19th Australasian Fluid Mechanics Conference
Melbourne, Australia
8-11 December 2014

Immersed Boundary Method and its Applications in a Variety of Complex Flow Problems

F. B. Tian, J. Young and J. C. S. Lai

School of Engineering and Information Technology
The University of New South Wales, Canberra, ACT 2600, Australia

Abstract

The immersed boundary (IB) method is a novel strategy to treat the boundary condition of a solid immersed in a fluid. In the original IB method, the forces exerted by the solid on the fluid are spread to grid points in the vicinity of the solid boundary in order to account for the effect of the solid. Then the Navier-Stokes equations with additional body forces are solved on a Cartesian mesh. This treatment can get reasonable velocity distribution near the fluid-solid interface. The primary advantage of the IB method is that the grid generation is greatly simplified and the mesh movement/regeneration is avoided.

Since the initial idea of the IB method, many additional features have been developed to enhance the capability and to improve the performance of the method. In this paper, we will introduce our recent developments for the IB solvers: the IB method based on the lattice Boltzmann method, the sharp-interface IB method based on the finite difference method, and extensions to other physical equations. A variety of applications of the IB solvers will be demonstrated. The applications include insect flight, fish swimming, red blood cells, fluid-structure interaction during phonation, heat transfer and electrodynamics.

Introduction

Numerical methods based on fixed grids have attracted growing interest in recent years due to their advantages in handling complex/moving boundaries. The immersed boundary (IB) method, which is the most notable among them, has gained popularity for a wide range of applications in recent years. The IB method, first developed by Peskin [8], is a novel strategy to treat the boundary condition of a solid immersed in a fluid. In the original version of the IB method, a continuous force is distributed as a body-force term onto the volumetric mesh in the vicinity of the boundary in order to account for the effect of the boundary. The Navier-Stokes equations with additional body forces are then discretized on a fixed Cartesian grid. Later, several families of the IB method have been developed, examples are the direct forcing approach based on local flow reconstruction in [7,21,24] and the projection approach by Taira & Colonius [12]. The underlying ideas of these works are very different depending on the specific implementations. Nevertheless, all of these versions of the method are able to treat the irregular and time-varying boundaries using a fixed, single-block Cartesian grid. Therefore, they share the merits of simple grid generation, efficient computation on the structured grid, and easy partition based on domain decomposition.

Given its advantages, the IB method is particularly suitable for simulation of the flows involving complex geometries and large-deformation boundaries, for example the biological and biomedical fluid–structure interaction (FSI) problems at the tissue and organ levels [15]. In this paper, we will introduce our recent progresses in developing the IB method and its applications in insect flight, fish swimming, red blood cells, FSI during phonation, heat transfer and electrodynamics.

Immersed boundary–lattice Boltzmann method

In this method, the fluid flow is solved by the lattice Boltzmann (LB) method. In the present LB method, the kinematics of the fluid is governed by the discrete LB equation of a single relaxation time [2,17],

$$g_i(\mathbf{x}+\mathbf{e}_i\Delta t, t+\Delta t) - g_i(\mathbf{x},t) = -\frac{1}{\tau}[g_i(\mathbf{x},\mathbf{t}) - g_i^{eq}(\mathbf{x},\mathbf{t})] + \Delta t G_i, \qquad (1)$$

where $g_i(\mathbf{x},t)$ is the distribution function for particles with velocity $\mathbf{e}_i$ at position $\mathbf{x}$ and time t, Δt is the size of the time step, $g_i^{eq}(\mathbf{x},\mathbf{t})$ is the equilibrium distribution function, τ represents the nondimensional relaxation time, and G_i is the term representing the body force effect on the distribution function. In Eq. (1), g_i^{eq} and G_i are calculated according to

