated around a flat plate and symmetric airfoil by using the Joukowski transformation.

using a developed version of the ISLBM (DILBM). A body-fitted lattice system around the airfoil is generated using the Joukowski transformation (8). In this transformation the nonuniform grid around a circular cylinder is mapped onto an O-type nonuniform body-fitted grid around a flat plate or an airfoil. Figure 4 shows some samples of the generated grids around a flat plate and symmetric airfoil. The reported data are obtained from simulations on a 221 × 251 grid.

## IV. DEVELOPED INTERPOLATION LATTICE BOLTZMANN METHOD

The standard LBM cannot be directly implemented on a nonuniform body-fitted grid. In order to use the LBM for such a grid system, collision and streaming steps have to be

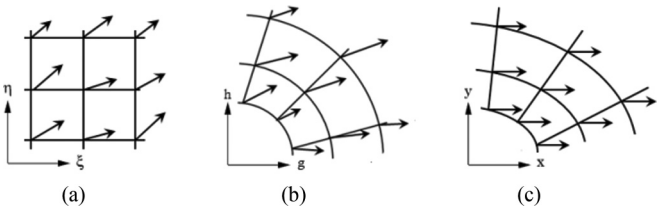



FIG. 5. Position of the grid points in the three domains after the streaming step: (a) the computational domain, (b) the intermediate domain, and (c) the physical domain.

modeled. Since the collision equation needs only the local information, it can be directly implemented in the physical domain on all points of the body-fitted grid.

After the collision step, the streaming step starts. At this step, the distribution function for each $e_\alpha$ at its new position $x - e_\alpha \delta t$ is known. In order to determine the new coordinates of the points in the computational domain, an intermediate domain is required. In fact, this domain communicates the positions of the particles in the physical domain and their corresponding positions in the computational domain after streaming step.

Figure 5 depicts computational, intermediate, and physical domains. Using new positions of the points in the physical domain after the streaming step, their corresponding positions in the intermediate domain are determined using an inverse of the Joukowski transformation

$$w = \frac{z}{2} \pm \sqrt{\frac{z^2}{4} - c^2}, \tag{11}$$

where $z = x + iy$. By determining the new positions of the points in the intermediate domain, their corresponding points in the computational domain can be determined using $\eta = \eta(w)$ and $\xi = \xi(w)$:

$$\xi + i\eta = \frac{1}{\pi} \ln\left(\frac{g + ih}{a}\right), \tag{12}$$

where $w = g + ih$. Using this procedure, we can correlate the body-fitted grid system and the computational grid system (uniform grid).

The distribution functions at the grid points are obtained using interpolation. The relative position of a grid point to its

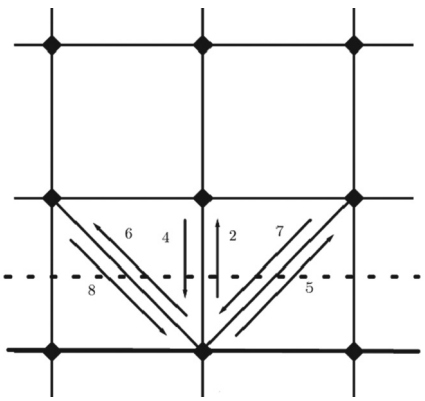



FIG. 6. Halfway bounce back boundary condition.

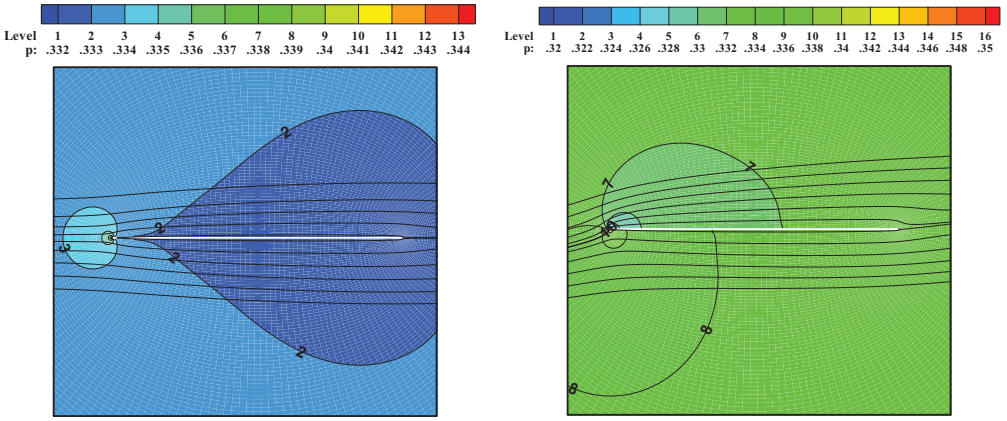



FIG. 7. (Color online) Pressure contours and streamlines around a flat plate for two different angles of attack: (a) 0° and (b) 10°.

poststreaming position can be written as

$$d\xi_i = \xi(\mathrm{w} - e_\alpha \delta_t) - \xi_i, \quad d\eta_j = \eta(\mathrm{w} - e_\alpha \delta_t) - \eta_i. \tag{13}$$

These relations depict the weighting factor of the interpolation of the distribution function.

