mathematics

MDPI

Review

Immersed Boundary Method for Simulating Interfacial Problems

Wanho Lee [1] and Seunggyu Lee [2,*]

[1] National Institute for Mathematical Sciences, Daejeon 34047, Korea; wlee@nims.re.kr

[2] Department of Mathematics and Research Institute of Natural Science, Gyeongsang National University, Jinju 52828, Korea

[*] Correspondence: sglee@gnu.ac.kr

Received: 8 October 2020; Accepted: 4 November 2020; Published: 6 November 2020

Abstract: We review the immersed boundary (IB) method in order to investigate the fluid-structure interaction problems governed by the Navier–Stokes equation. The configuration is described by the Lagrangian variables, and the velocity and pressure of the fluid are defined in Cartesian coordinates. The interaction between two different coordinates is involved in a discrete Dirac-delta function. We describe the IB method and its numerical implementation. Standard numerical simulations are performed in order to show the effect of the parameters and discrete Dirac-delta functions. Simulations of flow around a cylinder and movement of Caenorhabditis elegans are introduced as rigid and flexible boundary problems, respectively. Furthermore, we provide the MATLAB codes for our simulation.

Keywords: immersed boundary; interfacial problem; fluid-structure interaction; discrete Dirac-delta function

1. Introduction

Fluid–structure interaction is widely observed in various fields [1–5]. However, simulating fluid-structure interaction is a challenge problem in mathematics, because of its boundary layer transition. The immersed boundary (IB) method was originally proposed by Peskin in 1977 in order to model flow patterns around heart valves [6,7]. It is widely used for simulating fluid-structure interactions, such as turbulent flow [8,9], biomembrane [10,11], glioma invasion [12], and swimming eel [13]. In the IB method, the Eulerian variables are defined based on a fixed Cartesian coordinate system, and the Lagrangian variables move freely through the fixed Cartesian coordinates. Their interaction is expressed by a discrete Dirac-delta function. The advantages of the IB method are its simplicity and efficiency in handling complicated geometries without mesh regeneration.

In the original IB method, the massless boundary moves with the velocity of the ambient flow by the fluid force, and the density of the structure is spread out to the ambient flow in order to consider the inertia of the elastic structure [6]. The movement of the elastic boundary with mass can be easily simulated by the extension of the original IB method by Kim and Peskin, which is called the penalty IB method [14]. The penalty IB method has recently been extended to simulate the elastic boundary with mass [14] and the interaction between a rigid body and a surrounding fluid [15]. The numerical simulation with high Reynolds numbers is strictly limited when using the conventional IB method, because it is needed to capture the flow transition in a thin boundary layer at high Reynolds numbers. It can be resolved by refining the mesh grid locally [16,17] or imposing a velocity profile based on a wall model [18,19] at the cutting egde.

This paper briefly reviews the IB method and its numerical implementation. The derivation of the equation of motion for the IB method is quite intuitive; however, some issues remain regarding its implementation for beginners, such as the choice of delta functions and the effect of parameters. To resolve this, the actual program code is provided with the corresponding numerical simulations. Moreover, a practical example is introduced in order to explain the modeling of biological phenomena using the IB method.

The remainder of this paper is organized, as follows. We describe the mathematical formulations for the IB method in Section 2. Section 3 presents the discretization of the IB method. The numerical results are presented to show the effect of the Reynolds number and discrete Dirac-delta function in Section 4. Finally, Section 5 presents the conclusion. Furthermore, we provide the MATLAB code for the numerical implementation from the corresponding author's webpage and its structure can be found in Appendix A.

2. Mathematical Formulations

Let $\mathbf{X}(s,t) = (X(s,t), Y(s,y))$ be the position of the Lagrangian points with label s, describing the configuration in a two-dimensional space at particular time t. If we assume that the evolution of the configuration is determined by the (elastic) energy according to the functional $E[\mathbf{X}]$, the functional is expressed in the following form:

$$\wp E[\mathbf{X}(\cdot,t)] = \int (-\mathbf{F}(s,t)) \cdot \wp \mathbf{X}(s,t) ds, \tag{1}$$

where $\wp$ is a variation symbol that distinguishes the symbol from the Dirac-delta function δ. Taking the Fréchet derivative of E at $\mathbf{X}(s,t)$, we can define the force density that is generated by the material $\mathbf{F}(s,t)$ as follows:

$$\mathbf{F}(s,t) = -\frac{\wp E[\mathbf{X}(\cdot,t)]}{\wp \mathbf{X}(s,t)}. \tag{2}$$

The elastic energy functional is given by

$$E[\mathbf{X}(\cdot,t)] = \frac{\sigma}{2} \int \left(\left| \frac{\partial \mathbf{X}}{\partial s} \right| - 1 \right)^2 ds, \tag{3}$$

where σ is the stiffness coefficient and $|\cdot|$ is the L^2-norm. Subsequently, we obtain

$$\wp E[\mathbf{X}(\cdot,t)] = -\sigma \int \frac{\partial}{\partial s} \left[\left(\left| \frac{\partial \mathbf{X}}{\partial s} \right| - 1 \right) \frac{\partial \mathbf{X}/\partial s}{|\partial \mathbf{X}/\partial s|} \right] \cdot \wp \mathbf{X} ds. \tag{4}$$

Thus, the elastic force density can be rewritten as

$$\mathbf{F}(s,t) = \frac{\partial}{\partial s} [T(s,t) \boldsymbol{\tau}(s,t)], \tag{5}$$

where

$$T(s,t) = \sigma \left(\left| \frac{\partial \mathbf{X}}{\partial s} \right| - 1 \right), \tag{6}$$

is the tension of the configuration and

$$\boldsymbol{\tau}(s,t) = \frac{\partial \mathbf{X}/\partial s}{|\partial \mathbf{X}/\partial s|}, \tag{7}$$

is the unit tangent vector to the configuration. This is the only formulation for the elastic force, and it can be modified by adding other energy functional to Equation (3) in order to consider the corresponding force.