$$\begin{aligned} g_i^{eq} &= \omega_i\rho\left[1+\frac{\mathbf{e}_i\cdot\mathbf{u}}{c_s^2}+\frac{\mathbf{uu}:(\mathbf{e}_i\mathbf{e}_i-c_s^2\mathbf{I})}{2c_s^4}\right], & (2)\\ G_i &= \left(1-\frac{1}{2\tau}\right)\omega_i\left[\frac{\mathbf{e}_i-\mathbf{u}}{c_s^2}+\frac{\mathbf{e}_i\cdot\mathbf{u}}{c_s^4}\mathbf{e}_i\right]\cdot\mathbf{f}, & (3)\end{aligned}$$

where ω_i are the weighing factors, $\mathbf{u}$ is the velocity of the fluid, c_s is the speed of sound defined by $c_s = \Delta x/\sqrt{3}\Delta t$, and $\mathbf{f}$ is the body force acting on the fluid. The relaxation time τ is related to the kinematic viscosity ν in the Navier–Stokes equations:

$$\nu = (\tau-0.5)c_s^2\Delta t. \qquad (4)$$

In the 2D nine-speed (D2Q9) model [11], as shown in figure 1(a), the nine particle velocities are given by

$$\begin{aligned}\mathbf{e}_0 &= (0,0),\\ \mathbf{e}_i &= \left(\cos\frac{\pi(i-1)}{2},\sin\frac{\pi(i-1)}{2}\right)\frac{\Delta x}{\Delta t},\ i=\text{1-4},\\ \mathbf{e}_i &= \left(\cos\frac{\pi(i-9/2)}{2},\sin\frac{\pi(i-9/2)}{2}\right)\frac{\sqrt{2}\Delta x}{\Delta t},\ i=\text{5-8},\end{aligned}$$

where Δx is the lattice spacing. The weight factors are given by $\omega_0 = 4/9$, $\omega_i = 1/9$ for $i = 1$ to 4 and $\omega_i = 1/36$ for $i = 5$ to 8. In the 3D nineteen-speed (D3Q19) model [11] shown in figure 1(b), the particle velocities are defined by

$$\begin{aligned}\mathbf{e}_0 &= (0,0,0),\\ \mathbf{e}_i &= [(\pm1,0,0),(0,\pm1,0),(0,0,\pm1)]\frac{\Delta x}{\Delta t},\ i=\text{1-6},\\ \mathbf{e}_i &= [(\pm1,\pm1,0),(\pm1,0,\pm1),(0,\pm1,\pm1)]\frac{\Delta x}{\Delta t},\ i=\text{7-18}.\end{aligned}$$

The weight factors of D3Q19 model take the values $\omega_0 = 1/3$, $\omega_i = 1/18$ for $i = 1$ to 6 and $\omega_i = 1/36$ for $i = 7$ to 18. The values of $\mathbf{e}_i$ ensure that within one time step, a fluid particle moves to one of the neighboring nodes as shown in figure 1, or stays at its current location.

Once the particle density distribution is known, the fluid density, velocity and pressure are then computed from

$$\rho = \sum_i g_i,\ \mathbf{u} = \frac{\sum_i \mathbf{e}_i g_i + \frac{1}{2}\mathbf{f}\Delta t}{\rho},\ p = \rho c_s^2. \qquad (5)$$

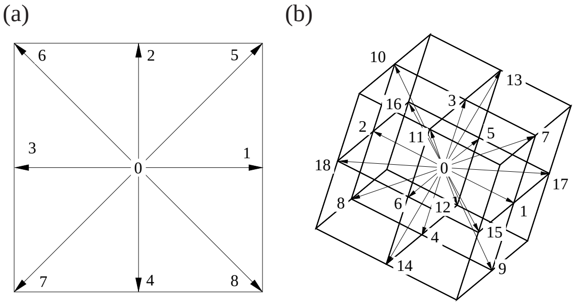

Figure 1: Lattice Boltzmann models: (a) Nine base vectors representing 9 possible velocity directions in the D2Q9 lattice model; (b) Nineteen base vectors representing 19 possible velocity directions in the D3Q19 lattice model.