In order to control the interpolation error in an acceptable range, the accuracy of the interpolation method and also appropriate discretization of the physical domain are of great importance in the presented method. Since the LBM has second-order accuracy, the interpolations have to be second order as well [18]. Using a linear interpolation scheme will inevitably generate numerical diffusivity and viscosity that make the DILBM unable to simulate the viscous fluid flow correctly.

Therefore, a second-order upwind approximation is implemented to calculate the distribution after the streaming step:

$$f_\alpha(x, t+\delta t) = \sum_{k=0}^{2}\sum_{l=0}^{2} a_{i,k} b_{j,k}\, p(\xi_{i+k^* id}, \eta_{j+l^* jd}, t+dt), \tag{14}$$

$$id = \mathrm{sgn}(1, d\xi_i), \quad jd = \mathrm{sgn}(1, d\eta_i), \tag{15}$$

where $id$ and $jd$ determine the interpolation direction and $a$ and $b$ are coefficients of the approximation equation and are calculated as

$$\begin{aligned}
a_{i,0} &= \frac{(|d\xi_i| - \Delta\xi)(|d\xi_i| - 2\Delta\xi)}{2\Delta\xi^2}, \\
a_{i,1} &= -\frac{(|d\xi_i|)(|d\xi_i| - 2\Delta\xi)}{\Delta\xi^2}, \\
a_{i,2} &= \frac{(|d\xi_i|)(|d\xi_i| - \Delta\xi)}{2\Delta\xi^2}, \\
b_{i,0} &= \frac{(|d\eta_i| - \Delta\eta)\,(|d\eta_i| - 2\Delta\eta)}{2\Delta\eta^2}, \\
b_{i,1} &= -\frac{(|d\eta_i|)(|d\eta_i| - 2\Delta\eta)}{\Delta\eta^2}, \\
b_{i,2} &= \frac{(|d\eta_i|)(|d\eta_i| - 2\Delta\eta)}{2\Delta\eta^2},
\end{aligned} \tag{16}$$

where $\Delta\xi$ and $\Delta\eta$ are lattice lengths in the curvilinear coordinate system and are calculated as

$$\Delta\xi = \xi_\infty/(NX - 1), \quad \Delta\eta = \eta_\infty/(NY - 1). \tag{17}$$

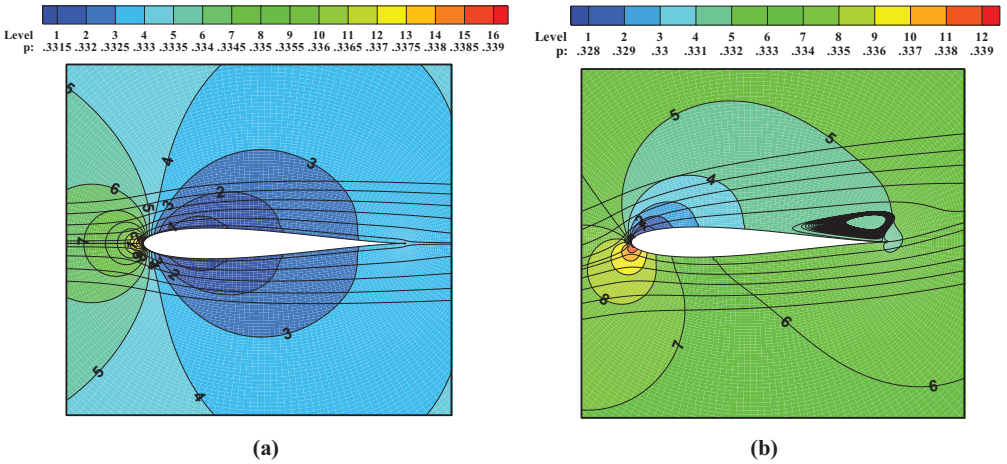
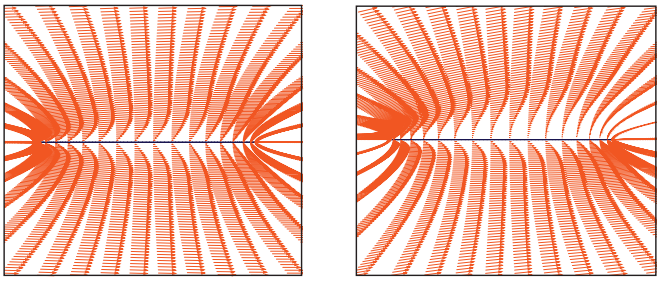



FIG. 8. (Color online) Pressure contours and streamlines around a symmetric airfoil for two different angles of attack: (a) 0° and (b) 10°.