Let $\mathbf{u} = (u(\mathbf{x},t), v(\mathbf{x},t))$ be the velocity at the position $\mathbf{x}$ and time t. Subsequently, we can define the velocity of the configuration $\mathbf{U}(s,t)$, as follows:

$$\mathbf{U}(s,t) = \frac{\partial \mathbf{X}(s,t)}{\partial t} = \mathbf{u}(\mathbf{X}(s,t),t) = \int \mathbf{u}(\mathbf{x},t)\delta(\mathbf{x} - \mathbf{X}(s,t))\, d\mathbf{x}\,, \tag{8}$$

where $\delta(\mathbf{x})$ denotes the two-dimensional Dirac-delta function $\delta(x)\delta(y)$. Note that the material derivative of $\mathbf{u}$ is defined as

$$\frac{D\mathbf{u}}{Dt} = \frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u}, \tag{9}$$

and it implies that

$$\frac{D\mathbf{u}}{Dt}(\mathbf{X}(s,t),t) = \frac{\partial^2 \mathbf{X}(s,t)}{\partial t^2}. \tag{10}$$

Conversely, the system would naturally evolve to minimize the time integration of the difference between the kinetic and potential energy functionals:

$$\begin{aligned}
0 &= \wp \left[\int \left(\frac{1}{2} \int M(s) |\mathbf{U}(s,t)|^2 ds - E(\mathbf{X}(\cdot,t)) \right) dt \right] \\
&= \int \int \left(M(s) \frac{\partial^2 \mathbf{X}(s,t)}{\partial t^2} - \mathbf{F}(s) \right) \cdot \wp \mathbf{X}(s) ds dt \\
&= \int \int \left(M(s) \int \frac{D\mathbf{u}}{Dt}(\mathbf{x},t) \delta(\mathbf{x} - \mathbf{X}(s,t))\, d\mathbf{x} - \mathbf{F}(s) \right) \cdot \wp \mathbf{X}(s) ds dt \\
&= \int \int \int \left(M(s) \frac{D\mathbf{u}}{Dt}(\mathbf{x},t) - \mathbf{F}(s) \right) \cdot \mathbf{v}(\mathbf{x},t) \delta(\mathbf{x} - \mathbf{X}(s,t))\, d\mathbf{x}\, ds dt,
\end{aligned} \tag{11}$$

where $\mathbf{v}(\mathbf{x},t)$ is the variational derivative of the configuration at position $\mathbf{x}$ and time t, which implies $\mathbf{v}(\mathbf{X}(s),t) = \wp \mathbf{X}(s,t)$. Because the expression contains both Lagrangian and Eulerian variables, we should eliminate the Lagrangian variables by defining the fluid mass density $\rho(\mathbf{x},t)$ and fluid force density $\mathbf{f}(\mathbf{x},t)$, as follows:

$$\rho(\mathbf{x},t) = \int M(s) \delta(\mathbf{x} - \mathbf{X}(s,t)) ds, \tag{12}$$

$$\mathbf{f}(\mathbf{x},t) = \int \mathbf{F}(s,t) \delta(\mathbf{x} - \mathbf{X}(s,t)) ds. \tag{13}$$

For simplicity, we assume a constant mass density, which is, $M(s) \equiv M$ and $\rho(\mathbf{x},t) \equiv \rho$.

The equations of fluid motion are governed by the Navier–Stokes (NS) equation, as follows:

$$\rho \left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} \right) = -\nabla p + \mu \Delta \mathbf{u} + \mathbf{f} \ \text{ in } \Omega \times (0, \infty), \tag{14}$$

$$\nabla \cdot \mathbf{u} = 0, \tag{15}$$

where $p(\mathbf{x},t)$ is the pressure, μ is the viscosity, and $\Omega = (0,L)^2$ is the square domain. Let $Re = \rho U^* L^* / \mu$ and $We = \rho (U^*)^2 L^* / \sigma$ be the Reynolds and Weber numbers with the characteristic velocity U^* and length L^*, respectively. Subsequently, the non-dimensionalized version of Equations (14) and (15) can be expressed, as follows:

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = -\nabla p + \frac{1}{Re} \Delta \mathbf{u} + \frac{1}{We} \mathbf{f}, \tag{16}$$

$$\nabla \cdot \mathbf{u} = 0. \tag{17}$$

3. Discretization

Various numerical schemes have been developed to solve the NS equation. In this study, we employ the implicit scheme, as follows [20]:

$$\frac{\mathbf{u}^{n+1}-\mathbf{u}^{n}}{\Delta t}+\mathbf{u}^{n}\cdot\nabla_{d}\mathbf{u}^{n}=-\nabla_{h}p^{n+1}+\frac{1}{Re}\Delta_{h}\mathbf{u}^{n+1}+\frac{1}{We}\mathbf{f}^{n}, \tag{18}$$

$$\nabla_{h}\cdot\mathbf{u}^{n+1}=0. \tag{19}$$

Here, ∇_h is the discrete gradient operator using the central difference and Δ_h is the discrete Laplacian operator defined as

$$\nabla_{h}\phi(\mathbf{x}):=(D_{x}\phi(\mathbf{x}),D_{y}\phi(\mathbf{y}))=\left(\frac{\phi(x+h,y)-\phi(x-h,y)}{2h},\frac{\phi(x,y+h)-\phi(x,y-h)}{2h}\right), \tag{20}$$

$$\Delta_{h}\phi(\mathbf{x})=\frac{\phi(x+h,y)+\phi(x,y+h)-4\phi(x,y)+\phi(x-h,y)+\phi(x,y-h)}{h^{2}}. \tag{21}$$

For the convection term $\mathbf{u}^{n}\cdot\nabla_{d}\mathbf{u}^{n}=(uD_{x}^{\pm}u+vD_{y}^{\pm}u,uD_{x}^{\pm}v+vD_{y}^{\pm}v)$, we use the skew symmetry, as follows:

$$(uD_{x}^{\pm})(u(\mathbf{x}))=\frac{(u(x+h,y)+u(x,y))u(x+h,y)-(u(x-h,y)+u(x,y))u(x-h,y)}{4h}. \tag{22}$$

Other terms are computed in a similar manner.

Applying the projection method [21], we split Equation (18) into two equations with the temporal velocity $\tilde{\mathbf{u}}$, as follows:

$$\frac{\tilde{\mathbf{u}}-\mathbf{u}^{n}}{\Delta t}=-\mathbf{u}^{n}\cdot\nabla_{d}\mathbf{u}^{n}+\frac{1}{We}\mathbf{f}^{n}, \tag{23}$$

$$\frac{\mathbf{u}^{n+1}-\tilde{\mathbf{u}}}{\Delta t}=-\nabla_{h}p^{n+1}+\frac{1}{Re}\Delta_{h}\mathbf{u}^{n+1}. \tag{24}$$

Taking the divergence operator into Equation (24), the Poisson equation for the pressure is derived by Equation (19), as follows:

$$\Delta_{h}p^{n+1}=\frac{1}{\Delta t}\nabla_{d}\cdot\tilde{\mathbf{u}}. \tag{25}$$