The energy of the plate can be divided into elastic energy due to deformation, E_p, and the kinetic energy, E_k, which are defined as,

$$E_p = \frac{1}{2}\int \sum_{i,j=1}^{n_d} [\psi_{ij}(T_{ij}-\delta_{ij})^2+\gamma_{ij}(B_{ij})^2]dA, \qquad (6)$$

$$E_k = \frac{1}{2}m_p \int |\frac{\partial \mathbf{X}}{\partial t}|^2 dA, \qquad (7)$$

where m_p is the density of the plate, n_d is the dimension of the plate, $T_{ij} = \left(\partial \mathbf{X}/\partial s_i \cdot \partial \mathbf{X}/\partial s_j\right)^{1/2}$ is the stretching and shearing effects, $B_{ij} = \left(\partial^2 \mathbf{X}/\partial s_i \partial s_j \cdot \partial^2 \mathbf{X}/\partial s_i \partial s_j\right)^{1/2}$ is the bending and twisting effects, ψ_{ij} is the stretching and shearing coefficients, γ_{ij} is the bending and twisting coefficients and summation convention is not applied on both i and j. For 1D filament, $n_d = 1$ and the integral is taken along the filament, while for 2D plate, $n_d = 2$ and the integral is taken over the whole plate. The potential energy of external force (hydrodynamic force in the present work) is expressed by

$$E_h = \int \mathbf{F}_f \cdot (\mathbf{X}-\mathbf{X}^0)dA, \qquad (8)$$

where $\mathbf{X}^0$ is the initial position of the structure. Then the total potential energy of the plate can be defined by $\Pi = E_k - E_p + E_h$. By using the Hamilton's principle together with the variational derivative of the total potential energy, the governing equation for the plate is obtained,

$$m_p \frac{\partial^2 \mathbf{X}}{\partial t^2} + \sum_{i,j=1}^{n_d}\left[-\frac{\partial}{\partial s_i}\left(\sigma_{ij}\frac{\partial \mathbf{X}}{\partial s_j}\right)+\frac{\partial^2}{\partial s_i \partial s_j}\left(\gamma_{ij}\frac{\partial^2 \mathbf{X}}{\partial s_i \partial s_j}\right)\right] = \mathbf{F}_f, \qquad (9)$$

where $\sigma_{ij} = \psi_{ij}(1-\delta_{ij}/T_{ij})$. Note that Eq. (9) is equivalent to that in Ref. [17,25] for 1D filament and Ref. [5,16] for 2D plate if the stretching and shearing effects are small.

Explicitly including the inertial force of the structure in the IB method when calculating the hydrodynamic stress on the solid surface may easily destabilize the simulation. To address this issue, we have incorporated a penalty IB method in the LB method [17]. In this method, the plate itself is assumed to be massless, but a ghost plate of density m_p is attached to the physical plate through virtual springs of stiffness K_v (see figure 2). The ghost plate only affects the dynamics of the physical plate but is not seen by the flow solver directly. Thus, the density of the fluid is a constant ρ, and the Lagrangian force is modified to incorporate the ghost plate,

$$\mathbf{F} = \mathbf{F}_k + \mathbf{F}_e,\ \mathbf{F}_k = K_v[\mathbf{Y}-\mathbf{X}],\ m_p\frac{\partial^2 \mathbf{Y}}{\partial t^2} = -\mathbf{F}_k, \qquad (10)$$

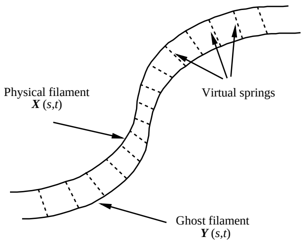

Figure 2: The physical and ghost filaments are tethered together by virtual springs [17].