(a) (b)

FIG. 9. (Color online) Velocity vectors around a flat plate for two different angles of attack: (a) 0° and (b) 10°.

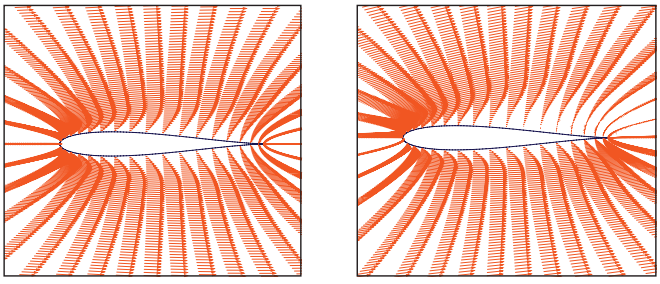
It should be noted that using interpolation would increase the stability of the numerical solution. This makes it possible to model fluid flow problems at higher Reynolds numbers.

## V. BOUNDARY CONDITIONS

The LBM performance is severely dependent on boundary conditions. In fact, the definition of a boundary condition directly effects the accuracy and stability of the method. Conventionally, three types of boundary conditions are defined for various boundaries of the flow field including periodic, nonreflecting, and bounce back.

Since we are using an O-type grid, there is a branch cut boundary starting at the trailing edge and normally a periodic boundary condition is applied on this boundary. For outflow boundaries, a nonreflecting boundary condition is used. Bounce back is employed on solid boundaries in order to satisfy the no-slip condition [19].

Halfway bounce back is a modified version of the full bounce back boundary condition that is applied with the definition of the virtual boundary inside the wall. We have employed this boundary condition on solid boundaries. To apply this condition, the solid boundaries are positioned between two series of points including inner points and outer points as shown in Fig. 6. The particles at inner points move toward the solid boundary and are reflected by the outer points. In fact, the no-slip boundary condition is imposed on the location between the outer and inner points and has second-order accuracy [20].



(a) (b)

FIG. 10. (Color online) Velocity vectors around a symmetric airfoil for two different angles of attack: (a) 0° and (b) 10°.

## VI. RESULTS AND DISCUSSION

The performance of the proposed method was evaluated by doing several test cases including flow around a flat pate and an airfoil. For all of these test cases the Reynolds number was 500 and the free stream velocity was 0.1 m/s. The far field pressure was set equal to 1/3, and the ratio of the streaming length to the radius of a cylinder $dx/a$ was 0.01. The time step, in units of $a/U$, was 0.001.

The results were compared with those of conventional CFD calculations, which were done by the authors using an in-house developed computer code. Some of the results of the code were reported in [21]. The CFD calculations were performed on a grid size of 125 × 90 with a time step size of 0.001 s for the flat plate problem; for the airfoil problem we used 150 × 100 and a time step size of 0.001 s.