Equation (23) can be solved directly, because it has an explicit formulation, and the fast Fourier transformation is used to solve Equations (24) and (25). The details can be found in [20]. Here, the fluid forcing term $\mathbf{f}^n$ is given by

$$\mathbf{f}^{n}(\mathbf{x})=\sum_{s\in G_{h}}\mathbf{F}^{n}(s)\delta_{h}(\mathbf{x}-\mathbf{X}^{n}(s))\Delta s, \tag{26}$$

where G_h is the set of indices for the Lagrangian points, δ_h is the discrete Dirac-delta function, which was discussed in the later section, and Δs is the step size of the label of the configuration. Here, the discrete force density for the structure $\mathbf{F}^n(s)$ is defined by

$$\mathbf{F}^{n}(s)=D_{s}\left(T^{n}(s)\boldsymbol{\tau}(s)\right), \tag{27}$$

$$T^{n}(s)=\sigma\left(|D_{s}\mathbf{X}^{n}(s)|-1\right), \tag{28}$$

$$\boldsymbol{\tau}^{n}(s)=\frac{D_{s}\mathbf{X}^{n}(s)}{|D_{s}\mathbf{X}^{n}(s)|}, \tag{29}$$

$$D_s\mathbf{X}^n(s) = \frac{(X^n(s+0.5\Delta s) - X^n(s-0.5\Delta s), Y^n(s+0.5\Delta s) - Y^n(s-0.5\Delta s))}{\Delta s}, \tag{30}$$

where D_s is the discrete differential operator with respect to s. Finally, the discretization of the transport equation for IB is given, as follows:

$$\mathbf{U}^n(s) = \frac{\mathbf{X}^{n+1}(s) - \mathbf{X}^n(s)}{\Delta t} = \sum_{\mathbf{x}\in g_h} \mathbf{u}^n(\mathbf{x})\delta_h(\mathbf{x} - \mathbf{X}^n(s))h^2, \tag{31}$$

where g_h is the set of Eulerian points.

4. Numerical Results

4.1. Code and Simulation

In this section, we discuss the MATLAB code of the IB method for solving a moving boundary problem with a simple and basic example. The code updates the fluid velocity $\mathbf{u}^{n+1}$ and boundary position $\mathbf{X}^{n+1}$ from the given values $\mathbf{u}^n$ and $\mathbf{X}^n$.

To explain how to obtain the updated boundary position, we first set a domain of fixed Cartesian coordinates for a fluid and let the IB structure on the fixed domain. Here, the following ellipse is chosen for the initial condition of the boundary:

$$\left(\frac{x - 0.5L}{0.25L}\right)^2 + \left(\frac{y - 0.5L}{0.125L}\right)^2 = 1,$$

where L is the length of the domain.

```
% initial boundary configuration
ip=[(2:N),1]; im=[N,(1:(N-1))];
for k=0:10000
    theta=k*2*pi/10000;
    Temp(k+1,1)=(L/2)+L/4*cos(theta);
    Temp(k+1,2)=(L/2)+L/8*sin(theta);
end
length(1)=0; Nb=1; Leng=0;
X(1,1)=Temp(1,1); X(1,2)=Temp(1,2);
for k=1:10000
    length(k+1)=length(k)+sqrt((Temp(k+1,1)-Temp(k,1))^2+(Temp(k+1,2)-Temp(k,2))^2);
    if ((length(k+1)>(Nb*ds))&&(length(k)<(Nb*ds)))
        Nb=Nb+1;
        X(Nb,1)=Temp(k,1); X(Nb,2)=Temp(k,2);
        Leng=Leng+sqrt((X(Nb,1)-X(Nb-1,1))^2+(X(Nb,2)-X(Nb-1,2))^2);
    end
end
kp=[(2:Nb),1]; km=[Nb,(1:(Nb-1))];

% fluid setting
u=zeros(N,N,2); p=zeros(N,N); xgrid=zeros(N,N); ygrid=zeros(N,N);
for j=0:(N-1)
    xgrid(j+1,:)=j*h; ygrid(:,j+1)=j*h;
end
```

Next, the elastic boundary force density is evaluated from the IB structure. It can be obtained from a minus gradient of the elastic energy functional with respect to $\mathbf{X}$. An elastic energy function can be classified based on three types of contributions: stretching, bending, and tether energies. In the example, the force is only generated from the stretching energy with a zero rest length.

```
%% generate the boundary force dencity
F=(X(kp,:)+X(km,:)-2*X)/(ds*ds);
```

Using the evaluated boundary force density in the previous step, we spread out the boundary force density to a fluid force density while using the Dirac-delta function. This is one of the key steps for the fluid-structure interaction in the IB method. There are some choices of the Dirac-delta function, and its details will be discussed in the next section.

```
function f=spread(F,X)
global h N ds Nb
c=ds/(h*h);
f=zeros(N,N,2);
for k=1:Nb
    s=X(k,:)/h; i=floor(s); r=s-i;
    i1=mod((i(1)-1):(i(1)+2),N)+1;
    i2=mod((i(2)-1):(i(2)+2),N)+1;
    w=phi1(r(1)).*phi2(r(2));
    f(i1,i2,1)=f(i1,i2,1)+(c*F(k,1))*w;
    f(i1,i2,2)=f(i1,i2,2)+(c*F(k,2))*w;
end
```

The next step is to update the fluid velocity by solving the NS equation while using the spread fluid force density on the fluid (fixed) domain. Subsequently, the Dirac-delta function is used again to obtain the velocity on the elastic IB boundary.

```
function U=interp(u,X)
global Nb h N;
U=zeros(Nb,2);
for k=1:Nb
    s=X(k,:)/h; i=floor(s); r=s-i;
    i1=mod((i(1)-1):(i(1)+2),N)+1;
    i2=mod((i(2)-1):(i(2)+2),N)+1;
    w=phi1(r(1)).*phi2(r(2));
    U(k,1)=sum(sum(w.*u(i1,i2,1)));
    U(k,2)=sum(sum(w.*u(i1,i2,2)));
end
```

Finally, we can obtain the updated boundary position by solving the transport equation. See the Appendix for a detailed code of the elastic membrane relaxation in viscous fluids.

```
%% move the boundary at the local fluid velocity
X=X+dt*interp(u,X);
```

Figure 1 shows the vorticity and pressure contours around the damped vibration elastic body after code execution. Note that the vorticity is a physical quantity that represents a local circulation in the fluid and it is defined as a flow rate rotation, and the pressure difference is derived from the boundary force density. If the reader uses the code to simulate other problems, then it is important to modify the initial boundary and force density generation. We will introduce the simulation of the rigid and flexible boundary problems in Sections 4.3 and 4.4, respectively.

