entropy

MDPI

Article

An Explicit-Correction-Force Scheme of IB-LBM Based on Interpolated Particle Distribution Function

Bowen Liu and Weiping Shi *

School of Mathematics, Jilin University, Changchun130012, China; liubowen0830@gmail.com
* Correspondence: shiwp@jlu.edu.cn

Abstract: In order to obtain a better numerical simulation method for fluid–structure interaction (FSI), the IB-LBM combining the lattice Boltzmann method (LBM) and immersed boundary method (IBM) has been studied more than a decade. For this purpose, an explicit correction force scheme of IB-LBM was proposed in this paper. Different from the current IB-LBMs, this paper introduced the particle distribution function to the interpolation process from the fluid grids to the immersed boundary at the mesoscopic level and directly applied the LBM force models to obtain the interface force with a simple form and explicit process. Then, in order to ensure the mass conservation in the local area of the interface, this paper corrected the obtained interface force with the correction matrix, forming the total explicit-correction-force (ECP) scheme of IB-LBM. The results of four numerical tests were used to verify the order of accuracy and effectiveness of the present method. The streamline penetration is limited and the numerical simulation with certain application significance is successful for complex boundary conditions such as the movable rigid bodies (free oscillation of the flapping foil) and flexible deformable bodies (free deformation of cylinders). In summary, we obtained a simple and alternative simulation method that can achieve good simulation results for engineering reference models with complex boundary problems.

Keywords: lattice Boltzmann method; immersed boundary method; fluid–structure interaction; deformable body

check for updates

Citation: Liu, B.; Shi, W. An Explicit-Correction-Force Scheme of IB-LBM Based on Interpolated Particle Distribution Function. *Entropy* **2023**, *25*, 526. https://doi.org/10.3390/e25030526

Academic Editor: Antonio M. Scarfone

Received: 7 February 2023
Revised: 9 March 2023
Accepted: 16 March 2023
Published: 17 March 2023

1. Introduction

Fluid–structure interaction (FSI) has always been a research problem faced by various disciplines of natural science, such as the movement of fish and birds, the vibration of bridges, the computation of flow in porous media, the deformation of cells, the blood flow in the heart, the transportation of solid particles, and the surface design of cars and airplanes in aerodynamics. In the past few decades, researchers have made great efforts in the numerical simulation of FSI [1–5].

At present, the general method for complex fluid–solid coupling problems is the decoupling algorithm [6], which decouples and solves multiple equations involved in the problem. It has strong adaptability to complex problems and low computational complexity, and it is easy to integrate the existing advantages of solving single-type equations (such as fluid control equations and solid mechanics equations). Under the decoupling algorithm, the fluid–solid solution algorithm can be divided into two types according to the grid-processing technology of the fluid–solid interface: one is the body-fitted grid method, and the other is the fixed grid method. For the body-fitted grid method, the Arbitrary Lagrangian–Eulerian method is usually used [7,8]. This type of method has the advantage of ensuring a clear interface, but it has some drawbacks for some fluid problems. For example, when the grid deforms greatly, it is very difficult to construct a robust and high-accuracy scheme. Simultaneously, this method increases the computational cost due to the frequent mesh regeneration process, especially when considering complex and/or three-dimensional geometric shapes. To avoid these shortcomings, fixed grid technology uses regular fixed grids to discretize the fluid field, describing the influence of solids on the surrounding fluid with the additional physical force. This simplicity greatly reduces the computational cost and is very popular in FSI with complex geometries. However, the fixed grid method usually uses the interpolation method to process the interface, which will affect the accuracy of the numerical scheme. The development of a method that takes both speed and accuracy into consideration has never stopped in these decades. The immersive boundary method and the lattice Boltzmann method are typical representatives of fixed grid methods before they are combined.

The immersed boundary (IB) method was proposed by Peskin [9]: the flow field is solved on a fixed Euler grid, the interface is represented by Lagrangian grids, and the interface force which is treated as the source term of the fluid equation is obtained by the solid equation. This approach overcomes the defects of large deformation and is suitable for dealing with complex interface problems. The original IBM has first-order accuracy of space and the sharp boundary is smoothed to only one mesh width [10]. The substantial advantages of the IB method drive researchers to make further efforts to improve its accuracy. Yang, X. et al. [11] and Bao, Y. et al. [12] developed an optimization method of the interpolation functions. Fadlun et al. developed a ghost cell method [13]. Yang, J. [14] used the bi-linear interpolation scheme for IBM [14]. Griffith reviewed IBMS for elastic bodies and others [15]. The test proves the effectiveness of the IBMs for biological simulation.

LBM is a kind of computational fluid dynamics method for Euler grids [16,17]. This method can calculate flow fields under many complex conditions [18], such as turbulence for high Reynolds number problems, rarefied gas for big Knudsen numbers, thermal, sound waves, etc. Although most papers addressing IB-LBM only treat LBM as an alternative to the N-S solver, we think this underestimates the ability of LBM to solve the flow field. The LBM programming is simple and easy to parallelize, guaranteeing calculation speed [19]. The pure LBM method can also solve fluid–solid problems, and the difficulty lies in the treatment of boundary conditions. In past research, many boundary schemes of the complex boundary conditions have been developed, such as bounce-back interpolation scheme [20], non-equilibrium interpolation schemes [21], and so on. However, most of the LBM boundary conditions require the computation of the intersection of Euler mesh and solid boundary so far. Especially for moving rigid bodies by the LB model [22–25], the solid boundary, the internal solid nodes, and the external fluid nodes need to be constantly replaced, although some methods, such as the interpolation bounce method, can guarantee second-order accuracy [26,27].

The combination of IB-LBM is a manifestation of complementary advantages. The IBM of the IB-LBM can be regarded as a special LBM complex boundary treatment scheme, and its main advantage is the forward tracking feature, that is, the shape of the boundary is directly known and does not need to be reconstructed. The implementation of LBM is relatively simple, and its numerical cost is small. Currently, IB-LBM can be applied to many numerical simulations [26–31], and the theoretical improvement of IB-LBM has never stopped. Feng first presented the combination of the IBM and LBM to simulate the rigid particle–fluid interaction [32]. They computed the force of the fluid–solid interaction using a penalty method; the idea of the penalty method is to assume a slight change for each Lagrangian marker point on the immersed boundary, and then add a penalty force for each point to offset this change, so the points will be pulled towards to their reference position. This explicit IB-LBM is easy to compute and implement but shows unideal accuracy and stability of the results because of the limitation of artificial parameters and the non-strict satisfaction of the no-slip boundary condition [32]. To avoid the use of artificial parameters in the explicit IB-LBM, Feng et al. used the direct-force IBM to calculate the interface force. Although the IB-LBM method has improved efficiency and accuracy, the process is complicated because the force form which is obtained by solving the N-S equations is complex [33,34]. Wu proposed an implicit velocity correction method for IB-LBM [35]. The core idea of the method is to divide the flow field velocity into uncorrected velocity and corrected velocity, and the force between the fluid and solid interface is given by solving the corrected velocity of nodes on the immersed boundary. In the benchmark tests of the rigid body in fluid, the phenomenon of streamline penetration of the traditional IB-LBM is effectively avoided, indicating that the method can meet the non-slip boundary condition [36]. However, this method increases the complexity and instability of the calculation because this method adds an interpolation matrix inversion and implicit process of it at each time step [37]. Kang presented a multi-direct-forcing IB-LBM to decrease the computation cost of the implicit IB-LBM [38]. This method inherits the idea of velocity correction in the implicit IB-LBM but uses an iterative scheme instead of constructing and inverting a large matrix at each time step. However, the iteration only occurs on lattice nodes closed to the interface. This method may lead to a large number of iterative steps [38]. For this reason, T Seta proposed a non-iterative implicit IB-LBM method, but this method obviously can only provide relatively low accuracy and slightly different results compared with implicit and iterative schemes [39]. Wang developed an improved approximation method to combine the implicit IB-LBM method [40]. Gsell uses an analytical approximation of the non-reciprocity error to correct immersed boundary force and prevent boundary slip and flow penetration [41]. And an extremely simple force amplification technique proposed for IBM in the correction of boundary slip can be a reference for IB-LBM [42]. Afra, B. et al. developed a robust lattice spring model (LSM) with the spring tension in multiple directions, which could be an implicit scheme of IB-LBM for large deformation issues; this is an effective model and obtains good results in deformation simulation [43]. Additionally, some fluid control problems of flexible filaments with good simulation results have been studied by LSM [29,30].

We call the methods mentioned above macro-scale methods. This type of method models the fluid–structure interface only at the macroscopic scale, and the modeled physical quantities are also macroscopic physical quantities, so the combined meaning of IBM and LBM will be limited. To discover a deeper level of physical meaning and calculation methods, a few papers discussed IB-LBM on a mesoscopic scale. Niu first proposed an IB-LBM on the mesoscopic scale called the momentum-exchange-based IB-LBM [44]. The basic idea of this method is to interpolate the velocity distribution function of the LBM as a physical quantity of IBM and use the bounce-back scheme to treat the distribution function on the coupled interface, and then the interaction force on the interface can be calculated by the momentum exchange method. Hu proposed an iterative method to improve IB-LBM based on momentum exchange so that the original method can better meet the no-slip boundary conditions [45]. Yuan extended the method to the calculation of flexible bodies and the numerical results showed the adaptability of the method in the flexible body cases [46]. Tao presented an IB-LBM in which the no-slip boundary condition is implemented directly by correcting the distribution functions near the interface [47]. Wang used the half-bounce-back scheme to construct a boundary treatment method so that the IB-LBM on the mesoscopic scale can effectively treat the large curvature boundary conditions [48].

Regarding the accuracy order of IB-LBM, current reports are inconsistent; some papers report first-order [27,49] and some papers report second-order accuracy [35,50].