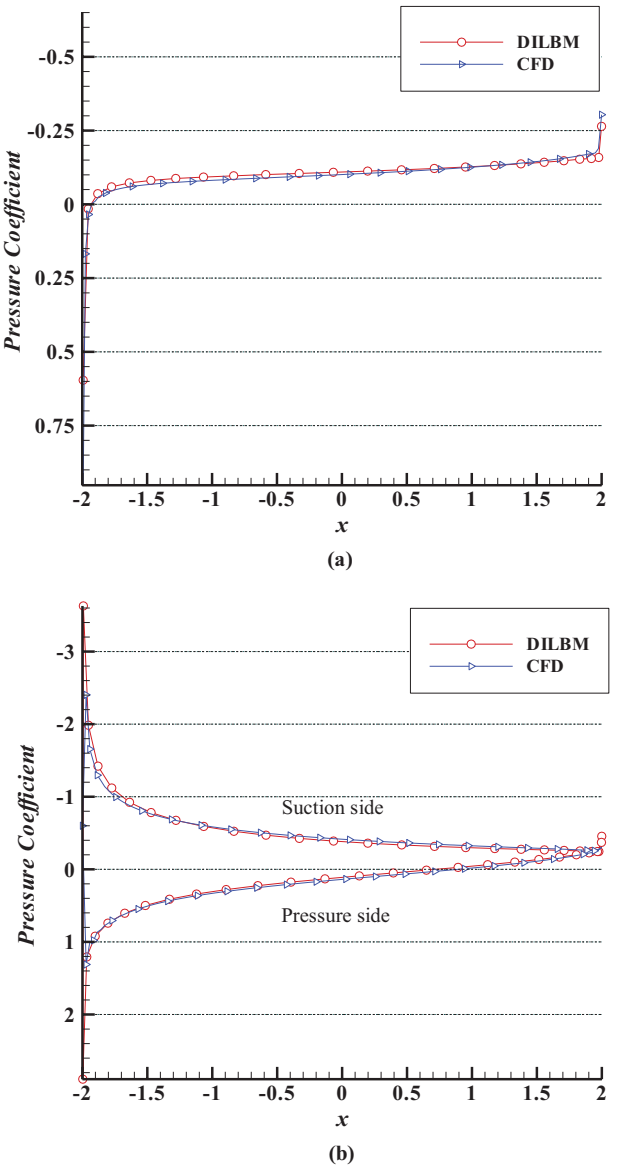



FIG. 11. (Color online) Variations of pressure coefficient along the flat plate surface for two different angles of attack: (a) 0° and (b) 10°.

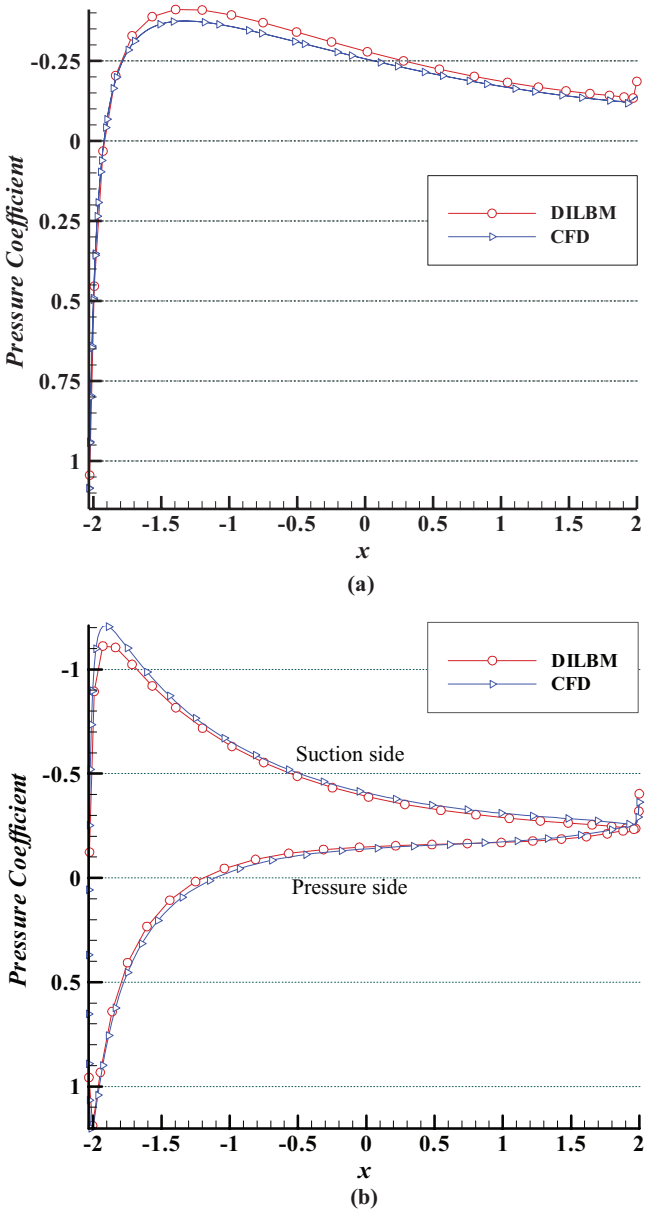



FIG. 12. (Color online) Variations of pressure coefficient along the symmetric airfoil surface for two different angles of attack: (a) 0° and (b) 10°.