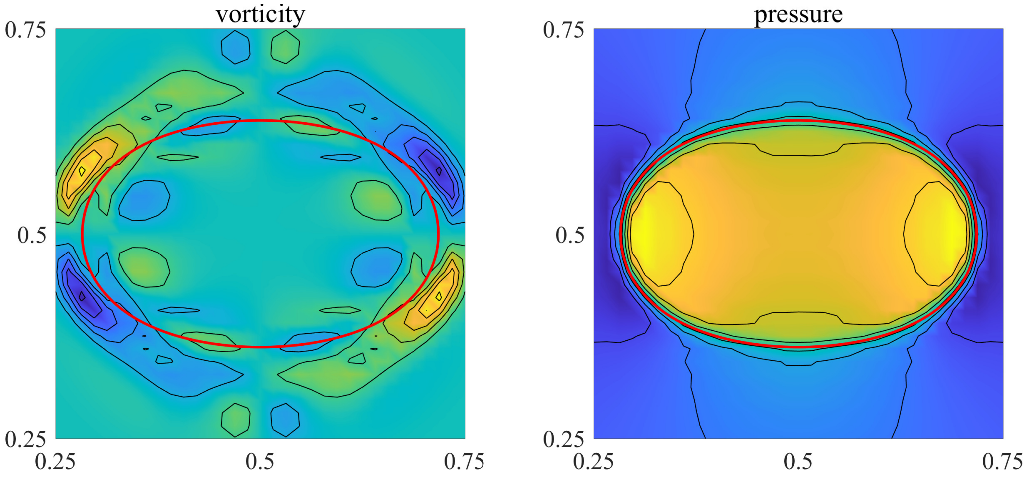


Figure 1. Vorticity (**left**) and pressure (**right**) contours around the damped vibration elastic body.

4.2. Discrete Delta Function

The Dirac-delta function, $\delta(x)$, plays an important role in the interaction of fluids and boundaries in the IB method. In Equations (8), (12), and (13) $\delta(x)$ are used for interpolation or spread out of the values between fluids and boundaries. The function $\delta(x)$ assumes the form $\delta(x) = \frac{1}{h}\phi(\frac{x}{h})$, where $\phi(r)$ satisfies the following properties [22]:

1. $\phi(r)$ is continuous,
2. $\phi(r) = 0$ for $|r| \geq 2$,
3. $\Sigma_{even}\phi(r-i) = \Sigma_{odd}\phi(r-i) = \frac{1}{2}$ for all r,
4. $\Sigma[\phi(r-i)]^2 = \textit{constant}$ for all r, and
5. $\Sigma(r-i)\phi(r-i) = 0$ for all r.

There are some common choices for the discrete delta functions or regular functions ϕ_i, such as 2-point, 4-point, 4-point cosine, and 6-point functions. Figure 2 shows the profile of the various regular functions, $\phi_i, i \in \{2, 4c, 4, 6\}$ that form the delta functions, δ_i. Among the choices, the only regular function ϕ_4, which satisfies all of the above five conditions, is defined, as follows:

$$\phi_4 = \begin{cases} \frac{1}{8}\left(3 - 2|r| + \sqrt{1 + 4|r| - 4r^2}\right) & |r| \leq 1, \\ \frac{1}{8}\left(5 - 2|r| - \sqrt{-7 + 12|r| - 4r^2}\right) & 1 \leq |r| \leq 2, \\ 0 & 2 \leq |r|. \end{cases} \tag{32}$$

Other regular functions can be obtained by changing the support size or adding/deleting conditions. For example, the four-point cosine approximation does not satisfy the second moment condition (5th condition in the list). The detailed formulas mentioned above can be found in [23].

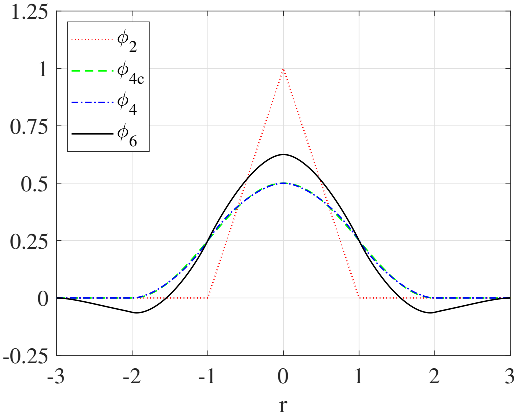


Figure 2. Various regular functions ϕ_i that form the delta functions δ_i.

We constructed a dumbbell-shaped boundary, as shown in the upper left panel of Figure 3, to investigate how the selection of the delta function affects the simulation. The distance between boundaries is equal to $h/4$ at $X = 2$. The initial boundary comprises a single closed curve and it is given a stretching force. Thus, with time, the boundary must change to a circular shape.

Figure 3 shows the simulation results of the selection of the delta function in the IB method. When using δ_{4c}, it is the most affected by close distance, and when using δ_2, it is the least affected. The delta function δ_6 has wider support than δ_{4c}. However, the actual impact thickness shows that function δ_6 is narrower than function δ_{4c} from the simulation results.

Figure 4 shows the area within the boundary over time in each case that is covered in Figure 3. In general, the traditional IB method has a disadvantage of causing volume loss within boundaries. The use of the δ_2 function in this simulation result appears to be the most vulnerable to volume retention within boundaries.

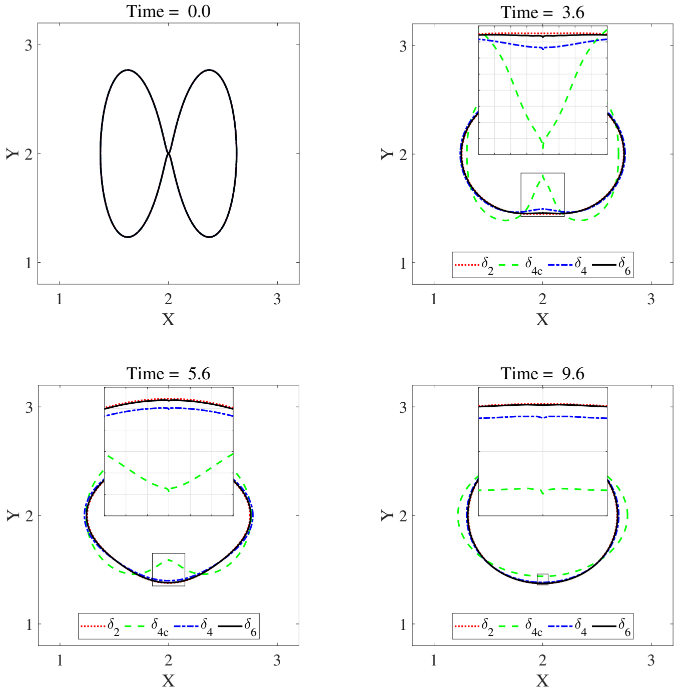


Figure 3. Simulation results based on selection of the delta function.

