EasyChair Preprint

№ 9453

Lattice Boltzmann Method for Simulating Flow over Moving Bodies

Van Duc Nguyen, Trung Thuc Ngo, Anh Tuan Le, Ich Long Ngo and Viet Dung Duong

December 11, 2022

Lattice Boltzmann method for simulating flow over moving bodies

Van Duc Nguyen[1], Trung Thuc Ngo[1], Anh Tuan Le[1], Ich Long Ngo[2,*], Viet Dung Duong[1,*]
[1] *School of Aerospace Engineering, University of Engineering and Technology, Vietnam National University, Hanoi, Vietnam*
[2] *School of Mechanical Engineering, Hanoi University of Science and Technology, No. 01, Dai Co Viet, Hai Ba Trung, Hanoi, Vietnam*
*Email: duongdv@vnu.edu.vn

Abstract. In engineering applications, designs of mechanical systems found in marine engineering, insect flight, and fish swimming, often meet the movements of the system's elements. These moving objects will contribute to the changes in hydrodynamics and thermodynamic coefficients. The complex motions are difficult to model because of complex geometry and high Reynolds number. In this study, we developed a fluid flow solver based on the Lattice Boltzmann method to resolve those difficulties. Validation of the solver is performed through two flow problems, including impulsively started flow past a single circular cylinder at Re=3000 and pitching airfoil with large amplitudes and frequencies. The obtained results are in good agreement with the experimental and computational results listed in the references.

Keywords: Lattice Boltzmann method, Moving boundary, Viscous Flow.

1. Introduction

In the computational fluid dynamics, the study of moving immersed objects is an extremely interesting topic. This study can be found in various investigation fields, including maritime engineering[1,2,3], the human heart[4,5], insect flying[6,8], and fish swimming[9,10,11]. Different methodologies for the development of the flow solver are proposed; one of these is the lattice Boltzmann method (LBM). Due to its simplicity and adaptability, the LBM gains a lot of interest.

LBM is constructed using the lattice Boltzmann equation (LBE)[12]. As a mesoscopic numerical approach, LBM provides an alternative approach for the fluid description. The LBM formulations are explicit algebraic equations, enabling accurate representation of nonlinear fluid systems. The solver can be simply paralleled on the high-performance computer because of LBE's independence. The interpolated bounce-back method is utilized to resolve the rigid wall's appearance[13]. However, this technique requires the remeshing of the grid at every step, leading to high computation costs. Therefore, the use of the immersed boundary method (IBM) is proposed. Although, this approach has been neglected for a long time before.

Following its adaption, IBM provides multiple versions for usage in LBM. These types had one thing in common the boundary movement is described through a forcing term added to the momentum equation. The correct interpretation of the presence of complex boundaries is a difficult task. It is possible to influence significantly the accuracy of the solver. The direct forcing method[14], [15] is a regularly utilized technique. For the evaluation of the forcing term, the direct forcing approach determines the forcing term by solving the momentum equation at the location where immersed boundaries exist. By adjusting the velocity at immersed boundary points, the appropriate velocity is found through interpolation and distribution functions. It is possible to directly extract the external force term from the momentum equations. When combined with a projection scheme, the boundary conditions can be satisfied approximatively in actual computations. Since there are no unknown variables that must be specified in the LBE formulation, the direct forcing is rather straightforward.

In this study, a solver is built by coupling the lattice Boltzmann method and the direct forcing method to model the moving objects in incompressible viscous flows. Although the solver has widely

been developed for bluff bodies, it does not apply to pitching airfoils [16]. Following is an outline of the remaining sections of this work. In Section 2, the traditional lattice Boltzmann and direct-forcing immersed boundary methods are presented briefly. In Section 3, two typical problems, including flow past a stationary circular cylinder and flow around a pitching airfoil, are simulated and compared with reference data to validate the robustness of the proposed solver. Section 4 concludes with a concluding summary.