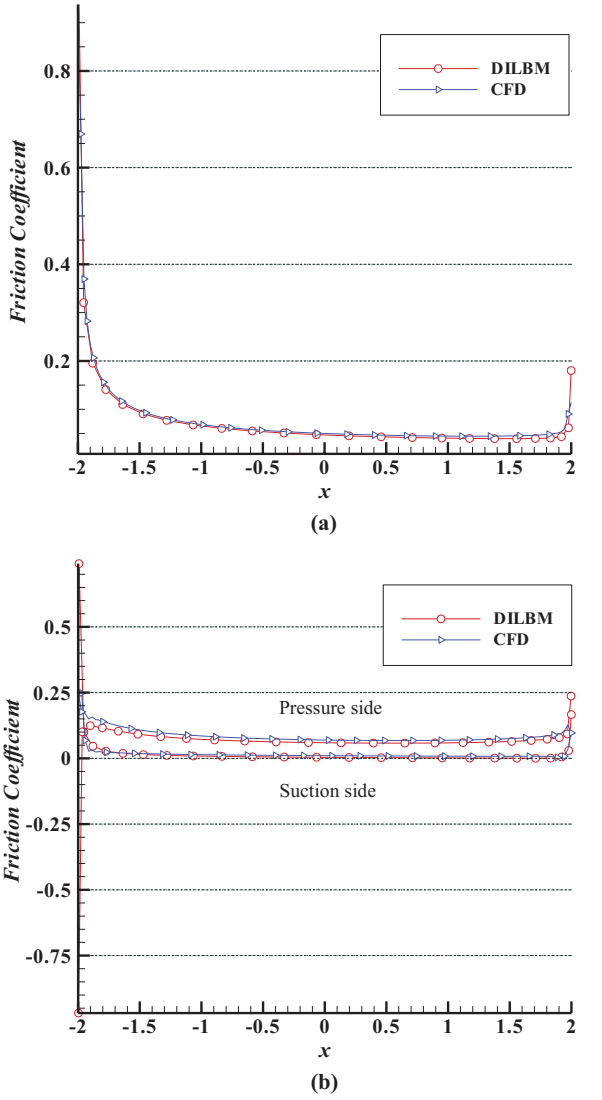



FIG. 13. (Color online) Variations of friction factor along the flat plate surface for two different angles of attack: (a) 0° and (b) 10°.

Figures 7 and 8 show pressure contours and streamlines around a flat plate and symmetric airfoil at two different angles of attack. It can be seen that the method can easily predict the vortices at the trailing edge region. Figures 9 and 10 depict velocity vectors around the flat plate and the airfoil at 0° and 10° angles of attack. The ability of the method in capturing the viscous boundary layer adjacent to the solid boundaries can be seen in these figures.

The pressure coefficient and fiction factor are defined as

$$C_p = \frac{p - p_\infty}{\frac{1}{2}\rho_\infty U_\infty^2}, \tag{18}$$

$$C_f = \frac{\tau}{\frac{1}{2}\rho_\infty U_\infty^2}, \tag{19}$$

TABLE I. Aerodynamic coefficients of the flat plate and the airfoil at zero angle of attack and Re = 500.

| Geometry | Method | $C_D$ | $C_{PS}$ |
|---|---|---|---|
| flat plate | present work | 0.139 | – |
| | CFD | 0.145 | – |
| airfoil | present work | 0.173 | 1.213 |
| | CFD | 0.170 | 1.152 |

TABLE II. Aerodynamic coefficients of the flat plate and the airfoil at 10° angle of attack and Re = 500.

| Geometry | Method | $C_D$ | $C_L$ | $C_{PS}$ |
|---|---|---|---|---|
| flat plate | present work | 0.204 | 0.695 | – |
| | CFD | 0.197 | 0.687 | – |
| airfoil | present work | 0.215 | 0.436 | 1.194 |
| | CFD | 0.212 | 0.441 | 1.146 |