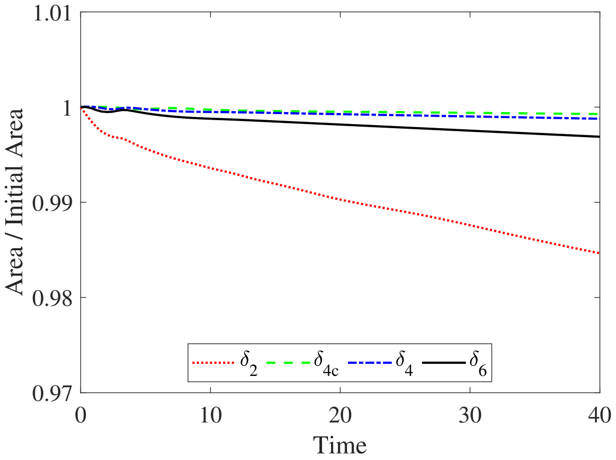


Figure 4. The area within the boundary over time in each case covered in Figure 3.

4.3. Effect of the Reynolds Number

Next, we apply the IB method for modeling flows around fixed rigid bodies. A simple approach for implementing a fixed rigid body using the traditional IB method is to place the target point($\mathbf{X}_T$) at the desired location and attach the boundary($\mathbf{X}$) whlie using a rigid spring. Here, the tethered force density is defined by

$$\mathbf{F}_T(s) = k_T(\mathbf{X}_T(s) - \mathbf{X}(s)),$$

where k_T is the stiffness constant.

The flow around obstacles, such as circular or square cylinders, has been extensively studied for Newtonian fluids [2,5,24–26]. Flows around these obstacles exhibit a variety of phenomena, such as boundary layer separation, vortex shading, lift, and drag forces. In general, the flow around a circular cylinder can be categorized by the range of the Reynolds number, as follows:

- $Re \leq 5$, regime of unseparated flow,
- $10 \leq Re \leq 40$, a fixed pair of Föppl vortices in the wake of the cylinder,
- $40 \leq Re \leq 150$, periodic the von Karman vortex street, and
- $150 \leq Re \leq 300$, transition to turbulence in vortex street.

Figure 5 shows vorticity and streamline distribution for fluid flow across 2D cylinders, depending on the various Reynolds numbers. The flow around the circular cylinder according to the Reynolds number is consistent with the literature. We can see that vortex shading is produced behind the obstacle when $Re = 100$ and 200. Moreover, the length of the vortex shedding and distance between Karman vortices in the square cylinder appears longer than in the circular cylinder. Figure 6 shows the drag coefficients of circular and square cylinders as a function of the Reynolds numbers. The drag coefficient is a dimensionless number used to quantify the drag of an object in a fluid environment. The higher the Reynolds number, the smaller the drag coefficient. The drag coefficient is higher in the square cylinder than in the circular cylinder.

Table 1 lists the drag coefficients and lift coefficients calculated for $Re = 100$ and 200. The value of the present result is higher than that of the references. This is attributed to constraints caused by treating rigid bodies while using the IB method. In the traditional IB method, the stiffness constant was set high to indicate a fixed rigid body in response to the flow. This makes numerical calculations unstable and makes it difficult to calculate the correct coefficients. Despite these limitations, simulating the rigid body using traditional IB methods has the advantage of dealing with complex boundaries. Figure 7 shows the flow around the obstacle, which includes not only circular and rectangular cylinders, but also star-shaped boundaries of complex boundaries. Looking at the flow around the star-shaped boundary, we can see that the pointed part of the shape affects the flow. In other words, it is also sufficient to use traditional IB methods to identify the behavioral characteristics of flows around complex boundaries.

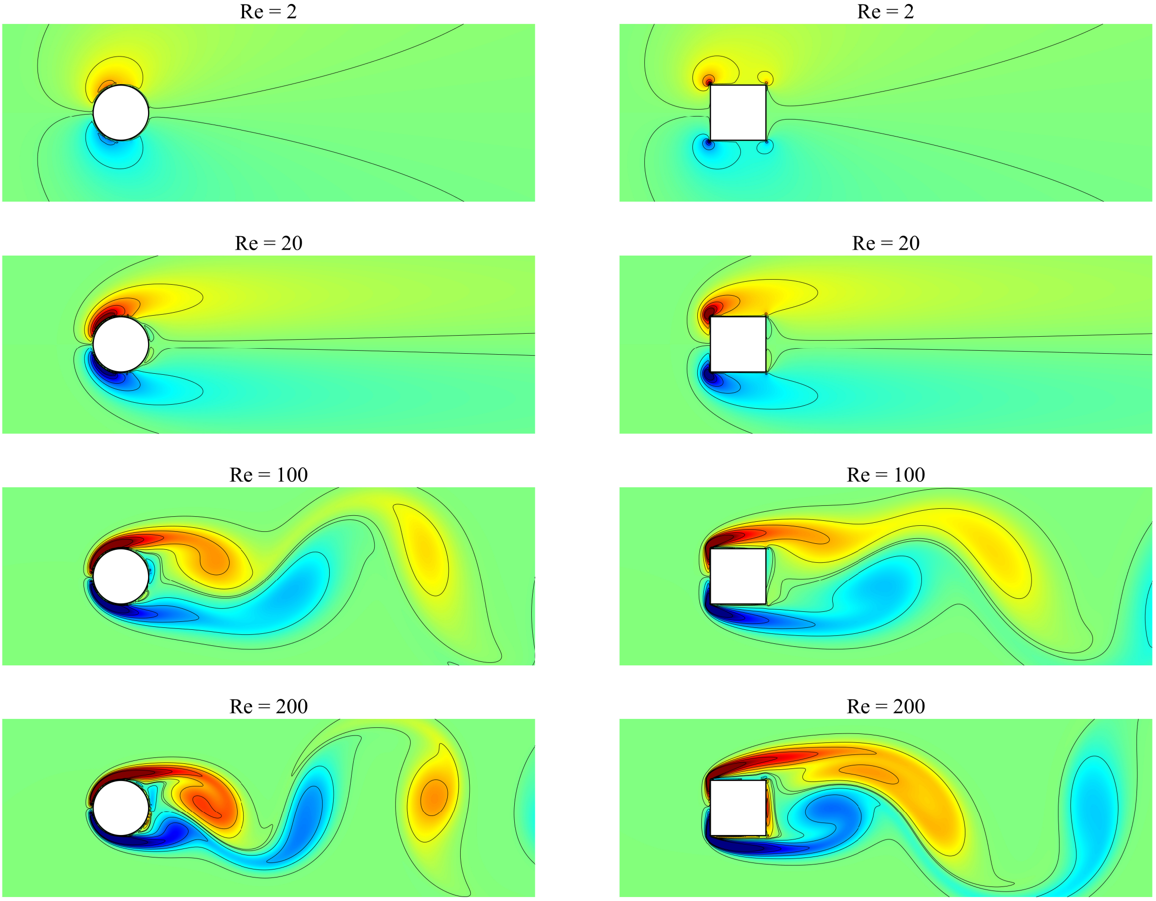


Figure 5. Vorticity and streamline distribution of fluid flow across 2D circular (left panels) and square (right panels) cylinders at $t = 10$ depending on the various Reynolds numbers.