2. Numerical method

2.1. The lattice Boltzmann method

The LBM is a mesoscopic dynamics-based technique. The fluid flow is made from a collection of pseudo-particles. On a discrete lattice grid, these particles perform constant streaming and collision. In this study, the particle's velocity is discretized to nine velocity vectors $\boldsymbol{c}_i$ at two-dimensional lattice locations. This velocity set known name the D2Q9 model, which is shown in Fig. 1. The vectors of velocity are:

$$\boldsymbol{c}_i = \begin{cases} (0,0) & i = 0, \\ (0,\pm 1)c, (\pm 1,0)c & i = 1,2,3,4, \\ (\pm 1,\pm 1)c & i = 5,6,7,8, \end{cases} \tag{1}$$

where $c = \Delta x/\Delta t$ represents the lattice speed, Δx is the lattice length, and Δt is the constant time step.

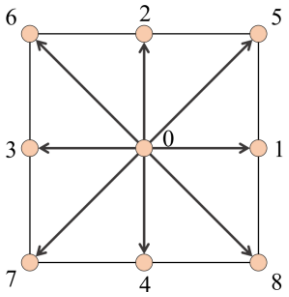

Figure 1. Sketch of the D2Q9 discrete velocity model.

The most important variable in the LBM is the particle distribution function $f_i(\boldsymbol{x}, t)$(PDF), which indicates the chance of meeting a particle with a velocity $\boldsymbol{c}_i$ at spatial point x and time t. The state of the fluid is updated by computing the particle distribution function using the discrete Boltzmann equation shown below

$$f_i(\boldsymbol{x} + \boldsymbol{c}_i\Delta t, t + \Delta t) = f_i(\boldsymbol{x}, t) - \frac{\Delta t}{\tau}\left[f_i(\boldsymbol{x}, t) - f_i^{eq}(\boldsymbol{x}, t)\right] \tag{2}$$

Here, the Bhatnagar-Gross-Krook collision model[17] with a single relaxation time is used and τ denotes the relaxation time. The term $f_i^{eq}(\boldsymbol{x}, t)$ represents the equilibrium distribution function, which has the following formula:

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

where $w_0 = 4/9, w_{1-4} = 1/9$ and $w_{5-8} = 1/36$ for D2Q9 lattice. $\rho(\boldsymbol{x}, t)$ and $\boldsymbol{u}(\boldsymbol{x}, t)$ denote the density and velocity of the macroscopic fluid, which may be computed using the following equations:

$$\begin{aligned} \rho &= \textstyle\sum_i f_i \\ \rho\boldsymbol{u} &= \textstyle\sum_i \boldsymbol{c}_i f_i \end{aligned} \tag{4}$$

The kinematic viscosity is linked to the relaxation time by the following formula:

$$\nu = c_s^2(\tau - \Delta t/2) \tag{5}$$

where c_s is the lattice sound speed. The pressure field may be calculated using the state equation for an ideal gas, $p = \rho c_s^2$.

2.2. The direct-forcing immersed boundary method

In the direct-forcing immersed boundary method, the boundary is represented by a collection of Lagrangian forcing points on the boundary. In contrast, the flow field is represented by Eulerian computational nodes including both the inside and outside of the boundary. Therefore, interpolation from surrounding nodes to boundary points is implemented for boundary force assessment and force distribution to neighboring nodes.