In order to find an alternative and optimal method, first of all, this paper tried to find a new method with deeper physical significance and an acceptable order of accuracy at the mesoscopic scale. Secondly, it absorbed the advantages of several previous methods, such as the original method with a simple form, and the implicit method for satisfaction of the no-slip boundary conditions, etc. Finally, it simulated the more complex working conditions (such as unsteady flow, movable rigid body, and flexible body) identically to the current explicit methods. Different from the current IB-LBMs, this paper used the particle distribution function (a mesoscopic physical quantity rather than the macroscopic physical quantity velocity in IBM) for the interpolation process, directly applied the LBM force models of the Euler grids to the Lagrangian grids, and obtained the interface force. In order to ensure the mass conservation in the local area of the interface, this paper further corrected the obtained interface force with the matrixes, forming an explicit correction force scheme of the IB-LBM proposed in this paper. From the analysis of the four groups of results, it can be seen that the method proposed in this paper is a feasible and effective method.

2. Related Work

- **Lattice Boltzmann Equation**

For the fluid domain, the applicable range of the continuum hypothesis is limited. In order to expand the description range of the flow field by the fluid-governing equations, the fluid domain can be described by continuous Boltzmann equations according to the gas kinetic theory. Since the collision integral term which closes the continuous Boltzmann equation is very complicated, it is necessary to simplify the collision integral term. Additionally, the Boltzmann-BGK equation based on the linear collision assumption is the most widely used at present [51], which is given by

$$\frac{\partial f}{\partial t} + \boldsymbol{\xi} \cdot \nabla_x f + \boldsymbol{a} \cdot \nabla_{\boldsymbol{\xi}} \cdot f = -\frac{1}{\tau_c}\left[f - f^{(eq)}\right] \tag{1}$$

where $f = f(\mathbf{x}, \boldsymbol{\xi}, t)$ is the particle distribution function, $\mathbf{x}$ is the space displacement vector, $\boldsymbol{\xi}$ is the velocity vector, t is the time, $f^{(eq)}$ is the equilibrium distribution function, τ_c is the relaxation time, and $\frac{1}{\tau_c}\left[f - f^{(eq)}\right]$ is the BGK collision term.

The Lattice Boltzmann equation is a discrete equation of the continuous Boltzmann equation realized by a special difference scheme. As for the applicable range of the continuum hypothesis, the flow field can also be described by the macroscopic equations of fluid mechanics, such as the N-S equation, the convection–diffusion equation, etc. The corresponding LB model equations of these macroscopic equations can be achieved through multi-scale expansion techniques. Among the IB-LBM calculation examples, the LBGK equation is the most typical [52].

Let $\{\boldsymbol{c}_0, \boldsymbol{c}_1, \ldots, \boldsymbol{c}_n\}, \quad i = 1, 2, \ldots, n$ be the discrete velocity space; then, there is a discrete velocity distribution function $f_i = f_i(\boldsymbol{x}, \boldsymbol{c}_i, t)$. Substituting these into Equation (1) and discretizing it in time and space, the lattice Boltzmann equation with external force term is given by

$$f_i(\boldsymbol{x} + \boldsymbol{c}_i\Delta t, t + \Delta t) - f_i(\boldsymbol{x}, t) = -\frac{1}{\tau}\Big(f_i(\boldsymbol{x}, t) - f_i^{eq}(\boldsymbol{x}, t)\Big) + S_i \tag{2}$$

where $\tau = \tau_c/\Delta t$ is the dimensionless relaxation time, usually $\tau > 0.5$ can guarantee stability.

It is worth noting that Equation (2) has second-order accuracy in space time but most IBMs and IB-LBMs have first-order accuracy [19].

The S_i of Equation (2) is the collision source term, and the discrete force scheme of Guo [53] is commonly used:

$$S_i = \left(1 - \frac{\Delta t}{2\tau}\right) w_i \left(\frac{\boldsymbol{c}_i - \boldsymbol{U}}{c_s^2} + \frac{(\boldsymbol{c}_i \cdot \boldsymbol{U}) \cdot \boldsymbol{c}_i}{c_s^4}\right) \cdot \boldsymbol{F}(\boldsymbol{X}, t) \tag{3}$$

The DnQm model proposed by Qian [54] is the most commonly used discrete velocity scheme and the D2Q9 model was used in this paper. The discrete velocity space $\boldsymbol{c}_i$ is expressed by

$$\boldsymbol{c}_i = \begin{cases} 0 & i = 0 \\ \left(\cos\left[(i-1)\frac{\pi}{2}\right],\ \sin\left[(i-1)\frac{\pi}{2}\right]\right)\frac{\Delta x}{\Delta t} & i = 1,2,3,4 \\ \left(\cos\left[(2i-1)\frac{\pi}{4}\right],\ \sin\left[(2i-1)\frac{\pi}{4}\right]\right)\frac{\Delta x}{\Delta t} & i = 5,6,7,8 \end{cases} \tag{4}$$

The discrete equilibrium distribution function is given by

$$f_i^{eq}(\boldsymbol{X}, t) = \rho w_i \left[1 + \frac{\boldsymbol{c}_i.\boldsymbol{u}}{c_s^2} + \frac{(\boldsymbol{c}_i.\boldsymbol{u})^2}{2c_s^4} - \frac{\boldsymbol{u}^2}{2c_s^2}\right] \tag{5}$$

where c_s is the lattice sound velocity and w_i is the weight coefficient given by

$$c_s = \frac{1}{\sqrt{3}}\frac{\Delta x}{\Delta t}, \quad w_i = \begin{cases} \frac{4}{9} & i = 0 \\ \frac{1}{9} & i = 1 \sim 4 \\ \frac{1}{36} & i = 5 \sim 8 \end{cases} \tag{6}$$

The dynamic viscosity in the LBM is given by

$$v = c_s^2(\tau - 0.5)\Delta t \tag{7}$$

The macro quantity velocity ρ and momentum $\rho \boldsymbol{u}$ can be obtained by

$$\rho = \sum_i f_i, \quad \rho \boldsymbol{u} = \sum_i \boldsymbol{c}_i f_i + \frac{\mathbf{f}\Delta t}{2} \tag{8}$$

- **Immersed Boundary Method**

As shown in Figure 1, IBM adopts the Euler–Lagrangian grid scheme. Consider a bounded region $\Omega \subset R^n(n = 2,3)$ as the coupling region, Ω_+ and Ω_- are the fluid and solid geometric regions respectively before immersion. Based on the immersed boundary assumption, let Γ=$\Omega_+ \cap \Omega_-$ be the fluid–solid coupling-immersed interface, Ω=$\Omega_+ \cup \Omega_-$ is the fluid calculation domain, and Γ satisfies the non-slip boundary condition.

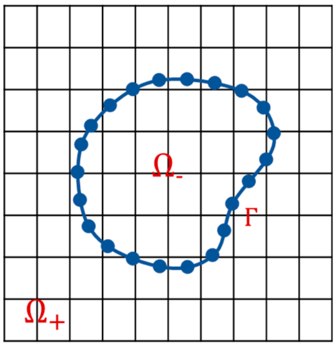


Figure 1. Computational domain with an immersed boundary in 2D.

The Euler coordinate points $x = (x_i, i = 1,2,3) \in \Omega$ describe the fixed grid coordinates of the fluid, and arc length coordinates $s(s_i, i = 1,2,3) \subset K$ as the Lagrange coordinates to describe interface particles. A mapping $\boldsymbol{X}(s,t) \in \Omega$ is established to show the physical position of the particles with arc length parameters s at time t. Let L be the geometric length of the interface, K be the collection of particles, and Γ_t be the shape of the interface at time t:

$$\Gamma_t = \boldsymbol{X}(K,t), \;\; \boldsymbol{X}(0,t) = \boldsymbol{X}(L,t) \tag{9}$$

For the interpolation of physical quantities, the delta function interpolation scheme proposed by Peskin [55,56] is the commonly used:

$$\mathbf{f}(\mathbf{x}) = \int_\Gamma \mathbf{F}(s)\delta(\mathbf{x} - \mathbf{X})ds, \; \mathbf{U}(\mathbf{X}) = \int_\Omega \mathbf{u}(\mathbf{x})\delta(\mathbf{x} - \mathbf{X})dx \tag{10}$$

where $\delta(\mathbf{x})$ is the delta function, which can only guarantee first-order accuracy. $\mathbf{f}, \mathbf{u}$ are force and velocity on the Euler grids for the fluid, and $\mathbf{F}$, $\mathbf{U}$ are force and velocity on the Lagrangian grids for the solid. Additionally, $\mathbf{F}$ is determined by the constitutive relation at the immersed boundary, usually Hooke's law.

As the comprehensive consideration of speed and error, two commonly used delta functions $\delta(x)$ [11] in this paper are given by

$$\delta(x) = \begin{cases} \frac{1}{8}\left(3 - 2|x| + \sqrt{1 + 4|x| - 4x^2}\right) & (|x| < 1) \\ \frac{1}{8}\left(5 - 2|x| - \sqrt{-7 + 12|x| - 4x^2}\right) & (1 \le |x| \le 2) \\ 0 & (|x| > 2) \end{cases} \tag{11}$$

A smooth function of $\delta(x)$ is given by

$$\delta'(x) = \begin{cases} \frac{1}{4}(1 + \cos(\frac{\pi x}{2})) & (|x| \le 2) \\ 0 & (|x| > 2) \end{cases} \tag{12}$$

- **Lattice Boltzmann-Immersed Boundary Method in other Work**

A traditional IB-LBM equation can be established through Equation (2), Equation (8), and Equation (10), but **f**, **F**, **U** are unknown. Once **f** or **F** is obtained, then **U** can be solved by solid equations.

We enumerate four representative IB-LBM methods at present. They are the traditional explicit method, the traditional implicit method, the direct stress integration method, and the bounce-back momentum exchange method, where the bounce-back momentum exchange method is a pioneering representative method of the mesoscopic method. The key to all methods is to obtain the **f** or **F** to solve the IB-LBM equations.

(1) The traditional explicit method [22]

The traditional explicit method is carried out under the assumption that the movement of the solid point satisfies Hooke's law.

$$\boldsymbol{F} = k \cdot \Delta \boldsymbol{X} \tag{13}$$

where $\Delta\boldsymbol{X}$ is the displacement of the boundary Lagrangian point, $\boldsymbol{F}$ is the imaginary bound-back force due to $\Delta\boldsymbol{X}$, and k is the given stiffness parameter.