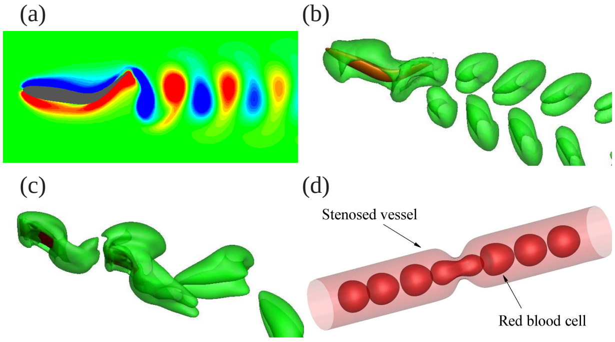

Figure 3: Applications of the IB-LB method: (a) 2D NACA0012 foil, (b) 3D elliptical foil, (c) two flags in tandem arrangement, and (d) a 3D red blood cell in a stenosed vessel [3].

where $\mathbf{F}_k$ is the spring force, $\mathbf{F}_e$ is the elastic force given by $\mathbf{F}_e = \sum_{i,j=1}^{n_d}\left[\frac{\partial}{\partial s_i}\left(\sigma_{ij}\frac{\partial \mathbf{X}}{\partial s_j}\right) - \frac{\partial^2}{\partial s_i \partial s_j}\left(\gamma_{ij}\frac{\partial^2 \mathbf{X}}{\partial s_i \partial s_j}\right)\right]$, K_v is the stiffness of virtual springs, and $\mathbf{Y}$ is the position vector of the point on the ghost plate connecting to point $\mathbf{X}$ on the physical plate. Essentially, the effect of the inertia of the ghost plate is cushioned through the virtual springs. This method has been used to simulate hydrodynamic interaction between the flexible plates/capsules and the incompressible viscous flow [13, 17–19].

The Lagrangian velocity, position of the physical plate, and body force can be discretized as follows,

$$\mathbf{U}^{n+1} = \sum_{\mathbf{x}} \mathbf{u}^{n+1}(\mathbf{x},t)\delta_h(\mathbf{x}-\mathbf{X}^n)\Delta V,\ \mathbf{X}^{n+1} = \mathbf{X}^n + \mathbf{U}^{n+1}\Delta t, \qquad (11)$$

$$\mathbf{f}^{n+1} = \sum_{\mathbf{s}} \mathbf{F}^{n+1}(\mathbf{s},t)\delta_h(\mathbf{x}-\mathbf{X}^n)\Delta A. \qquad (12)$$

In Eq. (12), $\Delta A = \Delta s$ and $\Delta s_1 \Delta s_2$ for the 1D and 2D plates, respectively, while in Eq. (11), $\Delta V = \Delta x \Delta y$ and $\Delta V = \Delta x \Delta y \Delta z$ for the 2D and 3D simulations, respectively. The notations $\sum_{\mathbf{s}}$ and $\sum_{\mathbf{x}}$ mean the sum over all the discrete points of $\mathbf{X}$ and the sum over all the discrete points of $\mathbf{x}$, respectively. δ_h is a smooth approximation of Dirac's delta function [9, 10]. This version of IB-LB method has been used in fish swimming, flag flapping and red blood cell [3, 13, 17–19, 22, 23], as shown in figure 3.

Sharp-interface immersed boundary–finite element method

In this section, we introduce the sharp-interface IB–finite element (FE) method for 3D FSI involving large deformations. The IB method for fluid solver was previously developed by

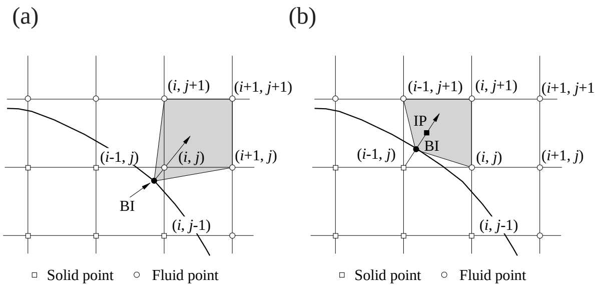

Figure 4: 2D schematics illustrating the sharp-interface immersed boundary method: (a) interpolation stencil, and (b) extrapolation stencil [3].