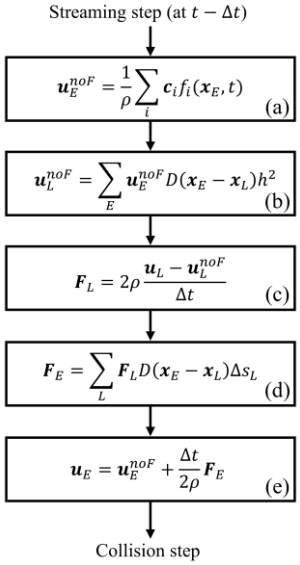

Figure 2. Algorithm for calculating the explicit direct-forcing IB -LBM

The direct-force LBM calculation process is shown in Figure 2. After the streaming steps, unforced velocities in Eulerian nodes ($\boldsymbol{u}_E^{noF}$) are estimated using the streamed particle distribution function in Step (a). Then, in Step (b), the unforced velocity at the boundary point ($\boldsymbol{u}_L^{noF}$) is determined by interpolating from nearby unforced velocities. In Step (c), the boundary force at the boundary point ($\boldsymbol{F}_L$) is calculated using the interpolated velocity ($\boldsymbol{u}_L^{noF}$) and the desired velocity ($\boldsymbol{u}_L$) specified by the no-slip condition in Step (c). Implementation of its propagation to surrounding nodes occurs in Step (d). Step (e) then updates the velocities of surrounding Eulerian nodes (forced). Here, Δs_L is the arc length of the boundary segment, D is the discrete delta function, and $h = \Delta x$ is the mesh spacing. The discrete delta functions D are given by:

$$D(\boldsymbol{x}_E - \boldsymbol{x}_L) = \frac{1}{h^2} d_h\left(\frac{\boldsymbol{x}_{iE} - \boldsymbol{x}_{iL}}{h}\right)\left(\frac{\boldsymbol{x}_{jE} - \boldsymbol{x}_{jL}}{h}\right) \tag{6}$$

with

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

Using IBM, the surface force of a body may be simply determined[18]

$$\boldsymbol{F} = -\sum \boldsymbol{F}(\boldsymbol{x}_L)\Delta s_L = -\sum \boldsymbol{F}(\boldsymbol{x}_E)h^2 \tag{8}$$

3. Validations

To validate the solver, a series of flow simulations through a single circular cylinder at $Re = 100$, 300, and 3000 together with a high-amplitude pitching airfoil at $Re = 3000$ is performed. Figure 3 illustrates flow configuration with computational domain and boundary conditions at the inlet, outlet, upper, and lower boundaries of both problems. The domain of computation is $100D \times 150D$ so the blockage is 1%, which is less than the required blockage ratio threshold of 6% [19]. For the configuration of boundary conditions displayed, Dirichlet-type and Neumann-type boundary conditions are applied to the inflow and outflow boundaries, respectively, while free-slip boundary conditions are applied to the upper and lower boundaries.

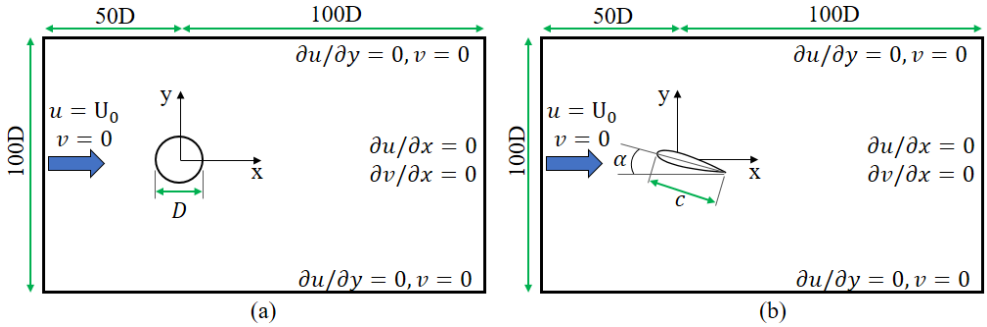

Figure 3. Sketch of the computational domain for two problems: (a) flow past a circular cylinder and (b) flow past a pitching airfoil.

3.1. Flow past a stationary circular cylinder

Table 1. Literature comparison of force coefficient for a circular cylinder at $Re = 100$ and 300.

	$\bar{C}_D$		C'_L		St	
Re	100	300	100	300	100	300
Ghias et al.[20]	1.36	1.40	0.32	0.67	0.16	0.21
De Palma et al.[21]	1.32	-	0.23	-	0.16	-
Rajani et al.[22]	1.33	1.28	0.17	0.60	0.15	0.21
Boukharfane et al.[23]	1.36	1.26	0.25	0.62	0.16	0.21
Vanna et al.[24]	1.32	1.34	0.22	0.63	0.16	0.21
Present	1.33	1.31	0.24	0.62	0.16	0.21

Now we examine the solver for flow past a circular cylinder. This is one of the famous benchmark problems for the evaluation of the accuracy of numerical method complex geometries. Thus, there are a large number of comparable results from various numerical methods. In this problem, the flow pattern changes on the Reynolds number, $Re = U_0 D/\nu$, where U_0 is the freestream velocity and D is the diameter of the cylinder.