(2) The traditional implicit method [35]

The velocity correction method was proposed by Wu by solving a system of linear equations (see [35] for details); the correction velocity at the Euler point is implicitly obtained, and then the force density at the Euler node is obtained.

$$f = 2\rho\frac{\boldsymbol{u} - \overline{\boldsymbol{u}}}{\Delta t} = 2\rho\frac{\Delta \boldsymbol{u}}{\Delta t} \tag{14}$$

where $\overline{\boldsymbol{u}}$ is the uncorrected velocity in fluid domain.

(3) The direct stress integration method [33]

In this method, the interface stress is obtained by introducing the N-S equation into the interface, which is similar to the stress integration method in the traditional LBM. The solution result of the solid force density is given by

$$F_\alpha = \rho\frac{U_\alpha - u_\alpha}{\Delta t} + \rho u_\beta \partial_\beta u_\alpha + \partial_\alpha p - \mu\partial_\beta^2 u_\alpha \tag{15}$$

where $p = c_s^2\rho$ is the positive pressure and α,β respectively represent the coordinate directions under the 2D condition of $\boldsymbol{F}$.

(4) The bounce-back momentum exchange method [44]

The bounce-back momentum exchange method opens a precedent for exploring the mesoscopic immersed boundary method. This method is established under the condition of elastic collision. The interface force density is given by

$$\boldsymbol{F} = \sum_i \boldsymbol{c}_i(f_i^{new} - f_i) = \sum_i \boldsymbol{c}_i\left(f_{-i} - f_i - 2w_i\rho\frac{\boldsymbol{c}_i\boldsymbol{u}}{c_s^2}\right) \tag{16}$$

where $-i$ is the opposite direction of i and f_i^{new} is the distribution function by the bound-back scheme.

3. The Present Explicit Correction Force Scheme for IB-LBM

When applying the direct stress integration method and the bound-back momentum exchange algorithm to solve the interface force, the problems of low accuracy and not satisfying the non-slip boundary are obvious [17]. For the former, it is easy to produce large noise, and the calculation is too cumbersome. This paper proposed a mesoscopic force calculation method that has a simple form and satisfies no-slip boundary conditions.

Based on the continuous medium around the interface and the physical quantity of the flow field being a continuous real number [57], we made the following assumptions:

(1) The discrete-velocity distribution function f_i can be used as an interpolation physical quantity in the direction from Euler to Lagrangian in Equation (17). (The crash of our program may suggest that it cannot be used as an interpolation quantity from Lagrangian to Euler);

(2) The velocity **U** obtained from the solid Equations (21,22) on the Lagrangian point can be regarded as the equilibrium velocity of the LBM force model in Equation (8);

(3) The force model of LBM in Equation (8) is still applicable at the Lagrangian point.

Satisfying the three abovementioned assumptions, we derived the force equation on the immersed boundary, as follows.

- **Explicit Force by the Interpolated Distribution Function and LBM Force Models**

Owing to our assumptions, we can remove the limit of original fixed grid technology by the interpolation method of the velocity distribution function [58].

We defined the set of all Lagrangian marked points by $L_p = \{\boldsymbol{X}_1, \boldsymbol{X}_2, \cdots, \boldsymbol{X}_N\}$. For any $\boldsymbol{X}_n \in L_p$, let $f_i(\boldsymbol{X}_n, t)$ be the reconstructed velocity distribution function at the Lagrangian point $\boldsymbol{X}_n$. We have

$$f_i(\boldsymbol{X}_n, t) = \sum_{m=1}^{M} f_i(\boldsymbol{x}_m, t) I(\boldsymbol{x}_m - \boldsymbol{X}_n) h^d \tag{17}$$

where h is the Euler grid length, d is the space dimension, and $I(\boldsymbol{x}_m - \boldsymbol{X}_n)$ is the discrete delta function [55]. Let $\boldsymbol{x}_m \in E_p = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_M\}$ be the Euler coordinate point set to be interpolated. Additionally, abbreviate $I(\boldsymbol{x}_m - \boldsymbol{X}_n)$ as

$$I_{mn} = I(\boldsymbol{x}_m - \boldsymbol{X}_n) = \frac{1}{h^d} \prod_{j=1}^{d} \delta\left(\frac{x_{m,j} - X_{n,j}}{h}\right) \tag{18}$$

where $x_{m,j}$ and $X_{n,j}$ are the coordinate values of the points $\boldsymbol{x}_m$ and $\boldsymbol{X}_n$ at the coordinate direction of **j**, respectively. $\delta(x)$ is given by Equation (11) or (12).

Under our assumptions, the moment equations satisfy

$$\rho(\boldsymbol{X}_n, t)\boldsymbol{U}(\boldsymbol{X}_n, t) = \sum_i \boldsymbol{c}_i f_i(\boldsymbol{X}_n, t) + A\boldsymbol{F}(\boldsymbol{X}_n, t)\Delta t \tag{19}$$

where ρ and $\boldsymbol{U}$ are the density and velocity of the fluid at the Lagrangian point, respectively.

The selection of A is related to the choice of the LBM force model. We obtain get a conclusion from ref. [17]. A can be 0.5, τ, and 0 with different $\mathrm{S_i}$ in Equation (2).

From Equations (19) and (20), the interfacial force exerted on the fluid by the solid can be calculated explicitly by

$$\boldsymbol{F}(\boldsymbol{X}_n, t) = \frac{\rho(\boldsymbol{X}_n, t)\boldsymbol{U}(\boldsymbol{X}_n, t) - \sum_i \boldsymbol{c}_i f_i(\boldsymbol{X}_n, t)}{A\Delta t} \tag{20}$$

The calculation of $\boldsymbol{U}(\boldsymbol{X}_n, t)$ is determined by the equations of the solid. We provide a general calculation equation; for any $\boldsymbol{X}_n \in L_p$, we have

$$\rho_s \frac{\partial^2 \boldsymbol{X}}{\partial t^2} = \boldsymbol{f} + \boldsymbol{F}_{in} + \boldsymbol{F}_{ext} \tag{21}$$

$$\boldsymbol{U}(\boldsymbol{X}_n, t) = \frac{\partial \boldsymbol{X}}{\partial t} \tag{22}$$

where ρ_s is the mass density of the solid node on the immersed boundary, $\boldsymbol{f}$ is the interface force exerted on the solid by the fluid, $\boldsymbol{F}_{in}$ is the internal force generated inside the solid because of external forces, and $\boldsymbol{F}_{ext}$ is the resultant force generated in other situations.

When the solid is a rigid body, we have

$$\boldsymbol{U}(\boldsymbol{X}_n, t) = 0; \tag{23}$$

When the solid is a deformable body, we need further modeling in $\boldsymbol{F}_{in}$ and $\boldsymbol{F}_{ext}$ to obtain $\boldsymbol{U}(\boldsymbol{X}_n, t)$; Section 4.4 shows an example for deformation.

If the $\boldsymbol{U}(\boldsymbol{X}_n, t)$ is obtained, $\mathbf{F}(\boldsymbol{X}_n, t + \Delta t)$ can be obtained.

- **Theoretical Correction of Explicit Force by Correction Matrix**

Because of the special nature of the δ function, the implicit method used the corrected velocity to obtain the velocity on the Euler grids [35]. Different from the implicit method obtaining the correct velocity on Euler grids, we directly corrected the force on the immersed boundary.

We need to modify the force $\mathbf{F}(\mathbf{X}_n, t)$ obtained in Equation (20). Additionally, let $\overline{\boldsymbol{L}} = \left[\overline{\boldsymbol{F}}(\boldsymbol{X}_1, t), \overline{\boldsymbol{F}}(\boldsymbol{X}_2, t), \ldots, \overline{\boldsymbol{F}}(\boldsymbol{X}_N, t)\right]^T$ be the force density vector to be corrected at the Lagrangian point, $\boldsymbol{L} = [\boldsymbol{F}(\boldsymbol{X}_1, t), \boldsymbol{F}(\boldsymbol{X}_2, t), \ldots, \boldsymbol{F}(\boldsymbol{X}_N, t)]^T$ be the corrected force density vector at the Lagrangian point, and $\boldsymbol{E} = [\boldsymbol{f}(\boldsymbol{x}_1, t), \boldsymbol{f}(\boldsymbol{x}_2, t), \ldots, \boldsymbol{f}(\boldsymbol{x}_M, t)]^T$ be the force density vector on the Euler point.

The physical quantities spread from the Lagrangian point to the Euler point, satisfying the following spread discrete function which is similar as Equation (18):

$$S_{mn} = S(\boldsymbol{x}_m - \boldsymbol{X}_n) = \frac{1}{h^d} \prod_{j=1}^{d} \delta\left(\frac{x_{m,j} - X_{n,j}}{h}\right) \tag{24}$$

IBM's interpolation process introduces errors in the force calculations [35]. We think that the correct force is numerically lost in the process of interpolation and spreading, so we restored the current force to obtain the correct force from an inverse process.

Introducing spread function matrix $\boldsymbol{S}$, interpolation function matrix $\boldsymbol{I}$, Euler space unit matrix $\boldsymbol{E}_h$, and Lagrangian space unit matrix $\boldsymbol{L}_s$, we have

$$\boldsymbol{I}\boldsymbol{E}_h\boldsymbol{E} = \overline{\boldsymbol{L}} \tag{25}$$

$$\boldsymbol{S}\boldsymbol{L}_s\boldsymbol{L} = \boldsymbol{E} \tag{26}$$

where

$$\boldsymbol{S} = \begin{bmatrix} S_{11} & S_{12} & \cdots & S_{1N} \\ S_{21} & S_{22} & \cdots & S_{2N} \\ \cdots & \cdots & \ddots & \vdots \\ S_{M1} & S_{M2} & \cdots & S_{MN} \end{bmatrix}, \quad \boldsymbol{I} = \begin{bmatrix} I_{11} & I_{12} & \cdots & I_{1M} \\ I_{21} & I_{22} & \cdots & I_{2M} \\ \cdots & \cdots & \ddots & \vdots \\ I_{N1} & I_{N2} & \cdots & I_{NM} \end{bmatrix} \tag{27}$$

$$\boldsymbol{E}_h = \begin{bmatrix} h^d & 0 & \cdots & 0 \\ 0 & h^d & \cdots & 0 \\ \cdots & \cdots & \ddots & \vdots \\ 0 & 0 & \cdots & h^d \end{bmatrix}, \boldsymbol{L}_s = \begin{bmatrix} \Delta s_1 & 0 & \cdots & 0 \\ 0 & \Delta s_2 & \cdots & 0 \\ \cdots & \cdots & \ddots & \vdots \\ 0 & 0 & \cdots & \Delta s_N \end{bmatrix} \tag{28}$$

After finishing Equations (25) and (26), we have

$$\boldsymbol{I}\boldsymbol{E}_h\boldsymbol{S}\boldsymbol{L}_s\boldsymbol{L} = \overline{\boldsymbol{L}} \tag{29}$$

where $\boldsymbol{T} = \boldsymbol{I}\boldsymbol{E}_h\boldsymbol{S}\boldsymbol{L}_s$ is the modified transformation matrix, then

$$\boldsymbol{L} = \boldsymbol{T}^{-1}\overline{\boldsymbol{L}} \tag{30}$$

So far, we have given a calculation method of correction force with matrix inversion. Obviously the matrix is a sparse matrix. The LU decomposition is suggested to be applied into Equation (30).