Mittal et al. [7] and later improved by Luo et al. [6] and Tian et al. [14, 15]. This method retains the sharp-interface representation of the fluid–solid interface and employs local flow reconstruction to facilitate the finite difference discretization near the immersed boundary. The 2D schematics of this method is shown in figure 4. When the second-order central difference scheme is applied to discretize the Navier–Stokes equations in the fluid region near the fluid-solid interface, incomplete stencils could be encountered. Specifically, on the node (i,j) in figure 4(a), the finite difference stencil will involve nodes $(i-1,j)$ and $(i,j-1)$ that are located inside the solid body. Here we introduce two methods to update the variables on node (i,j). In the first method, the variables on (i,j) are interpolated by using the interpolation stencil shown in figure 4(a). A body intercept (BI) point can be found by projecting the (i,j) onto the boundary along the surface normal. The variable, φ, in the local area around (i,j) is approximated by $\varphi = a_1 xy + a_2 x + a_3 y + a_4$, where a_1, a_2, a_3 and a_4 can be determined by using the values on BI, together with (i+1,j), $(i, j+1)$, and $(i+1, j+1)$. Then the value on (i,j) can be obtained by $\varphi_{i,j} = \sum_{m=1}^{4} \beta_m \varphi_m$ where φ_m is one of the 4 data points. In the second method, we first apply the extrapolation, and then use the finite difference method. As shown in figure 4(b), to calculate the value on (i,j), the values on $(i-1,j)$ and $(i,j-1)$ are first extrapolated, and those on (i,j) are then calculated by using the finite difference method. Take node $(i-1,j)$ as an example, the BI point can be determined by the same way as the interpolation. The image point (IP) can be found by taking the symmetrical point about the boundary. The value on IP can be determined by using the shaded stencil, i.e. the values at previous time step on BI, (i,j), $(i,j+1)$, and $(i-1,j+1)$. Then $\varphi^n_{i-1,j} = 2\varphi^n_{BI} - \varphi^n_{IP}$. Therefore, $\varphi^{n+1}_{i,j}$ can be updated by the finite difference method. Iteration is required in the cases where the points used to interpolate the unknown values are in the solid region or immediately next to the interface. In the practice, the numerical oscillations in the moving boundary problems associated with the sudden change of the stencils can be effectively reduced by applying the hybrid scheme of these two methods [6, 14, 15].

The FE formulation in the structure solver is derived from the standard virtual work method. Let the displacement in a volume element be represented by $\mathbf{u}(X,Y,Z) = \sum_k h_k(X,Y,Z)\mathbf{u}_k = [H]\{u\}$, where $h_k(X,Y,Z)$ is the shape function associated with the kth node in the element and $\mathbf{u}_k$ is the displacement $\mathbf{u}$ at this node. Other variables can be expanded in a similar manner.

Using the virtual work of the inertial load, body force $\{b\}$, and surface traction $\{f\}$ along with the expansion of the variables, the assembled equation system for the entire body can be written as

$$[M]\{\ddot{u}\} + [C]\{\dot{u}\} = \{P\} - \{F\}, \tag{13}$$

where $[M]$ is the mass matrix, $[C]$ is the mass-damping matrix, $\{P\}$ is the force vector from the external load, and $\{F\}$ is the body stress vector. These assembled terms can be found in Refs. [14, 15]. For general 3D bodies, hexahedral (or brick-type) quadratic 20-node elements [1] are used in the FE formulation. The FE formulation of the thin-walled structures includes the three-node plate elements and two-node frame elements, where each node has six global degrees of freedom, including three displacement components, u_i, and three angles of rotation, ϕ_i. The discrete equations can be written in the same forms as in the general structure form, except that $\{u\}$ represents the generalized displacement vector with ϕ_i included and $[\sigma^K]$ represents the generalized stress with moments included. The large-displacement and small-strain deformation in the structure solver is handled using the corotational scheme. The time stepping is achieved using a case of Newmark scheme [4].

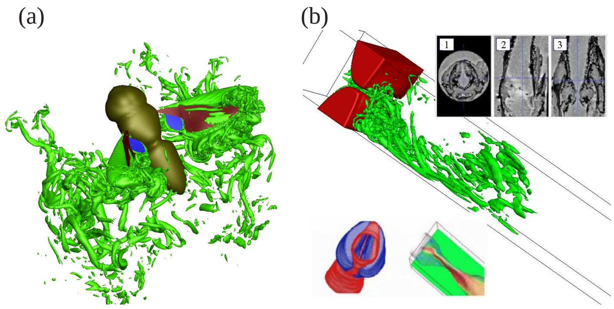

Figure 5: Applications of the sharp-interface IB–FE method: (a) vortical structures around a hoverfly and (b) modelling of vocal-fold vibration [15, 20].