Table 1. Lift and drag coefficients of flow around a circular cylinder at $Re = 100$ and 200.

	Lift Coefficient (CL)		**Drag Coefficient (CD)**		**Reference**
	Re 100	**Re 200**	**Re 100**	**Re 200**	
Braza et al. (1986)	±0.25	±0.75	1.364 ± 0.015	1.40 ± 0.05	[24]
Ding et al. (2007)	±0.287	±0.659	1.356 ± 0.010	1.348 ± 0.05	[25]
Harichandan et al. (2010)	±0.278	±0.602	1.352 ± 0.010	1.32 ± 0.05	[26]
Present result	±0.359	±0.726	1.517 ± 0.011	1.477 ± 0.05	tw
Present result (Square)	±0.29	±0.91	1.681 ± 0.009	1.702 ± 0.061	tw

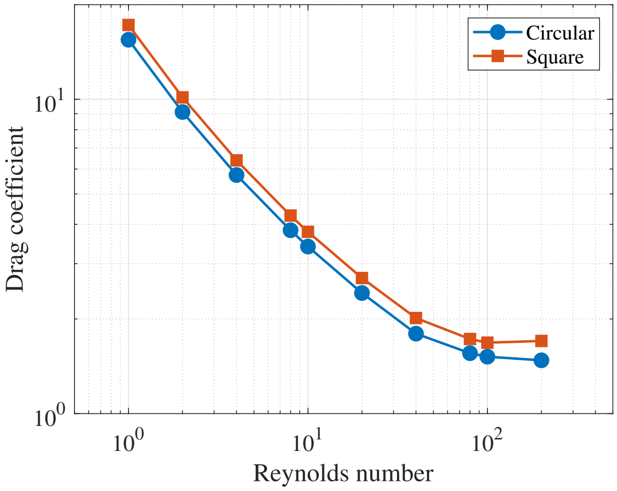


Figure 6. Drag coefficients of circular and square cylinders as a function of the Reynolds number.

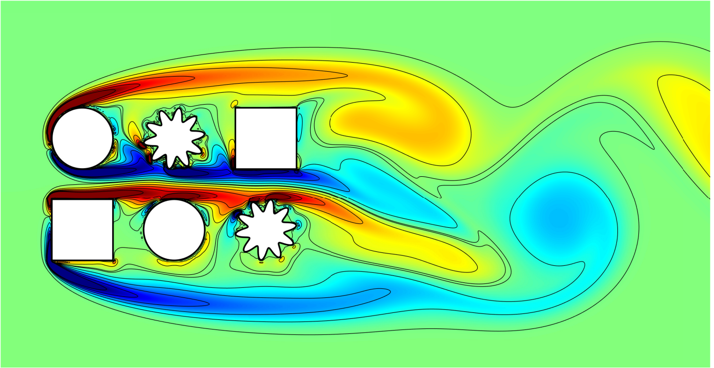


Figure 7. Flow simulation between various shapes, such as circles, squares, and star shapes at $Re = 200$.

4.4. Biological Problem

Simulations of eel, sperm, bacteria, and other swimmers have been developed whlie using the IB method. In this section, we discuss the simulation results of various motions of the *C. elegans* and present a simulation of the free swimming using the randomwalk. *C. elegans* is a nematode of approximately 1 mm length that lives in temperate soil environments. To simulate the movement of the *C. elegans*, we used one line representing the *C. elegans* body as the center skeleton. Two forces that produce movement are applied to the center line; the stretching force and bending force. The stretching force is used for representing the unextended body; the bending force is used for representing the body shape. The stretching energy is shown in Equation (3), and the bending energy is shown, as follows:

$$E[\mathbf{X}(\cdot,t)] = \frac{\sigma_b}{2}\int \left|\frac{\partial^2 \mathbf{X}}{\partial s^2}\cdot \mathbf{N} - c_0\right|^2 ds, \tag{33}$$

where σ_b is the bending coefficient, $\mathbf{N}$ is the unit normal vector to the body, and c_0 is the reference wave curvature. Various movements may be indicated, depending on the construction of c_0. For example, to simulate a forward nematode, c_0 is organized as a traveling wave, as follows:

$$c_0 = -k^2 A sin(ks - \omega t), \tag{34}$$

where $k = 2\pi/\lambda$ is the wave number with wavelength λ, A is the oscillation amplitude, and ω is the oscillation frequency.

Figure 8 shows the simulation results for various movements (forwarding motion, backward motion, rest, and coil turn) of *C. elegans* according to the c_0 configuration. The left side of each panel represents the wave shape, and the right side represents the snapshot of *C. elegans* over time. Unlike in the top panels, the wave shape in the bottom panels is consistently applied over time. In-silico model of *C. elegans* combines each of these motions to move forward or rotate to the desired point. Note that changes in wave curvature alone lead to different movements.

Randomwalk is a process that comprises a sequence of steps determined by chance. As shown in Figure 8, we simulated the randomwalk with the combination of each motion of *C. elegans*. Figure 9 shows the results of the randomwalk. Each motion consists of forward, rest, and coil turns, assuming a motion decision cycle of 0.25 s. In each panel, 'S' represents the starting point and 'E' represents the ending point. In the top panel, it can be seen that the progress of *C. elegans* changes direction immediately after the coil turn. In the lower panels, the trajectory of motion was compared according to the ratio of determination of moving (forward) and stop (rest and coil turn) motion. Each simulation was performed for 20 s and the ratio of rest and coil turns in stop motion set equal, which is 1 : 1. Table 2 shows the moving distance and turn angle according to the ratio of the moving and stop motions. We can notice that a higher rate of moving motion yields larger displacement, and a higher ratio of stop motion causes more rotation.