Table 1 shows the results adopted by the solver compared with available references. Regarding the studies at Re = 100 and Re = 300, the intrinsic unsteadiness of the flow must be statistically resolved. In particular, we focused on three quantities including the time-averaged drag coefficient, the root mean square of the lift coefficient, and the Strouhal number of the wake, which is calculated as $St = fD/U_0$. Here, f is the frequency of the vortex shedding. The present results obtained by the solver always conform provided by other investigations. Figure 4 shows the temporal evolution of the lift coefficients,

drag coefficients, and the nondimensional instantaneous vorticity field $\omega^* = (\partial u/\partial y - \partial v/\partial x)D/U_0$ at $Re = 100$. According to the figure, C_L and C_D has comparatively pure temporal histories. Similarly, the vorticity fields exhibit neither oscillations nor noise. In Table 2, a mesh convergence study is performed at $Re = 100$ with various grid spacing $D = 20\Delta x$, $40\Delta x$, $60\Delta x$, and $80\Delta x$. Here, the force coefficients are indicated as functions of $D/\Delta x$. These results indicate that the C_d, C_L', and St converge to 1.335, 0.241, and 0.163, respectively. These values are in agreement with previous numerical and experimental studies, as detailed in Table 2.

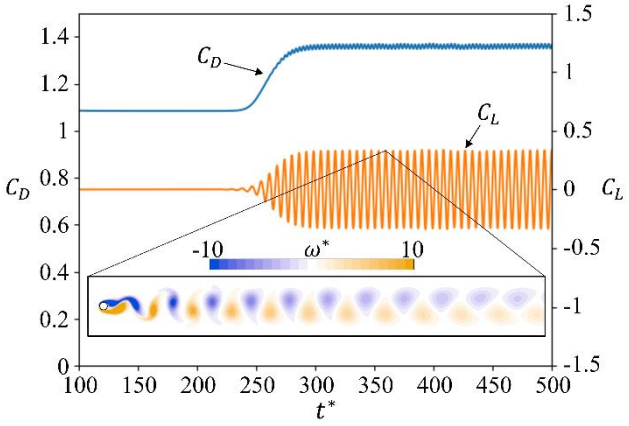

Figure 4. The time history of lift and drag coefficient for a circular cylinder at $Re = 100$; the instantaneous vorticity at $t^* = 358$.

Table 2. Mesh convergence results in a single circular cylinder at $Re = 100$. Evolution of the force coefficients as functions of the mesh resolution $D/\Delta x$.

$D/\Delta x$	$\bar{C}_D$	C_L'	St
20	1.428	0.211	0.172
40	1.361	0.237	0.164
60	1.331	0.244	0.163
80	1.335	0.241	0.163

Figure 5 shows the comparison of the instantaneous vorticity of impulsive flow past a single circular cylinder at $Re = 3000$. In this case, the nondimensional time is defined by $t^* = tU_0/R$ with $R = D/2$. Here, the evolution of the vorticity is presented at different nondimensional times (t^*=1.0, 3.0, and 5.0). Initially, the primary vorticity is formed at the rear side of the cylinder. The strong secondary vorticity that is created on the surface of the cylinder attempts to penetrate the primary vortex between $t^* = 1.0$ and $t^* = 3.0$. This secondary vorticity generates tertiary vortices on the cylinder's surface. This study captures the flow dynamics accurately. The present results of the surface vorticity are in excellent agreement with Lee [25]. A mesh convergence investigation is performed at $Re = 3000$ with different grid spacings $D = 100\Delta x$, $200\Delta x$, $300\Delta x$, and $400\Delta x$ is presented in Table 3. Acronym W is the width between two primary vortex centers and L is the length from the primary vortex center to the circle center along the x-axis. These results indicate that the L and W converge to 0.803 and 0.585, respectively. Deviations are extremely small and <4%. This shows the applicability of the solver for high Reynolds numbers, such as $Re = 3000$.