The fluid–structure coupling is done by iterating the two solvers by exchanging the boundary information until convergence is reached. The residuals as measured by the maximum errors of the displacement, the velocity, and the traction at the solid surface are used to determine whether final convergence is reached. To ensure the numerical stability of this staggered iteration, the velocity of the solid surface are updated in the flow solver in a gradual fashion according to $\mathbf{v}_b^{(k+1)} = \alpha \mathbf{v}_b^p + (1-\alpha)\mathbf{v}_b^{(k)}$, where $\mathbf{v}_b^p$ is the predicted velocity by the structure solver and α is the relaxation factor between 0 and 1. The displacement $\mathbf{X}_b$ in the flow solver and the traction $\mathbf{F}_f$ in the structure solver can be updated in a similar manner if necessary.

This method has been used in insect flight, and FSI during phonation, as shown in figure 5.

Immersed boundary method for heat transfer and electrodynamics

Heat transfer can be described by $\frac{\partial T}{\partial t} + \mathbf{u}\cdot\nabla T = k\nabla^2 T$. To account for the effect of the boundary, a heat source is applied in the convection-diffusion equation, $\frac{\partial T}{\partial t} + \mathbf{u}\cdot\nabla T = k\nabla^2 T + q$, where q is the heat source which is obtained by spreading Lagrangian heat source Q to the points near the boundary. For the Dirichlet boundary condition, Q at $n+1$ step is calculated by

$$Q^{n+1} = (T_B - T^n)/\Delta t + \mathbf{u}^n \cdot \nabla T^n - k\nabla^2 T^n, \tag{14}$$

where T_B is the boundary temperature. For the Neumann boundary condition, Q is determined by

$$Q^{n+1} = 2(Q_B + k\frac{\partial T^n}{\partial \mathbf{n}}). \tag{15}$$

In the electrodynamics applications, the Poisson equation, $\nabla^2\Phi = -q$, is generally used to describe the electric potential field Φ. This equation can be rewritten as $\partial\Phi/\partial\tau = k_e(\nabla^2\Phi + q)$ where τ is the pseudo time. Then similar treatment as in heat transfer can be used.

Conclusions

The IB method based on LB method and FE method has been briefly introduced. The applications in insect flight, fish swimming, red blood cells, FSI during phonation, heat transfer and electrodynamics have been demonstrated. It shows that the IB method is effective in modelling complex flows, fluid-structure interactions and convection-diffusion processes.

Acknowledgements

This research was partly supported by the Australian Research Council's Discovery Project Funding Scheme (project number DP130103850).

References

[1] Bathe, K. J., *Finite Element Procedures*, Prentice-Hall, Englewood Cliffs, NJ, 1996.

[2] Chen, S. and Doolen, G. D., Lattice Boltzmann method for fluid flows, *Annu. Rev. Fluid Mech.*, **30**, 1998, 329–364.

[3] Deng, H. B., Xu, Y. Q., Chen, D. D., Dai, H., Wu, J. and Tian, F. B., On numerical modeling of animal swimming and flight, *Comput. Mech.*, **52**, 2013, 1221–1242.

[4] Doyle, J. F., *Nonlinear Analysis of Thin-walled Structures: Statics, Dynamics, and Stability*, Springer–Verlag, New York, 2001.

[5] Huang, W. X. and Sung, H. J., Three-dimensional simulation of a flapping flag in a uniform flow, *J. Fluid Mech.*, **653**, 2010, 301–336.

[6] Luo, H., Dai, H., Ferreira de Sousa, P. and Yin, B., On the numerical oscillation of the direct-forcing immersed-boundary method for moving boundaries, *Comput. Fluids*, **56**, 2012, 61–76.