- **The Total Process for FSI by the Proposed Method**

Combining Equation (2), Equation (20), and Equation (30), we can obtain the correction force calculation difference equations proposed in this paper:

$$f_i(\boldsymbol{x} + \boldsymbol{c}_i\Delta t, t + \Delta t) - f_i(\boldsymbol{x}, t) = \sum \Omega_i \ \ on\ \Omega_+ \cup \Omega_- \tag{31}$$

$$\boldsymbol{L}^t = \left(\boldsymbol{T}^t\right)^{-1}\overline{\boldsymbol{L}}^t \ \ on\ \ \Gamma \tag{32}$$

$$\rho_s \frac{\boldsymbol{U}_n^{t+1} - \boldsymbol{U}_n^t}{\Delta t} = \boldsymbol{f}_n^t + \boldsymbol{F}_{in,n}^t + \boldsymbol{F}_{ext,n}^t \ \ on\ \ \Gamma \tag{33}$$

where Ω_i are the source term operators including the BGK collision operator and $\mathbf{L}^t$ represents the vector of the force density on the Lagrangian point with a time step of t, that is, $\mathbf{L}^t = \left[\mathbf{F}_1^t, \mathbf{F}_2^t, \ldots, \mathbf{F}_N^t\right]^T$;similarly, we have $\overline{\mathbf{L}}^t$, $\mathbf{S}^t$, $\mathbf{I}^t$, $\mathbf{L}_s^t$, $\mathbf{E}_h^t$ and $\mathbf{T}^t$.

Compared with the traditional explicit calculation method through the iterative program, it can be seen that the present method can realize the parallel calculation of fluids and solids. For the t-th time step, the following iteration is designed in Figure 2, where $\{\boldsymbol{P}\}$ is the collection of physical quantity P at discrete points.

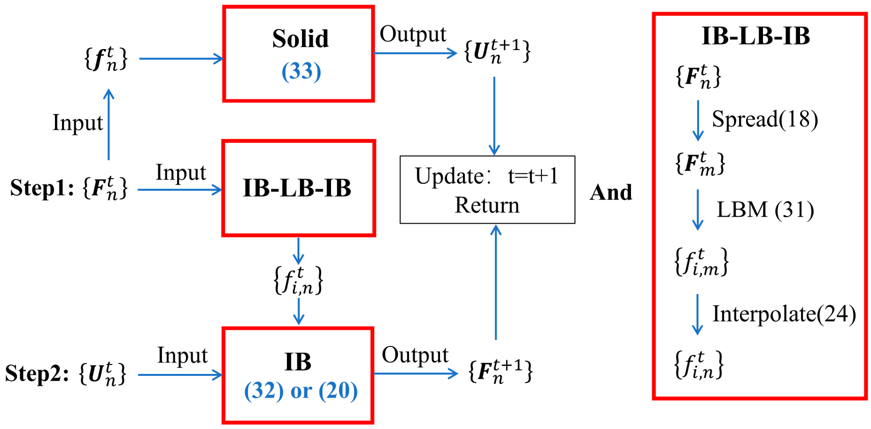


Figure 2. Flowchart of the new IB-LBM based on the proposed explicit correction force scheme.

Step 1.

IB-LB-IB part in Figure 2:

Input $\{\boldsymbol{F}_n^t\}$, and then the interpolated solid force density acting on the Euler point $\left\{\mathbf{f}_m^t\right\} = \left\{\sum_{n=1}^N \boldsymbol{F}_n^t S_{mn} \Delta s_n\right\}$; execute the LBM fluid solver (31) and output the Euler point velocity distribution function $\left\{f_{i,m}^t\right\}$ and the Lagrangian point velocity distribution function $\left\{f_{i,n}^t\right\} = \left\{\sum_{m=1}^M f_{i,m}^t I_{mn} h^d\right\}$. Then, update the macroscopic information of the flow field ρ^t, $\boldsymbol{u}^t$.,

Solid part in Figure 2:

Input $\{\boldsymbol{F}_n^t\}$, and then the fluid force density acting on the Lagrangian point $\boldsymbol{f}_n^t = -\boldsymbol{F}_n^t$; execute the solid solver (33) and output the velocity of the Lagrangian point $\{\boldsymbol{U}_n^{t+1}\}$. Update the position of the immersed boundary $\{\boldsymbol{X}_n^{t+1}\}$ at the same time.

Step 2.

IB part in Figure 2:

Input the $\left\{f_{i,n}^t\right\}$ of Step 1 and the $\{\boldsymbol{U}_n^t\}$, and then execute the explicit correction force program (32) or explicit force program (20), and output $\{\boldsymbol{F}_n^{t+1}\}$. Update t = t + 1 and return to Step 1.

For steady flows, set the threshold ε if the termination condition is met:

$$\frac{\|\boldsymbol{u}(\boldsymbol{x},t) - \boldsymbol{u}(\boldsymbol{x},t-\Delta t)\|_\infty}{\|\boldsymbol{u}(\boldsymbol{x},t)\|_\infty} < \varepsilon \tag{34}$$

For For unsteady flows, set the maximum simulation time $T = t_{max}$.

4. Results

4.1. Plane Poiseuille Flow

Poiseuille flow widely exists in industrial production and medical research and has guiding significance for the study of more complex physical problems [59].

In the following, a numerical experiment of plane Poiseuille flow is designed to verify the error and accuracy of the method proposed in this paper. The plane Poiseuille flow has the analytic solution in the velocity field which is expressed by

$$u(y) = \frac{\Delta P}{\nu}\frac{D^2}{2}\left(\frac{y}{D} - \frac{y^2}{D^2}\right) \tag{35}$$

where y is the height coordinate in the channel, D is the width of the channel, ΔP is the pressure difference inlet and outlet, and ν is the dynamic viscosity.

The maximum velocity value u_{max} is located at the center of the channel, which is calculated by

$$u_{max} = \frac{\Delta P D^2}{8\nu} \tag{36}$$

The design of the computational domain is shown in Figure 3, and the relevant parameters are dimensionless. Considering the Poiseuille flow in a rectangular domain, $\Omega = [0, L] \times [0, L]$, where $L = 20\Delta x$ and lattice length $\Delta x = 1$, the left side is the inlet and the right side is the outlet, which are set as the periodic boundary. The upper and lower sides are fixed solid-wall boundaries, which are treated by the non-equilibrium scheme. Two rigid IB boundaries with distances of $D = 13.95$ are set in the direction of the X axis. The force density of the Poiseuille flow is $\Delta \boldsymbol{P} = 1e^{-5}$. The Reynolds number $Re = 10$ and the relaxation time $\tau = 1$ remain constant. The initial numbers of Euler grids $Nx = 20$ in the X directions,$Ny = 20$ in the Y directions. The initial number of Lagrangian grids on the IB is 18 and the grid ratio is about 1.17 and the arc lengths between grids are equal. The grid numbers of four tests in each group were increased by two times compared with the previous group, namely, 20 × 20, 40 × 40, 80 × 80, 160 × 160. The number of Lagrangian grids per group was also increased by the same multiple.

Two error norms are used in the error calculation of the velocity field. The infinite norm of the error is expressed by

$$\|u^n - u^a\|_\infty = \max_{\substack{1 \le i \le Nx \\ 1 \le j \le Ny}} \left|u_{ij}^n - u_{ij}^a\right| \tag{37}$$

where u_{ij}^n, u_{ij}^a are the velocity numerical solution and the velocity analytical solution on the Euler grids, respectively.

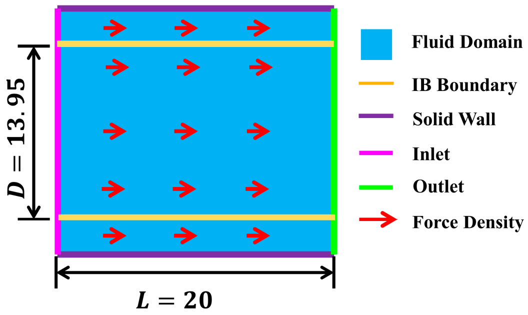


Figure 3. The design of the computational domain in 2D Poiseuille flow.

The L_2 norm of the velocity error is given by

$$\|u^n - u^a\| = \sqrt{\frac{\sum_{i=1}^{Nx}\sum_{j=1}^{Ny}\left(u_{ij}^n - u_{ij}^a\right)^2}{Nx \times Ny}} \tag{38}$$

Figure 4a is a comparison curve between the analytical solution and the numerical solution under different grid schemes in the plane Poiseuille flow. It can be seen that each numerical solution curve has a relatively high degree of fit with the analytical solution curve, which proves the effectiveness of the algorithm proposed in this paper under steady conditions. Figure 4b shows two error curves before and after matrix correction with the change of the space step. This shows that whether the matrix correction is added or not, the convergence order of the method proposed in this paper will not be changed. Additionally, the purpose of the matrix correction is to ensure the mass conservation of the interface part (Section 4.2).