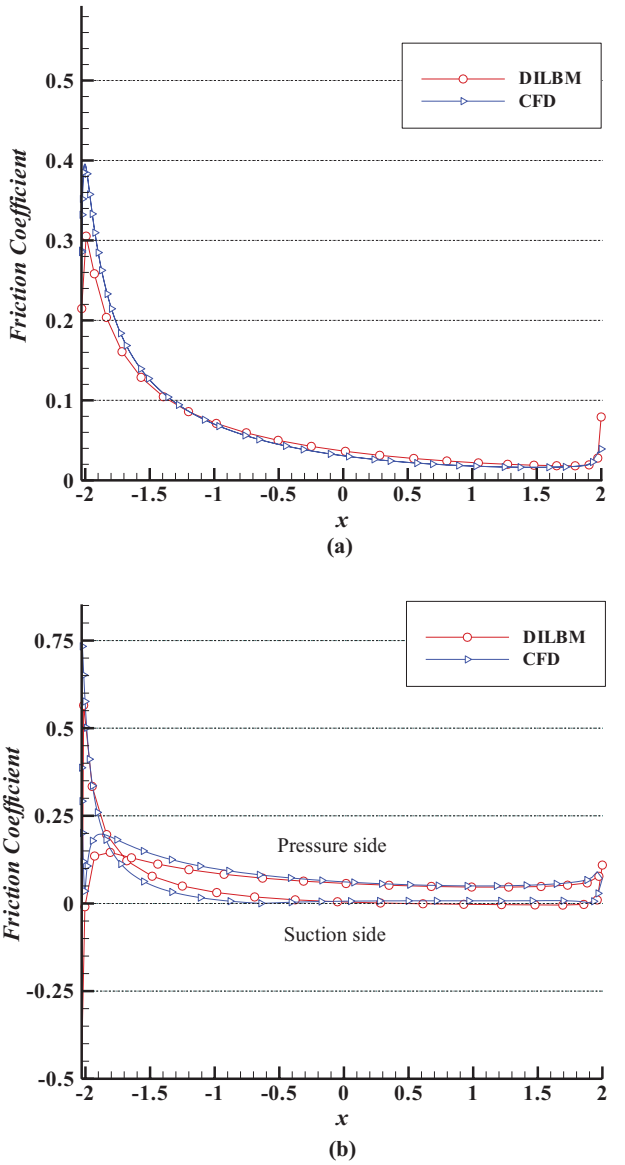



FIG. 14. (Color online) Variations of friction factor along the symmetric airfoil surface for two different angles of attack: (a) 0° and (b) 10°.

where p and $\tau$ are the static pressure and shear stress, respectively, and the subscript $\infty$ denotes the free stream conditions. The pressure coefficient along the suction and pressure sides of the flat plate and the airfoil are shown in Figs. 11 and 12, where the present results are compared with those of a conventional Navier-Stokes solver. There is very good agreement between the present results and those of the CFD calculations.

TABLE III. Grid convergence study of the airfoil.

| Grid | Resolution | $\Delta x_{\min}$ | Grid size ratio | $C_{PS}$ |
|---|---|---|---|---|
| A | 161 × 191 | $6.6 \times 10^{-4}$ | 0.68 | 1.237 |
| B | 181 × 211 | $5.4 \times 10^{-4}$ | 0.83 | 1.227 |
| C | 201 × 231 | $4.5 \times 10^{-4}$ | 1.00 | 1.219 |

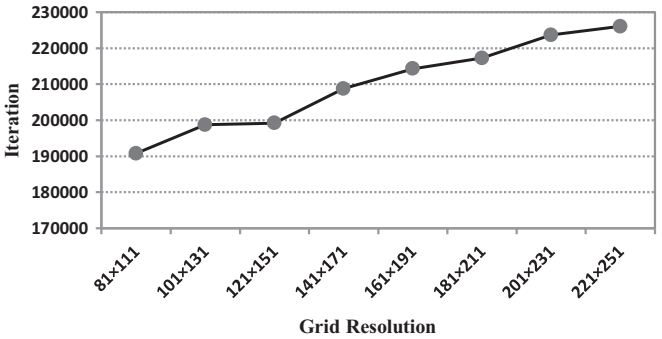



FIG. 15. Number of iterations to achieve convergence versus grid size.

Figures 13 and 14 give the friction factor along the pressure and suction sides of the airfoil and the flat plate at 0° and 10° angles of attack. Excellent agreement between the results of the present method and the CFD calculations is promising for application of the method for real engineering problems. Since the trailing edge of the Joukowski airfoil is a cusp, discontinuities appear in pressure coefficient and friction factor distributions at the trailing edge of the airfoil.

Tables I and II give the lift and the drag coefficients of the flat plate and airfoil and also the stagnation pressure coefficient $C_{ps}$ of the airfoil. These results are compared with the results of the conventional CFD solver.

Table I shows the results at zero angle of attack and the results of 10° are shown in Table II. There is considerably good agreement with the results of the CFD.

The accuracy of the presented scheme and also the interpolation used in the method was tested for the airfoil problem. The problem was solved in eight different grid sizes. Figure 15 shows the effect of grid size on the number of iterations to reach the steady-state conditions. It can be seen that the number of iterations increases with increasing grid size.