[7] Mittal, R., Dong, H., Bozkurttas, M., Najjar, F. M., Vargas, A. and von Loebbecke, A., A versatile sharp interface immersed boundary method for incompressible flows with complex boundaries, *J. Comput. Phys.*, **227**, 2008, 4825–4852.

[8] Peskin, C. S., Numerical analysis of blood flow in the heart, *J. Comput. Phys.*, **25**, 1977, 220–252.

[9] Peskin, C. S., The immersed boundary method, *Acta Numerica*, **11**, 2002, 479–517.

[10] Peskin, C. S. and McQueen., D. M., Fluid dynamics of the heart and its valves, in *Case Studies in Mathematical Modeling: Ecology, Physiology, and Cell Biology*, editors H. G. Othmer, F. R. Adler, M. A. Lewis and J. C. Dallon, Prentice-Hall, Englewood, New Jersey, 1996, 309–337, 309–337.

[11] Qian, Y. H., D'Humières, D. and Lallemand, P., Lattice BGK models for Navier–Stokes equation, *Europhys. Lett.*, **17**, 1992, 479–484.

[12] Taira, K. and Colonius, T., The immersed boundary method: a projection approach, *J. Comput. Phys.*, **225**, 2007, 2118–2137.

[13] Tian, F. B., Role of mass on the stability of flag/flags in uniform flow, *Appl. Phys. Lett.*, **103**, 2013, 034101.

[14] Tian, F. B., Dai, H., Luo, H., Doyle, J. F. and Rousseau, B., Computational fluid–structure interaction for biological and biomedical flows, in *Proceedings of the ASME 2013 Fluids Engineering Division Summer Meeting*, Incline Village, Nevada, 2013, 16408, 16408.

[15] Tian, F. B., Dai, H., Luo, H., Doyle, J. F. and Rousseau, B., Fluid–structure interaction involving large deformations: 3D simulations and applications to biological systems, *J. Comput. Phys.*, **258**, 2014, 451–469.

[16] Tian, F. B., Lu, X. Y. and Luo, H., Onset of instability of a flag in uniform flow, *Theor. Appl. Mech. Lett.*, **2**, 2012, 022005.

[17] Tian, F. B., Luo, H., Zhu, L., Liao, J. C. and Lu, X. Y., An immersed boundary-lattice Boltzmann method for elastic boundaries with mass, *J. Comput. Phys.*, **230**, 2011, 7266–7283.

[18] Tian, F. B., Luo, H., Zhu, L. and Lu, X. Y., Interaction between a flexible filament and a downstream rigid body, *Phys. Rev. E*, **82**, 2010, 026301.

[19] Tian, F. B., Luo, H., Zhu, L. and Lu, X. Y., Coupling modes of three filaments in side-by-side arrangement, *Phys. Fluids*, **23**, 2011, 111903.

[20] Tian, F. B., Young, J. and Lai, J. C. S., Aerodynamic characteristics of hoverflies during hovering flight, in *The Eighth International Conference on Computational Fluid Dynamics*, Chengdu, China, 2014.

[21] Tseng, Y. H. and Ferziger, J. H., A ghost-cell immersed boundary method for flow in complex geometry, *J. Comput. Phys.*, **192**, 2003, 593–623.

[22] Xu, Y. Q., Tang, X. Y., Tian, F. B., Peng, Y. H., Xu, Y. and Zeng, Y. J., IB–LBM simulation of the haemocyte dynamics in a stenotic capillary, *Comput. Method Biomec.*, **17**, 2014, 978–985.

[23] Xu, Y. Q., Tian, F. B. and Deng, Y. L., An efficient red blood cell model in the frame of IB–LBM and its application, *Int. J. Biomath.*, **6**, 2013, 1250061.

[24] Yang, J. M. and Balaras, E., An embedded-boundary formulation for large-eddy simulation of turbulent flows interacting with moving boundaries, *J. Comput. Phys.*, **215**, 2006, 12–40.

[25] Zhu, L. and Peskin, C. S., Simulation of a flapping flexible filament in a flowing soap film by the immersed boundary method, *J. Comput. Phys.*, **179**, 2002, 452–468.