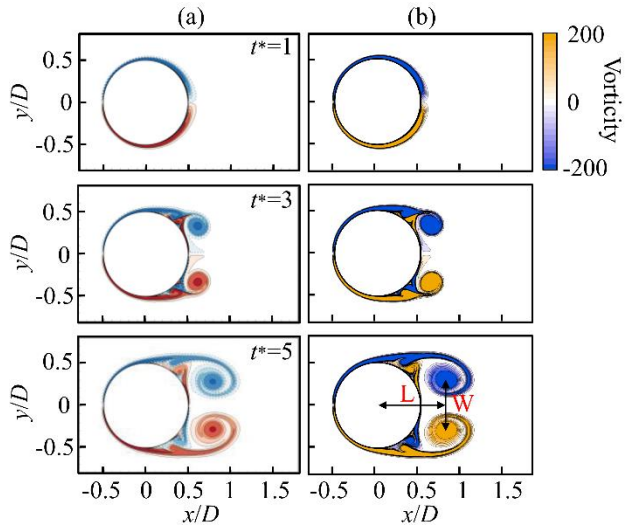

Figure 5. Comparison of impulsive flow past a stationary circular cylinder at $Re = 3000$ at $t^* = 1$, $t^* = 3$ and $t^* = 5$: the results of reference[25](left) and the present results (right).

Table 3. Mesh convergence results in a single circular cylinder at $Re = 3000$. L, W denote the length and width of the coupled vortex.

$D/\Delta x$	L		W	
100	0.816	1.6%	0.563	3.7%
200	0.808	0.6%	0.577	1.3%
300	0.803	0%	0.584	0.1%
400	0.803	0%	0.585	0%

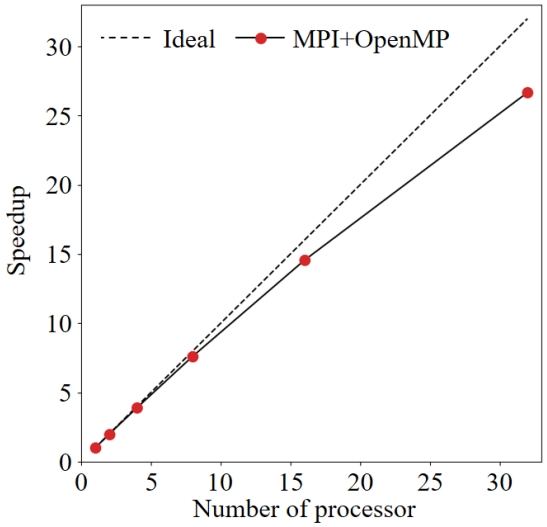

Figure 6. Parallel speedup as a function of the number of processors obtained from a 32-core Linux computer.

Figure 6 shows that the speedup of the impulsively started cylinder problem is plotted on a linear scale. The dotted line represents the ideal speedup. The straight dotted line represents the actual speedup while the solver is paralleled by OpenMP techniques on the Message-Passing Interface (MPI)

environment. There is a gradual increase in actual speedup. This result confirms that the solver provides a significant parallel speedup.

3.2. Pitching airfoil test

In this section, we investigate the large-amplitude pitching airfoil at $Re = 4000$. The airfoil is chosen as a NACA 0018 airfoil. The employed asymmetric pitching motion is a modified version of Theodorsen's theory, which assumes small and sinusoidal oscillations. The lift force is accurately predicted through Theodorsen's theory. The acceleration time is $t_a^* = 0.15T^*$, where $T^* = f_p^{-1}$ is the pitching period. The kinematic change is composed of piecewise functions in which the acceleration components are fourth-order polynomials:

$$\alpha = \begin{cases} \dot{a}_1 t^*, & (0 \le t^* < t_1^*), \\ \frac{\dot{a}_1}{2t_a^{*3}}(t^* - t_2^*)^4 + \frac{\dot{a}_1}{t_a^2}(t^* - t_2^*)^3 + \alpha_0, & (t_1^* \le t^* < t_2^*), \\ -\frac{\dot{a}_2}{2t_a^3}(t^* - t_2^*)^4 + \frac{\dot{a}_2}{t_a^{*2}}(t^* - t_2^*)^3 + \alpha_0, & (t_2^* \le t^* < t_3^*), \\ \dot{a}_2(t^* - t_4^*)^4 - \frac{\dot{a}_2 t_a^*}{2} - \alpha_0, & (t_3^* \le t^* < t_4^*), \\ \frac{\dot{a}_2}{2t_a^{*3}}(t^* - t_5^*)^4 + \frac{\dot{a}_2}{t_a^{*2}}(t^* - t_5^*)^3 - \alpha_0, & (t_4^* \le t^* < t_5^*), \\ -\frac{\dot{a}_1}{2t_a^{*3}}(t^* - t_5^*)^4 + \frac{\dot{a}_1}{t_a^{*2}}(t^* - t_5^*)^3 - \alpha_0, & (t_5^* \le t^* < t_6^*), \\ \dot{a}_1(t^* - T^*), & (t_6^* \le t^* < T^*), \end{cases} \tag{9}$$

where $\dot{a}_1$ and $\dot{a}_2$ are pitch rates in regions $(0 \le t^* < t_1^*)$ and $(t_3^* \le t^* < t_4^*)$, respectively. Both parameters are given by

$$\begin{aligned} \dot{a}_1 &= \frac{2\alpha_0}{\xi T - t_a}, \\ \dot{a}_2 &= -\frac{2\alpha_0}{(1-\xi)T - t_a}. \end{aligned} \tag{10}$$

The asymmetry parameter ξ controls the angle of attack α to peaks of α_0 at $t_2^* = 0.5\xi$. In this study, the reduced frequency $k = \pi f_p c/U_0 = 0.22$, pitching amplitude $\alpha_0 = 64°$, and asymmetry parameter $\xi = 0.4$.

Figure 7a shows the time evolution of the lift coefficient in a period. According to the figure, the value of C_L close approximately to the experimental result[26]. The peaks of force occur at the start of the acceleration phase. As the angle of attack diminishes until it approaches $45°$, a plateau zone forms. This approximation demonstrated good agreement for the examined pitch rate. Figure 7b to 7f shows the evolution of instantaneous vorticity on the airfoil surface.

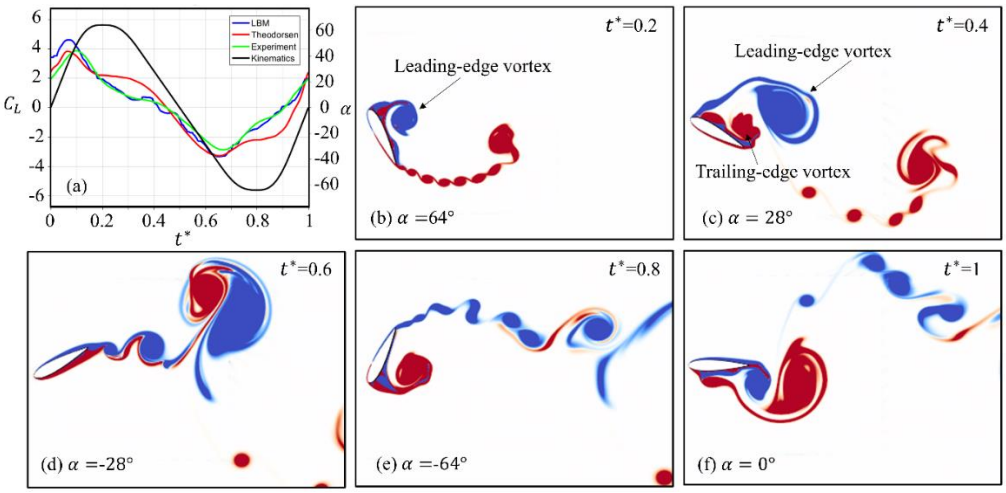

Figure 7. Comparison of the lift coefficient with experimental results and Theodorsen's theory (a); Time evolution of wake on the surface of the airfoil (b-f).