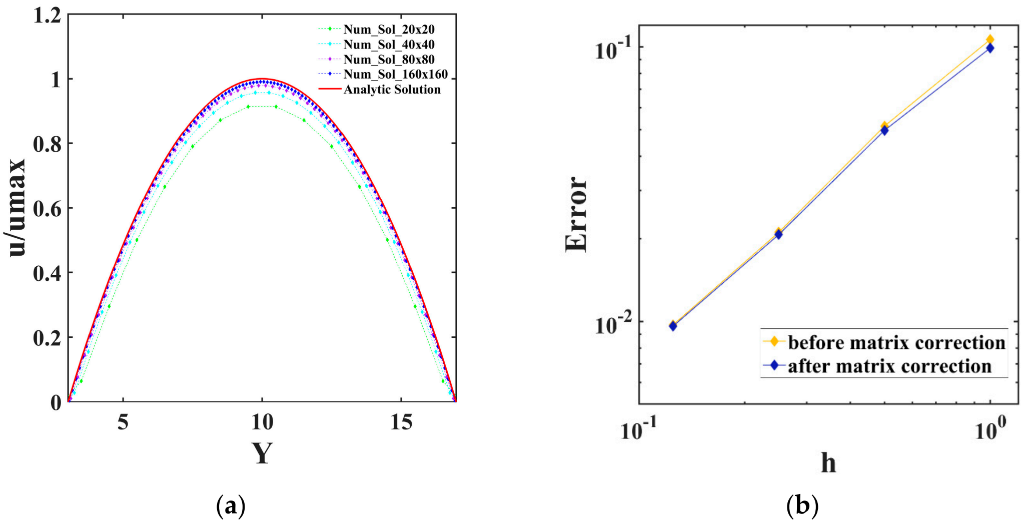


Figure 4. The numerical solutions of four cases: (**a**) compared with the analytic solution; (**b**) errors in the logarithmic scale of four cases before and after matrix correction.

Table 1 shows the specific results of the L2 norm and the infinite norm of the error under the four grid schemes. We see that the method proposed in this paper had first-order accuracy and the numerical results were consistent with the velocity analytical solution. Although LBM equations can obtain approximate second-order accuracy in space, most of the boundary treatments, such as elastic boundaries, are reduced to first-order accuracy.

Table 1. The L_2 norm, the infinite norm of the error, and the convergence orders.

Mesh	$\|u_n - u_a\|$	Rate	$\|u_n - u_a\|_\infty$	Rate
20 × 20	0.0797334		0.0858161	
40 × 40	0.0413924	0.9458183	0.0429297	0.9992722
80 × 80	0.0207219	0.9982096	0.0211221	1.0232228
160 × 160	0.00960611	1.1091321	0.00971167	1.1209620

As the computational advantage of the IB-LBMs is the excellent parallel strategy and it is reported that IB-LBMs are 23 times faster than the IBM Navier–Stokes solver [60] (though this is controversial), this explicit method can be better than IBM with the N-S solver and implicit IB-LBMs.

4.2. Flow over a Fixed Circular Cylinder

Determining flow over a circular cylinder is a classic test in fluid mechanics. A lot of experimental results and numerical results are available, so they are used to validate the IB-LBMs [38–47]. Herein, we further verify the accuracy of the IB-LBM proposed in this paper under the unsteady condition and mass conservation (the strictly satisfied no-slip boundary condition).

In this case, $Re = \frac{U_\infty D}{\nu}$, where U_∞ is the inlet velocity, D is the diameter of the cylinder, and ν is the dynamic viscosity. The drag coefficient C_D and the lift coefficient C_L are obtained by

$$C_D = \frac{F_D}{\frac{1}{2}\rho U_\infty^2 D} \tag{39}$$

$$C_L = \frac{F_L}{\frac{1}{2}\rho U_\infty^2 D} \tag{40}$$

where the drag force $F_D = -\int_\Omega f_x dxdy = \int_\Gamma F_x ds$ and the lift force $F_L = -\int_\Omega f_y dxdy = \int_\Gamma F_y ds$; here, the subscripts x and y are the x-direction and y-direction in the domain.

The Strouhal number S_t is defined to characterize the periodic effect in unsteady flow and expressed by

$$S_t = \frac{f_q D}{U_\infty} \tag{41}$$

where f_q is the vortex shedding frequency.

The whole computational domain considers a rectangular domain $\Omega = [0, 400\Delta x] \times [0, 600\Delta x]$, the lattice length $\Delta x = 1$, the cylinder diameter $D = 32\Delta x$, the number of the Lagrangian grids is 72, and the grid ratio is about 1.4. The upper and lower boundaries are fixed solid boundaries, the left is the inlet with the velocity $U_\infty = 0.2$, and the right is the outlet with the velocity gradient of zero. The non-equilibrium scheme is used to deal with the abovementioned four boundaries [61]. The Reynolds numbers were set as 20, 40, 100, and 200 respectively, and the comparison of matrix correction (30) was also considered in the numerical simulations.

Table 2 shows the calculation and comparison results of three important parameters, i.e., drag coefficient C_D, lift coefficient C_L, and Strouhal number S_t. They were compared with the results of the experiment [62], other numerical methods [63], and other IB-LBMs [35,45]. The results showed that the numerical method proposed in this paper was almost identical to other results. This shows that the calculation accuracy of the method proposed in this paper is good and the numerical results under unsteady flow are therefore verified.

Table 2. Comparison of the results on lift coefficient, drag coefficient, and Strohal number.

Parameter	Reference	Re = 200	Re = 100	Re = 40	Re = 20
C_D	Tritton [62]			1.48	2.22
	Calhoun [63]	1.17	1.33	1.62	2.19
	Hu [45]	1.394	1.418	1.660	2.213
	This study	1.471	1.518	1.755	2.335
C_L	Wu [35]		0.344		
	Calhoun [63]	0.67	0.298		
	Hu [45]	0.712	0.367		
	This study	0.711	0.3609		
St	Wu [35]	0.197	0.163		
	Calhoun [63]	0.202	0.175		
	Hu [45]	0.195	0.166		
	This study	0.216	0.185		

Figure 5a shows the vortex contours under four Reynolds numbers. It can be seen that they are basically consistent with the experimental phenomenon. When Re > 47, the flow separated and the unsteady flow phenomenon occurred. The Karman vortex street phenomenon was clearly visible at Re = 100 and Re = 200 and the vortex separation frequency increased with the increase of Reynolds number. Figure 5b shows two sets of drag coefficient curves and two sets of lift coefficient curves for Re = 100 and Re = 200 which correspond to the vortex contours of the same Reynolds number in Figure 6a. We see that as the Reynolds number increased, the vortex separation was obvious, which is reflected in the increase of the period numbers of the lift coefficient within a limited time. Its specific values are explained in Table 2. Figure 5a,b show that this method is accurate and effective in simulating unsteady flow.

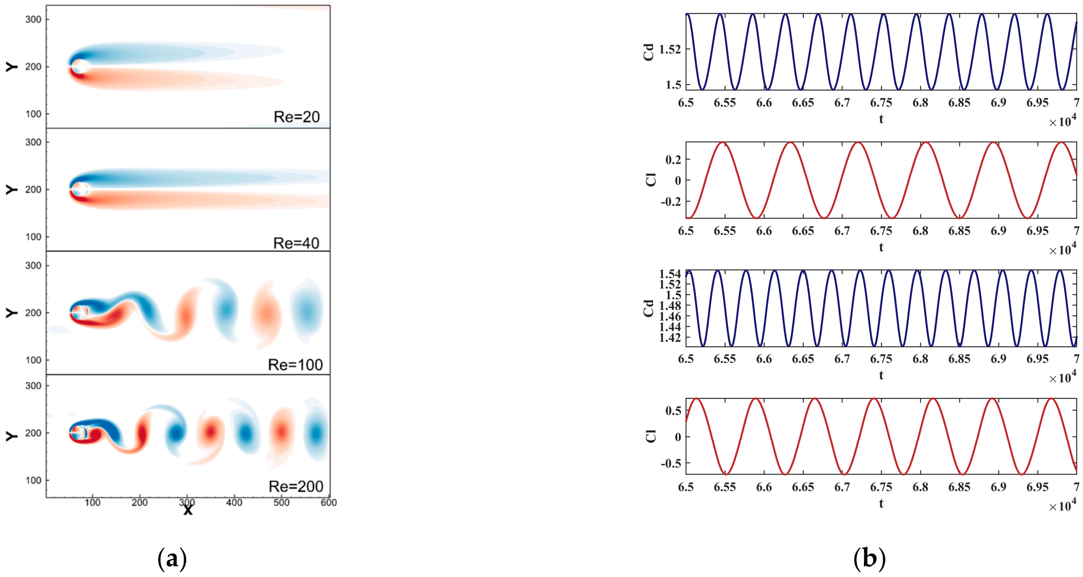
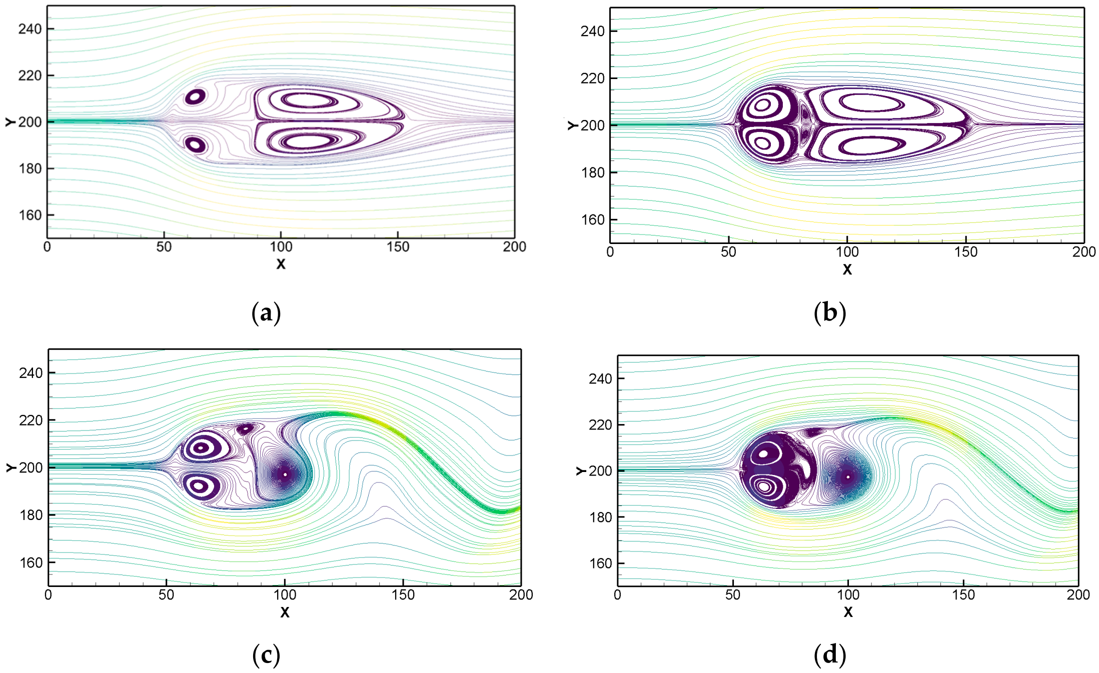


Figure 5. The results of different Reynolds numbers: (**a**) the vortex contours; and (**b**) the curves of lift coefficient Cl and drag coefficient Cd with Re = 100 and Re = 200.