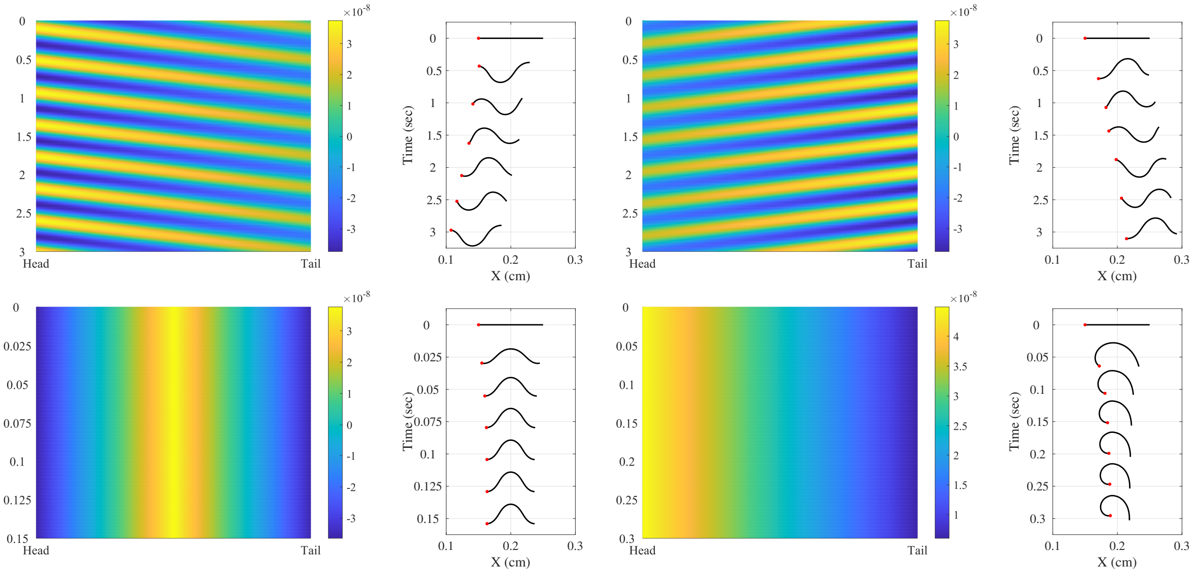


Figure 8. Simulation results for various movements of elagans according to the $c0$ configuration. left-up: forwarding motion, right-up: backward motion, left-bottom: rest, and right-bottom: coil turn.

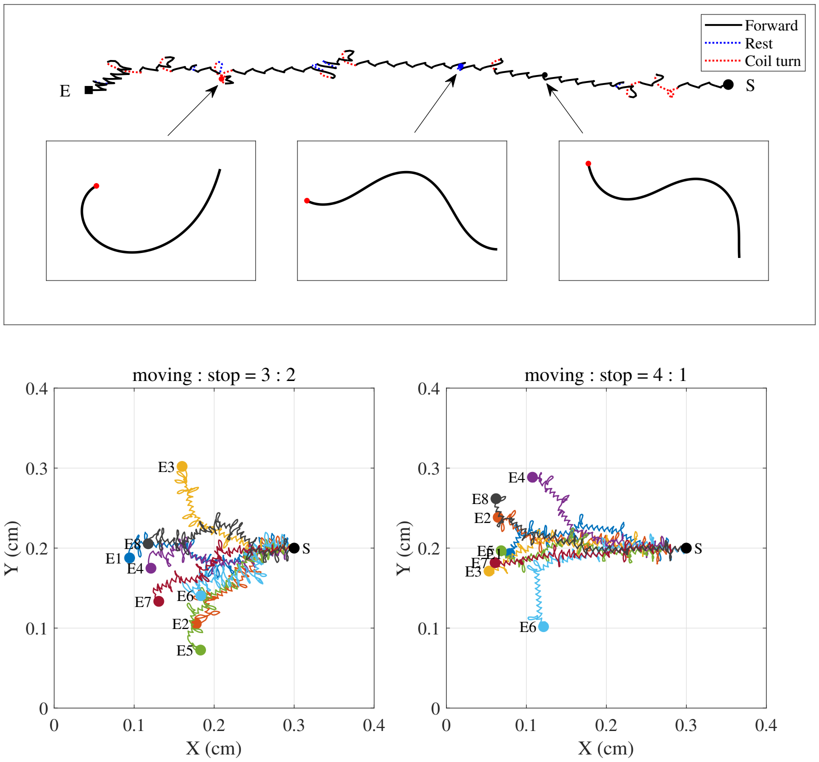


Figure 9. (top panel) Trajectory of *C. elegans* randomwalk and snapshot of forward(black), rest(blue), and coil turn(red) and (bottom panels) comparison of trajectories according to the ratio of moving and stop motion. In each panel, ’S’ and ’E’ represent the starting and the ending points, respectively.

Table 2. Moving distance and turn angle according to the ratio of moving and stop motion.

	Motion Ratio 3:2		**Motion Ratio 4:1**	
	mean	**std**	**mean**	**std**
moving distance (cm)	0.1755	0.0169	0.2223	0.0120
turn angle (°)	22.8487	17.1738	11.3732	10.5050

5. Conclusions

In this paper, we briefly reviewed the IB method in order to investigate the fluid-structure interaction problems that are governed by the NS equation and provided the MATLAB codes for numerical implementation. The essence of the IB method is its simplicity to represent the (not limited to simple) interaction between fluid and structure by coupling Eulerian and Lagrangian representations. The equations of motion and discretization were described in the IB framework. The computational results are shown as an example to facilitate the understanding of the IB method, including the choice of discrete delta functions, the effect of the Reynolds number, and the application for the biological problem. In a typical example or the volume conservation, it was shown that the choice of the discrete delta function affects the dynamics. The error estimation of regular functions is a topic that has been actively discussed. We also showed that the use of the conventional IB method is sufficient to identify the behavioral characteristics of flows around complex boundaries by considering the effect of the Reynolds number. Nevertheless, it remains an open problem to simulate flows around complex boundaries in the case of high Reynolds numbers.

Author Contributions: All authors, who contributed equally, read and approved the final manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: W.L. was supported by the National Institute for Mathematical Sciences (NIMS) grant funded by the Korea government (No. NIMS-B20900000). S.L. was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (No. 2020R1A2C1A01100114).

Acknowledgments: Many of the computations were performed using computing facilities at the National Institute for Mathematical Sciences (NFEC-2019-06-256100, NFEC-2012-03-158057). The authors also wish to thank the reviewers for the constructive and helpful comments on the revision of this article.

Conflicts of Interest: The authors declare no conflict of interest.