4. Conclusion

In this study, a direct-forcing IB-LBM solver was built and assessed through flow problems with flow past a circular cylinder and a pitching airfoil. The results indicate the ability to simulate truly arbitrarily complex geometries (static or in motion) in an incompressible viscous flow. The framework uses well-established computational geometry methods to perform all required geometry-related calculations. The solver's approach can be highly effective for modeling realistic insect wings with corrugations and sharp features. The above exams demonstrated the accuracy and efficiency of the proposed solver.

References

[1] H. S. Udaykumar, R. Mittal, P. Rampunggoon, and A. Khanna, "A Sharp Interface Cartesian Grid Method for Simulating Flows with Complex Moving Boundaries," *J. Comput. Phys.*, vol. 174, no. 1, pp. 345–380, Nov. 2001, doi: 10.1006/jcph.2001.6916.

[2] D. Kim and H. Choi, "Immersed boundary method for flow around an arbitrarily moving body," *J. Comput. Phys.*, vol. 212, no. 2, pp. 662–680, Mar. 2006, doi: 10.1016/j.jcp.2005.07.010.

[3] C. H. K. Williamson and R. Govardhan, "Vortex-induced vibrations," *Annu. Rev. Fluid Mech.*, vol. 36, no. 1, pp. 413–455, Jan. 2004, doi: 10.1146/annurev.fluid.36.050802.122128.

[4] C. S. Peskin, "Numerical analysis of blood flow in the heart," *J. Comput. Phys.*, vol. 25, no. 3, pp. 220–252, Nov. 1977, doi: 10.1016/0021-9991(77)90100-0.

[5] C. A. Taylor and M. T. Draney, "Experimental and computational methods in cardiovascular fluid mechanics," *Annu. Rev. Fluid Mech.*, vol. 36, no. 1, pp. 197–231, Jan. 2004, doi: 10.1146/annurev.fluid.36.050802.121944.

[6] A. Gilmanov and F. Sotiropoulos, "A hybrid Cartesian/immersed boundary method for simulating flows with 3D, geometrically complex, moving bodies," *J. Comput. Phys.*, vol. 207, no. 2, pp. 457–492, Aug. 2005, doi: 10.1016/j.jcp.2005.01.020.

[7] Z. J. Wang, "DISSECTING INSECT FLIGHT," *Annu. Rev. Fluid Mech.*, vol. 37, no. 1, pp. 183–210, Jan. 2005, doi: 10.1146/annurev.fluid.36.050802.121940.

[8] H. Liu and K. Kawachi, "A Numerical Study of Insect Flight," *J. Comput. Phys.*, vol. 146, no. 1, pp. 124–156, Oct. 1998, doi: 10.1006/jcph.1998.6019.

[9] R. Mittal, H. Dong, M. Bozkurttas, F. M. Najjar, A. Vargas, and A. von Loebbecke, "A versatile sharp interface immersed boundary method for incompressible flows with complex boundaries," *J. Comput. Phys.*, vol. 227, no. 10, pp. 4825–4852, May 2008, doi: 10.1016/j.jcp.2008.01.028.

[10] F.-B. Tian, H. Luo, L. Zhu, J. C. Liao, and X.-Y. Lu, "An efficient immersed boundary-lattice Boltzmann method for the hydrodynamic interaction of elastic filaments," *J. Comput. Phys.*, vol. 230, no. 19, pp. 7266–7283, Aug. 2011, doi: 10.1016/j.jcp.2011.05.028.

[11] C. Wu and L. Wang, "Where is the rudder of a fish?: the mechanism of swimming and control of self-propelled fish school," *Acta Mech. Sin.*, vol. 26, no. 1, pp. 45–65, Mar. 2010, doi: 10.1007/s10409-009-0305-z.

[12] S. Succi, *The lattice Boltzmann equation for fluid dynamics and beyond*. Oxford : New York: Clarendon Press ; Oxford University Press, 2001.