Figure 6 shows the streamline diagrams of Section 4.2 under Re = 100 and Re = 200. In the case of Re = 40 (steady flow), as shown in Figure 6a without correction, there occurred the penetration of some streamlines inside the cylinder, which also occur in the previous explicit method and the conservation of mass cannot be guaranteed in this region. After the calculation of the modified Equation (30), the streamline became clear and no longer penetrated, which ensured the conservation of mass and satisfied the no-slip boundary condition. The same successful correction effect can be obtained from the streamline comparison of Figure 6c,d; that is to say, the streamline correction scheme proposed in this paper is applicable to unsteady flow.



(**a**) (**b**)

(**c**) (**d**)

Figure 6. Comparison of the effect of matrix correction on the streamline penetration phenomenon at different Reynolds numbers: (**a**) Re = 40 without correction; (**b**) Re = 40 with correction; (**c**) Re = 200 without correction; and (**d**) Re = 200 with correction.

4.3. Free Oscillation of the Flapping Foil NACA0012

The oscillation analysis of airfoil is an important subject of airfoil aerodynamic characteristics. The relevant studies are the prime motivators for the new technology in this regard, such as marine, aerial, and energy territory [64]. Using numerical methods to simulate and analyze relevant experimental conditions, such as wake state, has become an effective research method at present [65].

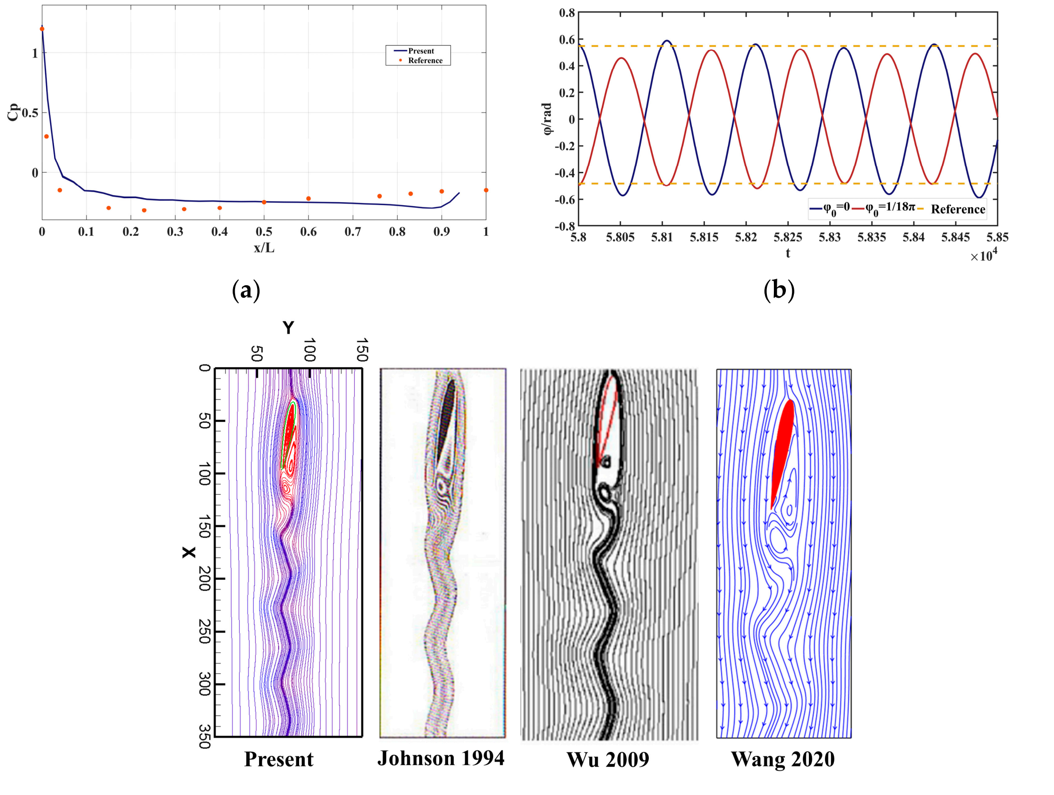
In this case, the two-dimensional airfoil NACA0012 was used and the chord length of the airfoil was L. The aerodynamic center $\boldsymbol{r_c}$ was fixed, which was located at the center line of the foil with the distance $L_c = 0.446L$ from the leading edge. Therefore, the airfoil is a movable, rigid body with one degree of freedom; the same settings can be seen in ref. [40]. The airfoil rotationally oscillated about the fixed point $\boldsymbol{r_c}$. The dynamic equation of the airfoil was an ordinary differential equation (ODE).

$$\int_L \boldsymbol{F_f} \times (\boldsymbol{X} - \boldsymbol{r_c}) \cdot e_z d_s = I_s \frac{\partial^2 \varphi}{\partial t^2} - C_s \frac{\partial \varphi}{\partial t} + K_s \varphi \tag{42}$$

where φ is the angle between the chord length and the horizontal direction, I_s is the moment of inertia, C_s is the damping coefficient, and K_s is the stiffness coefficient.

The numerical domain was a rectangular domain $\Omega = [0, 360\Delta x] \times [0, 1200\Delta x]$ and the lattice length was $\Delta x = 1$. The airfoil chord length was $L = 38\Delta x$ and the number of the Lagrangian grids was 72. The left side was the inlet and the inlet velocity was $U_\infty = 0.05$. The right side was the outlet and the velocity gradient was 0. The four boundaries were all processed in the non-equilibrium scheme. The Reynolds number was $Re = \frac{U_\infty L}{\nu}$ and the initial angle $\varphi_0 = 0$. Lift and drag coefficients of the airfoil were calculated using Equations (39) and (40). The ODE solution of the foil was solved by the conventional difference method and coupled with IB-LBM for the final solution. Four Reynolds numbers Re = 200, 500, 1000, and 2000 were used for numerical simulation.

Figure 7a shows the pressure coefficient curve along the edge of the airfoil when Re = 500, compared with ref. [66]. The pressure coefficient dropped rapidly from the center of the leading edge to a negative value and then approached 0, which also conforms to the changing characteristics of the NACA0012 airfoil standard pressure curve. Figure 7b shows a comparison of the periodic rotation angle curves of the airfoil when Re = 1000. It can be seen that the maximum angle was comparable to the vibration amplitude of ref. [40], and the maximum angle was about 0.6 rad. Similar to ref. [40], when changing the initial angle, the maximum rotation angle changed. Excessive maximum rotation angle in this case caused the program code to crash. Figure 7c shows the streamline comparison at $\frac{1}{4}$ T when Re = 1000 and the attack angle was 0.175 rad. In the simulation in this paper, the streamline had no penetration phenomenon. Compared with [35,48,67], the streamline results in this paper were similar to those in ref. [67]. The Strouhal number given in this paper was 0.855, which is similar to the 0.86 used in the literature [67].



(**a**)

(**b**)

(**c**)

Figure 7. Comparison of the results of other papers: (**a**) pressure coefficient curve (Re = 500); (**b**) amplitude curve (Re = 1000); and (**c**) streamline (Re = 1000) [35,48,67].

Figure 8 shows a comparison of the rotation angles at four Reynolds numbers. We see that the maximum rotation angle increased with the increase of the Reynolds number. When the rotation angle was too large, that is, the Reynolds number exceeded 4000, the numerical simulation failed. When Re = 200 and 500, the change of rotation angle was regular, and the rotation maximum amplitude was basically unchanged. However, when Re = 1000 and 2000, the rotation range increased and is related to the flow instability at a larger Reynolds number.

Figure 9 shows a comparison of the vortex contours of at four Reynolds numbers. It can be seen that when Re = 200, the flow was a laminar flow; when Re = 500, the flow became an unsteady flow, presenting a regular Karman vortex street; when Re = 1000, the downstream vortex shedding became unstable and presented a complex wake shape. Compared with Re = 1000, the unstable vortex shedding occurred earlier at Re = 2000, and this instability was more serious in the middle and later stages of the wake. When Re = 1000, 2000, we found that at the vertical line of the wake with the same horizontal position, vortex eyes appeared two to three times. This unstable vortex shedding is a common phenomenon and can be simulated by improving the model to realize the dual Karman vortex street [65].

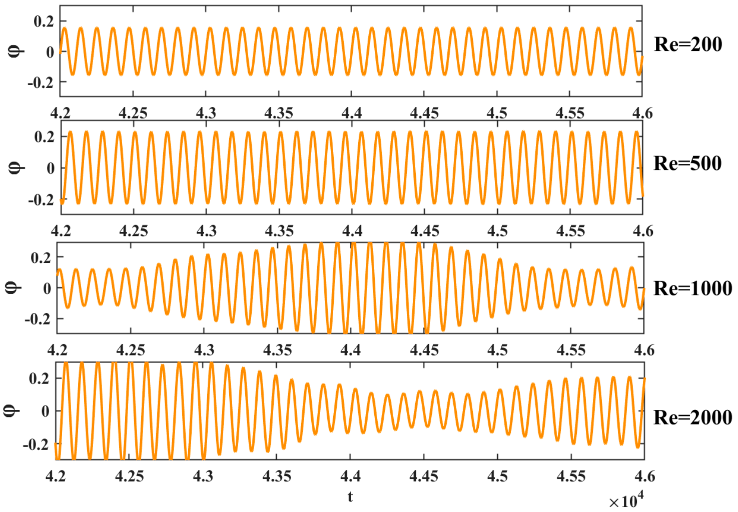


Figure 8. Rotation angle curves at four Reynolds numbers.