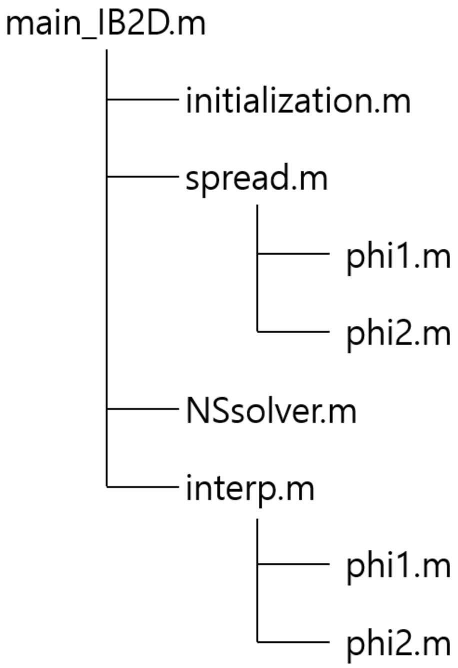
Abbreviations

The following abbreviations are used in this manuscript:

IB	immersed boundary
NS	Navier–Stokes

Appendix A. Code Structure

The MATLAB codes for our simulations are available from the corresponding author's webpage: https://sites.google.com/view/sglee/research.



- main_IB2D.m : main code, including animation code for vorticity and pressure.
- initialization.m : setting parameters, including the domain and grid sizes, Reynolds and Weber numbers, and the initial boundary configuration
- spread.m : spreading out the boundary force density to the fluid force density using the Dirac-delta function
- NSsolver.m : updating the fluid velocity and pressure by using fast Fourier transformation
- Interp.m : getting the velocity on the elastic IB boundary from the fluid velocity
- phi1.m : setting the Dirac-delta function along the x-direction
- phi2.m : setting the Dirac-delta function along the y-direction

References

1. Donea, J.; Giuliani, S.; Halleux, J. An arbitrary Lagrangian-Eulerian finite element method for transient dynamic fluid-structure interactions. *Comput. Methods Appl. Mech. Eng.* **1982**, *33*, 689–723. [CrossRef]
2. Gibou, F.; Min, C. Efficient symmetric positive definite second-order accurate monolithic solver for fluid/solid interactions. *J. Comput. Phys.* **2012**, *231*, 3246–3263. [CrossRef]
3. Jamalabadi, M. Analytical solution of sloshing in a cylindrical tank with an elastic cover. *Mathematics* **2019**, *7*, 1070. [CrossRef]
4. Jamalabadi, M. Optimal design of isothermal sloshing vessels by entropy generation minimization method. *Mathematics* **2019**, *7*, 380. [CrossRef]
5. Lee, S.; Li, Y.; Shin, J.; Kim, J. Phase-field simulations of crystal growth in a two-dimensional cavity flow. *Comput. Phys. Commun.* **2017**, *216*, 84–94. [CrossRef]
6. Peskin, C.S. Flow patterns around heart valves: A numerical method. *J. Comput. Phys.* **1972**, *10*, 252–271. [CrossRef]
7. Peskin, C.S. The immersed boundary method. *Acta Numer.* **2002**, *11*, 497–517. [CrossRef]
8. Iaccarino, G.; Verzicco, R. Immersed boundary technique for turbulent flow simulations. *Appl. Mech. Rev.* **2003**, *56*, 331–347. [CrossRef]
9. James, N.; Maitre, E.; Mortazavi, I. Immersed boundary methods for the numerical simulation of incompressible aerodynamics and fluid-structure interactions. *Ann. Mathématiques Blaise Pascal* **2013**, *20*, 139–173. [CrossRef]
10. Bächer, C.; Gekle, S. Computational modeling of active deformable membranes embedded in three-dimensional flows. *Phys. Rev. E* **2019**, *99*, 062418. [CrossRef]
11. Lee, S. Mathematical model of contractile ring-driven cytokinesis in a three-dimensional domain. *Bull. Math. Biol.* **2018**, *80*, 583–597. [CrossRef] [PubMed]
12. Lee, W.; Lim, S.; Kim, Y. The role of myosin II in glioma invasion: A mathematical model. *PLoS ONE* **2017**, *12*, e0171312. [CrossRef]
13. Fauci, L.J. A computational model of the fluid dynamics of undulatory and flagellar swimming. *Am. Zool.* **1996**, *26*, 599–607. [CrossRef]
14. Kim, Y.; Peskin, C.S. Penalty immersed boundary method for an elastic boundary with mass. *Phys. Fluids* **2007**, *19*, 053103. [CrossRef]
15. Kim, Y.; Peskin, C.S. A penalty immersed boundary method for a rigid body in fluid. *Phys. Fluids* **2016**, *28*, 033603. [CrossRef]
16. Lee, K.; Yang, K. Large Eddy simulation of turbulent flow past a circular cylinder in the subcritical and critical regimes. *J. Mech. Sci. Technol.* **2017**, *31*, 1729–1737. [CrossRef]
17. Posa, A.; Vanella, M.; Balaras, E. An adaptive reconstruction for Lagrangian, direct-forcing, immersed-boundary methods. *J. Comput. Phys.* **2017**, *351*, 422–436. [CrossRef]
18. Kang, S. An improved near-wall modelling for large-eddy simulation using immersed boundary methods. *Int. J. Numer. Methods Fluids* **2015**, *78*, 76–88. [CrossRef]
19. Yang, X.; Sadique, J.; Mittal, R.; Meneveau, C. Integral wall model for large eddy simulations of wall-bounded turbulent flows. *Phys. Fluids* **2015**, *27*, 025112. [CrossRef]
20. Peskin, C.S.; McQueen, D.M. Fluid Dynamics of the Heart and its Valves. In *Case Studies in Mathematical Modeling: Ecology, Physiology, and Cell Biology*; Othmer, H.G., Adler, F.R., Lewis, M.A., Dallon, J.C., Eds.; Prentice-Hall: Englewood Cliffs, NJ, USA, 1996.
21. Chorin, A. Numerical solution of the Navier–Stokes equations. *Math. Comput.* **1968**, *22*, 745–762. [CrossRef]
22. Liu, Y.; Mori, Y. Properties of discrete delta functions and local convergence of the immersed boundary method. *SIAM J. Numer. Anal.* **2012**, *50*, 2986–3015. [CrossRef]
23. Stockie, J.M. Analysis and Computation of iMmersed Boundaries, with Application to Pulp Fibres. Ph.D. Thesis, University of British Columbia, Vancouver, BC, Canada, 1997.
24. Braza, M.; Chassaing, P.; Minh, H.H. Numerical study and physical analysis of the pressure and velocity fields in the near wake of a circular cylinder. *J. Fluid Mech.* **1986**, *165*, 79–130. [CrossRef]

25. Ding, H.; Shu, C.; Yeo, K.S.; Xu, D. Numerical simulation of flows around two circular cylinders by mesh-free least square-based finite difference methods. *Int. J. Numer. Methods Fluids* **2007**, *53*, 305–332. [CrossRef]
26. Harichandan, A.B.; Roy, A. Numerical investigation of low Reynolds number flow past two and three circular cylinders using unstructured grid CFR scheme. *Int. J. Heat Fluid Flow* **2010**, *31*, 154–171. [CrossRef]

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.