In order to validate the accuracy of the scheme, three additional computations are performed [22]. We chose the highest-resolution grid (the 201 × 231grid) as a reference grid and made two grids A and B with different resolutions, which

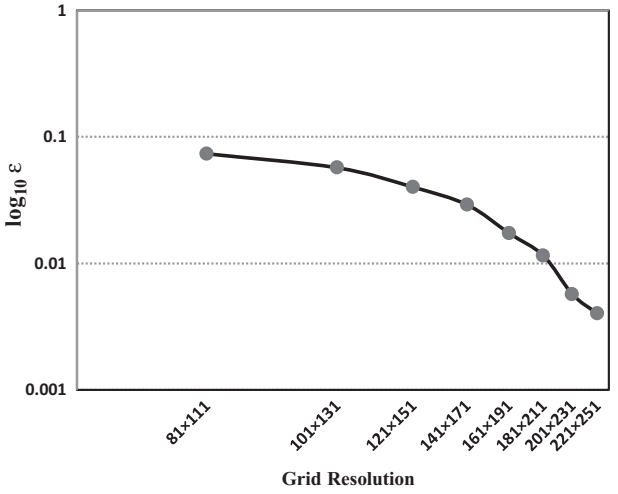



FIG. 16. A $\log_{10}$ plot of error versus grid size.

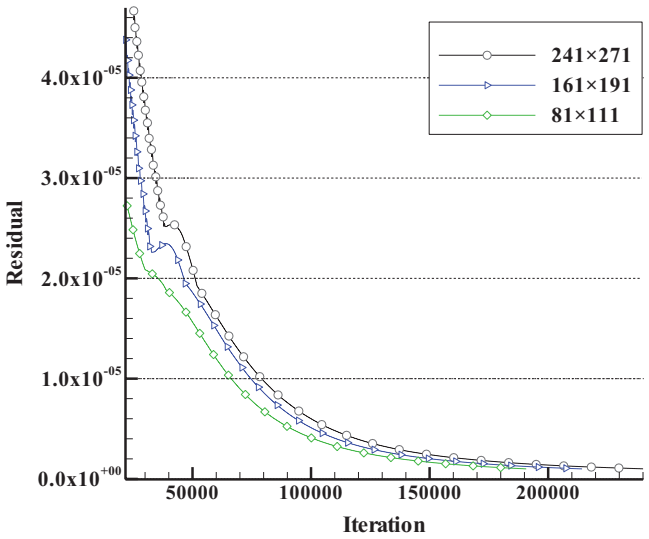



FIG. 17. (Color online) Convergence rate of the DILBM.

are listed in Table III. The minimum grid size normal to the wall boundary is defined by $0.01/\sqrt{\mathrm{Re}} = 4.5 \times 10^{-4}$ in order to clarify the effect of the minimum grid size. The ratio of the grid size between the different grids is not 2 due to the restriction in computational resources as it is usually required in the accuracy estimation [22]. Thus the evaluation method proposed by Roache [23] when the ratio of the grid size is not 2 is used here.

If we assume the exact solution as $\Phi$ and the solution on a grid size of $r$ as $\Phi_r$, $\Phi = \Phi_r + \alpha r^n$ should be satisfied, where $\alpha$ is a constant and $n$ is the order of the scheme. By comparing the results by three different grid resolutions, the exact solution can be eliminated as

$$\Phi_{rA} + \alpha r_A^n = \Phi_{rB} + \alpha r_B^n, \quad \Phi_{rB} + \alpha r_B^n = \Phi_{rC} + \alpha r_C^n, \tag{20}$$

where $\Phi_{rA}$, $\Phi_{rB}$, $\Phi_{rC}$ and $r_A$, $r_B$, $r_C$ are the solution and the grid size for grids A, B, and C, respectively. Since the ratio of the grid size between the different grids is not 2, the accuracy $n$ is calculated recursively by

$$\frac{\Phi_{rA} - \Phi_{rB}}{\Phi_{rB} - \Phi_{rC}} = \frac{r_B^n - r_A^n}{r_C^N - r_B^n}. \tag{21}$$

The pressure at the stagnation point $C_{ps}$ is compared to calculate the accuracy of the scheme; the effect of the additional interpolation does not appear when comparing the different grids.

From this analysis, the accuracy of the current code is $n = 1.95$, which is nearly 2. This shows that current scheme has second order of accuracy.

Figure 16 shows the accuracy of the method in prediction of the drag coefficient and the error in prediction of the coefficient. The error is calculated as

$$\varepsilon = \frac{C_d - C_{d_E}}{C_{d_E}}, \tag{22}$$

where $C_{d_E}$ and $C_d$ show exact and used method results, respectively. Note that the developed lattice Boltzmann code is fully explicit and the solution process is continued to reach steady-state conditions.