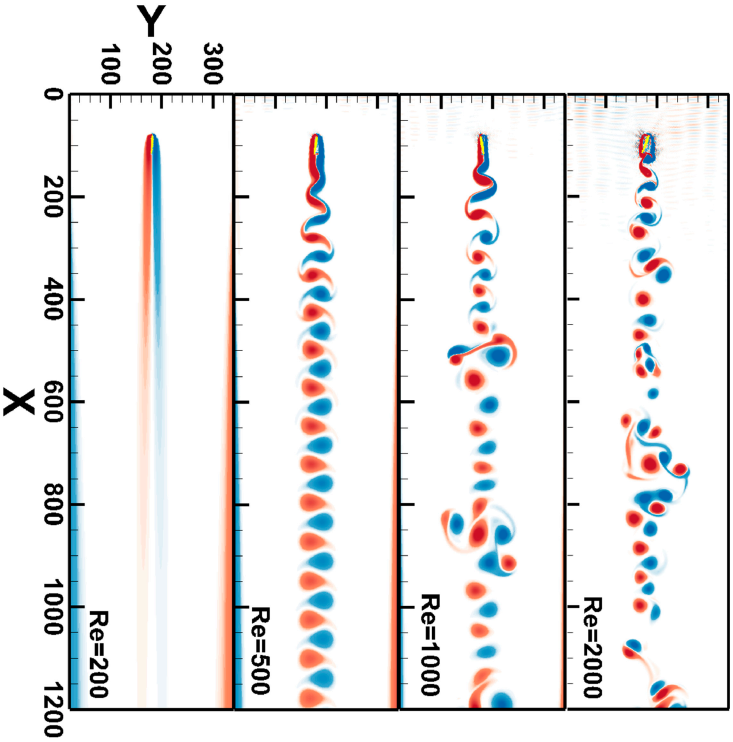


Figure 9. The vortex contours at four Reynolds numbers.

4.4. A Flow Past Free Deformation of Cylinders

The deformation of the cylinders with flow around them constitutes an abstract model of practical engineering. In the numerical model of porous media, the skeleton model is often abstracted as cylinders around flow, so the research on the pore deformation of porous media can be transformed into the deformation of the cylinder group [68]. Additionally, changing the geometric parameters and solid force model, the cylinder deformation can provide a reference model in biology, such as in the deformation calculation of red blood cells [69].

The general deformation equation of an enclosed body is expressed by

$$M\frac{\partial^2 \boldsymbol{X}}{\partial t^2} = \boldsymbol{F}_{fluid} + \boldsymbol{F}_{inner} = \boldsymbol{F}_{fluid} + \boldsymbol{F}_T + \boldsymbol{F}_N \tag{43}$$

where M is the mass ratio, $\boldsymbol{F}_{fluid}$ is the external force of the fluid acting on the solid, $\boldsymbol{F}_T$ is the internal force in the tangential direction, and $\boldsymbol{F}_N$ is the internal force in the normal direction. For $\boldsymbol{F}_T$, $\boldsymbol{F}_N$, this paper provides a force model that is proportional to the deformation which satisfies Newton's law, and the sum of internal forces at all points is 0.

Adding the stretch coefficient K_s and the bending coefficient K_b and using the finite difference method, the specific deformation difference equation can be obtained by

$$M\frac{\boldsymbol{X}_n^{t+1} - 2\boldsymbol{X}_n^t + \boldsymbol{X}_n^{t-1}}{\partial t^2} = \boldsymbol{F}\left(X_n^t, t\right) + \boldsymbol{F}_T^n + \boldsymbol{F}_N^n \tag{44}$$

Then,

$$\boldsymbol{F}_T^n\left(\boldsymbol{X}_n^t, t\right) = K_s \cdot \sum_{k=n-1}^{n} A_k\left(\Delta s_k^t - \Delta s_k^0\right)\boldsymbol{\tau}_k^t \tag{45}$$

$$\boldsymbol{F}_N^n\left(\boldsymbol{X}_n^t, t\right) = K_b \cdot \sum_{k=n-1}^{n+1} B_k\left(\Delta \varphi_k^t - \Delta \varphi_k^0\right)\boldsymbol{N}_k^t \tag{46}$$

where $\boldsymbol{\tau}_n^t = \frac{\boldsymbol{X}_{n+1}^t - \boldsymbol{X}_n^t}{\Delta s}$ is the unit tangent vector and $\boldsymbol{N}_n^t$ is the unit normal vector orthogonal to the vector $\left(\boldsymbol{X}_{n+1}^t - \boldsymbol{X}_{n-1}^t\right)$. The angle $\Delta\varphi$ can be obtained from the relevant vectors.

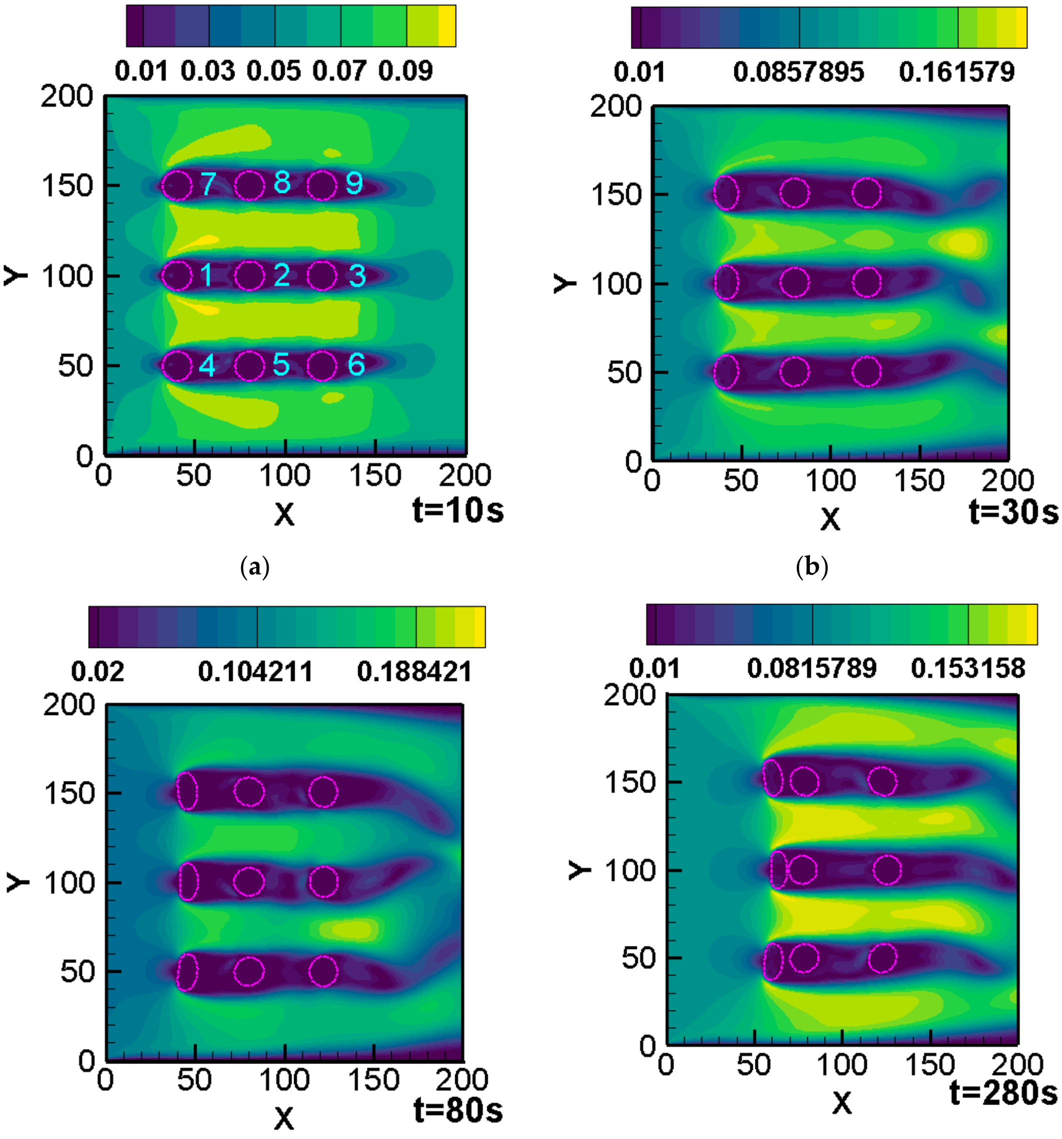
Mark $\boldsymbol{X}_{a,b} = \boldsymbol{X}_a^t - X_b^t$; then, the coefficients A_k and B_k can be expressed by

$$A_k = \begin{cases} 1, & k = n \\ -1, & k = n-1 \end{cases} \tag{47}$$

$$B_k = \begin{cases} \frac{\left|\boldsymbol{X}_{n,n-2} - \boldsymbol{X}_{n,n-1}\right|}{\left(\left|\boldsymbol{X}_{n,n-2} - \boldsymbol{X}_{n-1,n-2}\right| + \left|\boldsymbol{X}_{n,n-2} - \boldsymbol{X}_{n,n-1}\right|\right)}, k = n-1 \\ 1,\ k = n \\ \frac{\left|\boldsymbol{X}_{n+2,n} - \boldsymbol{X}_{n+1,n}\right|}{\left(\left|\boldsymbol{X}_{n+2,n} - \boldsymbol{X}_{n+2,n+1}\right| + \left|\boldsymbol{X}_{n+2,n} - \boldsymbol{X}_{n+1,n}\right|\right)}, k = n+1 \end{cases} \tag{48}$$

The computational domain was a rectangular area $\Omega = [0, 2L] \times [0, L]$, $L = 200\Delta x$, the grid length was $\Delta x = 1$, and the time step was $\Delta t = 10^{-3}s$. The left side was the inlet and the inlet velocity was $U_\infty = 0.1$. The right side was the outlet and the velocity gradient was 0. The four boundaries were processed as the non-equilibrium scheme. Choosing deformable 2D cylinders, the diameter of each cylinder was $16\Delta x$ and the number of Lagrangian points was 36. The center of the first cylinder was $(40\Delta x, 50\Delta x)$; a cylinder was arranged every $40\Delta x$ in the X direction and every $50\Delta x$ in the Y direction (see Figure 10a for specific labels of nine cylinders). The Reynolds number was $Re = 400$, the maximum time steps setting was 240,000, and the calculation stopped if cylinders collided.