[13] V. D. Duong, V. D. Nguyen, V. T. Nguyen, and I. L. Ngo, "Low-Reynolds-number wake of three tandem elliptic cylinders," *Phys. Fluids*, vol. 34, no. 4, p. 043605, Apr. 2022, doi: 10.1063/5.0086685.

[14] Z.-G. Feng and E. E. Michaelides, "The immersed boundary-lattice Boltzmann method for solving fluid–particles interaction problems," *J. Comput. Phys.*, vol. 195, no. 2, pp. 602–628, Apr. 2004, doi: 10.1016/j.jcp.2003.10.013.

[15] Z.-G. Feng and E. E. Michaelides, "Proteus: a direct forcing method in the simulations of particulate flows," *J. Comput. Phys.*, vol. 202, no. 1, pp. 20–51, Jan. 2005, doi: 10.1016/j.jcp.2004.06.020.

[16] S. Tao, Q. He, B. Chen, and S. Huang, "A distribution function correction-based immersed boundary- lattice Boltzmann method with truly second-order accuracy for fluid-solid flows." arXiv, Mar. 25, 2018. Accessed: Nov. 15, 2022. [Online]. Available: http://arxiv.org/abs/1803.09380

[17] 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 Systems," *Phys. Rev.*, vol. 94, no. 3, pp. 511–525, May 1954, doi: 10.1103/PhysRev.94.511.

[18] M.-C. Lai and C. S. Peskin, "An Immersed Boundary Method with Formal Second-Order Accuracy and Reduced Numerical Viscosity," *J. Comput. Phys.*, vol. 160, no. 2, pp. 705–719, May 2000, doi: 10.1006/jcph.2000.6483.

[19] Q. Zheng and Md. M. Alam, "Intrinsic features of flow past three square prisms in side-by-side arrangement," *J. Fluid Mech.*, vol. 826, pp. 996–1033, Sep. 2017, doi: 10.1017/jfm.2017.378.

[20] R. Ghias, R. Mittal, and T. Lund, "A Non-Body Conformal Grid Method for Simulation of Compressible Flows with Complex Immersed Boundaries," in *42nd AIAA Aerospace Sciences Meeting and Exhibit*, Reno, Nevada, Jan. 2004. doi: 10.2514/6.2004-80.

[21] P. De Palma, M. D. de Tullio, G. Pascazio, and M. Napolitano, "An immersed-boundary method for compressible viscous flows," *Comput. Fluids*, vol. 35, no. 7, pp. 693–702, Aug. 2006, doi: 10.1016/j.compfluid.2006.01.004.

[22] B. N. Rajani, A. Kandasamy, and S. Majumdar, "Numerical simulation of laminar flow past a circular cylinder," *Appl. Math. Model.*, vol. 33, no. 3, pp. 1228–1247, Mar. 2009, doi: 10.1016/j.apm.2008.01.017.

[23] R. Boukharfane, F. H. Eugênio Ribeiro, Z. Bouali, and A. Mura, "A combined ghost-point-forcing / direct-forcing immersed boundary method (IBM) for compressible flow simulations," *Comput. Fluids*, vol. 162, pp. 91–112, Jan. 2018, doi: 10.1016/j.compfluid.2017.11.018.

[24] F. D. Vanna, F. Picano, and E. Benini, "A sharp-interface immersed boundary method for moving objects in compressible viscous flows," *Comput. Fluids*, vol. 201, p. 104415, Apr. 2020, doi: 10.1016/j.compfluid.2019.104415.

[25] S.-J. Lee, "Numerical Simulation of Vortex-Dominated Flows Using the Penalized VIC Method," in *Vortex Dynamics and Optical Vortices*, H. Perez-De-Tejada, Ed. InTech, 2017. doi: 10.5772/65371.

[26] S. Ōtomo, S. Henne, K. Mulleners, K. Ramesh, and I. M. Viola, "Unsteady lift on a high-amplitude pitching aerofoil," *Exp. Fluids*, vol. 62, no. 1, p. 6, Jan. 2021, doi: 10.1007/s00348-020-03095-2.