The convergence criterion is considered as

$$\delta = \max \frac{|u(t - \Delta t) - u(t)|}{|u(t)|} \leqslant 1.0 \times 10^{-7}. \tag{23}$$

Figure 17 shows the convergence rate of the method for three different grid sizes. Every calculation is performed until Eq. (23) is satisfied and $C_d$ converges to a constant value.

## VII. CONCLUSION

In this study the ISLBM was developed for fluid flow simulation around complex geometries using the body-fitted coordinate system. The method contains the inherent advantages of the standard LBM such as local calculations and consequently adaptively with parallel processing. The developed method is efficient and easily applicable for fluid flow problems with a curved boundary and nonuniform grid. Moreover, the method has embedded artificial viscosity, which leads to better stability. Since in the presented method the physical domain is independent of discretization of the velocity space, interpolation is mandatory and the accuracy of the results strongly depend on the type of approximation used for interpolation. Moreover, the interpolation coefficients are independent of the flow field and the interpolation step does not increase the computational time significantly.

The DILBM suppresses the pressure and friction factor oscillations because the grid points are located on the boundaries. The method can resolve the flow fields with higher Reynolds numbers and higher angles of attack and a camber airfoil in which the vortex shedding phenomenon occurs. Grid generation for transformable geometries is simple and there is no need to use traditional complex and time-consuming grid generation methods. The method can be used for resolving flow fields in a Cartesian grid system with clustering in high gradient regions. Therein the algebraic equations are replaced by the Joukowski transformation to make the connection between grid points in the computational domain and the physical domain in the algebraic grid generation methods.

Moreover, the presented idea gives promising results for the combination of the LBM with other aerodynamic mappings such as those by Eppler, von Kármán and Trefftz, Theodorsen, and van de Vooren. Using these combinations gives the ability to solve fluid flow around various types of airfoils.

[1] S. Succi, *The Lattice Boltzmann Equation for Fluid Dynamics and Beyond* (Oxford University Press, New York, 2001).
[2] Y. T. Chew, C. Shu, and X. D. Niu, J. Stat. Phys. **107**, 329 (2002).
[3] M. C. Sukop and D. T. Thome, *Lattice Boltzmann Modeling* (Springer, Berlin, 2006).
[4] G. R. McNamara and G. Zanetti, Phys. Rev. Lett. **61**, 2332 (1988).
[5] P. L. Bhatangar, E. P. Gross, and M. Krook, Phys. Rev. **94**, 511 (1954).
[6] X. He and L. S. Luo, Phys. Rev. E **55**, R6333 (1997).

[7] X. He, L. S. Luo, and M. Dembo, J. Comput. Phys. **129**, 357 (1996).
[8] X. He and G. Doolen, J. Comput. Phys. **134**, 306 (1997).
[9] R. Mei and W. Shyy, J. Comput. Phys. **143**, 426 (1998).
[10] G. Peng, H. Xi, C. Duncan, and S.-H. Chou, Phys. Rev. E **59**, 4675 (1999).
[11] H. Xi, G. Peng, and S. H. Chou, Phys. Rev. E **59**, 6202 (1999).
[12] T. Lee and C. L. Lin, J. Comput. Phys. **171**, 336 (2001).
[13] O. Fillipova and D. Hanel, J. Comput. Phys. **147**, 219 (1998).
[14] D. Yu, R. Mei, and W. Shyy, Int. J. Numer. Methods Fluids **39**, 99 (2002).
[15] J. Tölke and M. Krafczyk, Comput. Math. Appl. **58**, 898 (2009).
[16] M. Schönherr, K. Kucher, M. Geier, M. Stiebler, S. Freudiger, and M. Krafczyk, Comput. Math. Appl. **61**, 3730 (2011)
[17] J. Katz and A. Plotkin, *Low-Speed Aerodynamics* (Cambridge University Press, Cambridge, 2001).
[18] X. He, Int. J. Mod. Phys. C **8**, 737 (1997).
[19] T. Imamura, K. Suzuki, T. Nakamura, and M. Yoshida, J. Comput. Phys. **202**, 645 (2005).
[20] W. R. Zheng and F. H. Ping, Commun. Theor. Phys. **35**, 593 (2001).
[21] M. H. Hekmat, M. Mirzaei, and E. Izadpanah, J. Mech. Sci. Technol. **23**, 2479 (2009).
[22] J. H. Ferziger and M. Peric, *Computational Methods for Fluid Dynamics*, 3rd ed. (Springer, Berlin, 2002).
[23] P. J. Roache, J. Fluid Eng. **116**, 405 (1994).