Figure 10 shows the velocity contour of flow field and deformation process of nine cylinders. The numerical domain was selected as $200\Delta x \times 200\Delta x$. Comparing Figure 10a–d, it can be seen that the degrees of deformation and moving speeds of the cylinders (1,4,7) on the first column were greater than those of the second column, and similarly, the deformation and speed of the second column were greater than those of the third column. It can be seen that the cylinders on both sides (7,8,9 and 4,5,6) basically changed symmetrically, and the degree of deformation was smaller than that of the middle cylinder. Looking at the cylinders on the first column, cylinder 1 approached the rear cylinder earlier than cylinders 4 and 7. The velocity contours of the near-wake region behind the cylinder show that the attenuation of the wake velocity is the main reason for the different degrees of deformation.

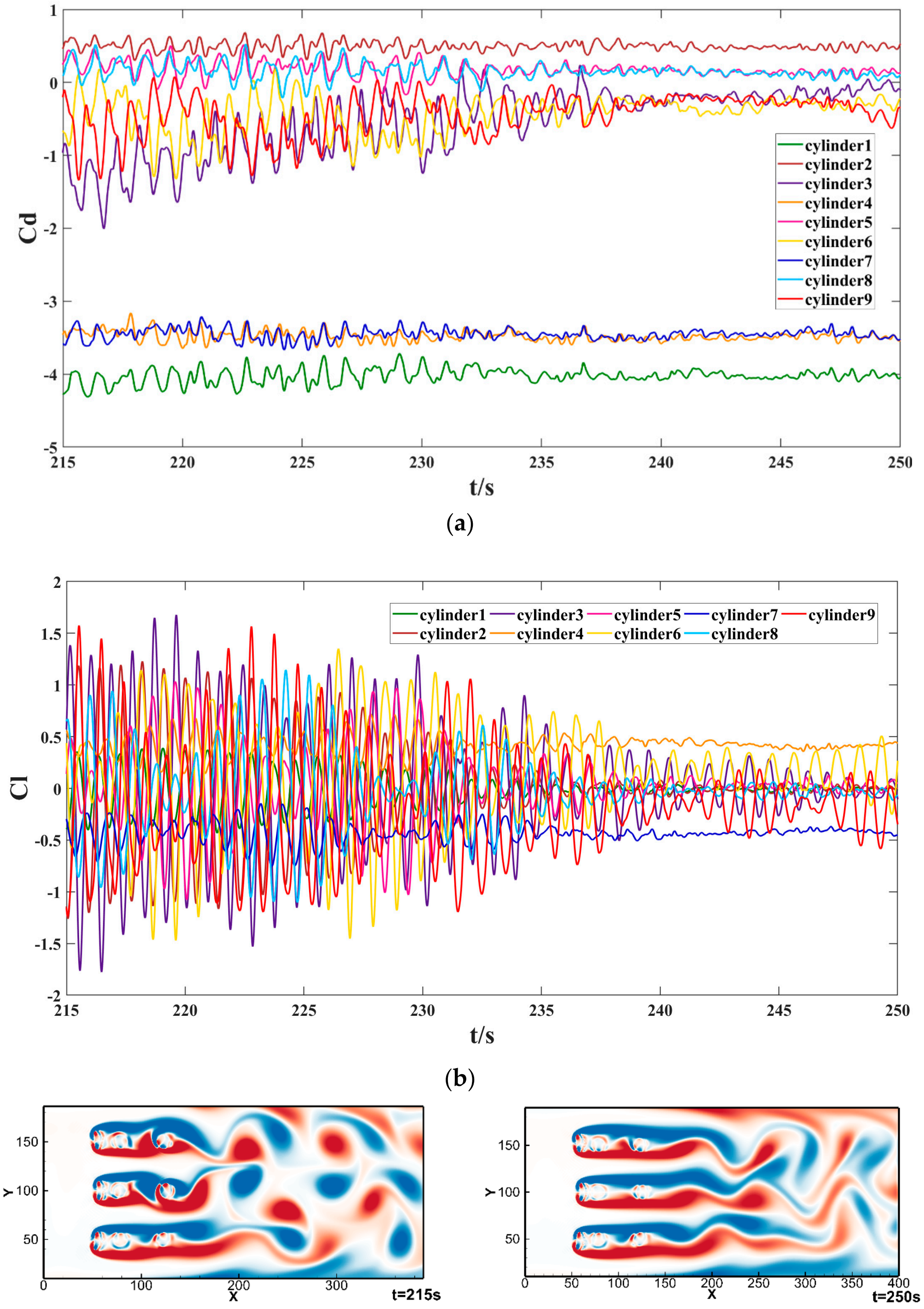


(**a**) (**b**)

(**c**) (**d**)

Figure 10. Deformation and velocity diagrams at four times: (**a**) 10 s; (**b**) 30 s; (**c**) 80 s; and (**d**) 280 s.

Figure 11a,b are the lift and drag coefficient curves with time from 215 s to 250 s. It can be seen that the overall drag coefficient and lift coefficient decreased during this time period, especially for the third column (3, 6, 9). Due to geometric deformation, we think that the degree of attenuation in the near-wake region increased, which resulted in less drag force and thus less flow separation. The attenuation of the lift coefficient curve Figure 11b means the attenuation of the degree of flow separation. These inferences can be reflected from the vortex contours in Figure 11c,d and the flow separation of Figure 11c is greater than that of Figure 11d.



(**a**)

(**b**)

(**c**) (**d**)

Figure 11. Diagram for deformation from 215 s to 250 s: (**a**) drag coefficient curve; (**b**) lift coefficient curve; (**c**) vortex contours at 215 s; and (**d**) vortex contours at 255 s.

5. Conclusions

We summarized the content of the article and divided it into five points, as follows:

(1) We first obtained an explicit force with a simple form by assuming the existing LBM force model on IB in Equation (20). At present, the two most widely used explicit forces are a direct force proposed by Feng [33] and a bounce-back force proposed by Niu [44]. Feng's direct force is similar to the stress integration method in LBM, and usually needs to calculate a complex stress tensor as in Equation (15); Niu's

bounce-back force is a combination of the bounce-back scheme and the momentum exchange method, and requires two steps: first, obtain the bounce-back distribution function (see [44] and a half-way bounce-back scheme optimized by Wang 2020 [48]), and then apply the momentum exchange method to obtain the interface force as in Equation (16). In contrast, our explicit force based on interpolating the distribution function and imposing the force model on the IB boundary can be realized in only one step, and the form is simple. Numerical experiments proved our force method and assumptions are accurate and effective;

(2) Then, we obtained the total explicit correction force scheme in Equation (30) by the correction matrix, which is an explicit scheme, only needing to add a matrix to Equation (20). The correction matrix T is only determined by the geometric position updated at time $t - 1$, and the implicit scheme needs to solve the equation at time t. From the results of the clear streamline in Figure 6, the present scheme can well guarantee the local mass conservation, that is, almost satisfy the no-slip boundary condition, especially for the unsteady flow in Figure 6d and the large curvature boundary condition in Figure 7c. The correction matrix can be obtained by the LU decomposition method or some iterative methods;

(3) The proposed explicit correction force scheme has two modes for the user to choose: the first mode is Equation (11) and the second mode is Equation (11) and Equation (12) to obtain the interface force. When the user conducts a large number of grids or three-dimensional conditions for numerical simulation, the first mode can be selected to ensure acceptable simulation time. For medium-scale simulation, we recommend the full explicit correction force scheme to ensure clear results at the interface;

(4) For the accuracy order, we have the same order as the general IBM or IB-LBM, that is, the first-order accuracy order in Table 1. However, compared with the IBM by N-S solver, LBM has an excellent parallel strategy and local grid refinement strategy, which helps obtaining numerical simulation results with smaller errors;

(5) The proposed scheme performed well on complex boundaries such as moving boundaries in Section 4.3 and flexible boundaries in Section 4.4. Although good simulation results can be achieved on flexible boundaries by improving traditional implicit methods, explicit methods have traditional advantages in simulating complex boundaries. The proposed reference models in Sections 4.3 and 4.4 have a background in the relevant discipline or engineering, and researchers can improve the proposed reference model to achieve good simulation results in specific cases.

Author Contributions: Conceptualization, W.S. and B.L.; methodology, B.L.; software, B.L.; validation, B.L. and W.S.; formal analysis, W.S.; investigation, B.L.; resources, B.L.; data curation, B.L.; writing—original draft preparation, B.L.; writing—review and editing, B.L. and W.S.; visualization, B.L.; supervision, W.S.; project administration, W.S.; funding acquisition, W.S. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Key Research and Development Program of China (Grant No. 2018YFA0703300) and the Natural Science Foundation of Jilin Province, China (Grant no. 20200201278JC).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: Chengchun Zhang from Key Laboratory of Bionic Engineering at Jilin University hosts the National Key Research and Development Program of China (Grant No. 2018YFA0703300), we thank his management work and financial support for this article.

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

References

1. Van Brummelen, E.H.; Geuzaine, P.J.B. Fundamentals of fluid-structure interaction. *Encycl. Aerosp. Eng.* **2010**, *153*, 1–7. [CrossRef]
2. Wang, Y.; Shu, C.; Teo, C.J.; Wu, J. An immersed boundary-lattice Boltzmann flux solver and its applications to fluid–structure interaction problems. *J. Fluids Struct.* **2015**, *54*, 440–465. [CrossRef]
3. Ma, J.; Wang, Z.; Young, J.; Lai, J.C.; Sui, Y.; Tian, F.B. An immersed boundary-lattice Boltzmann method for fluid-structure interaction problems involving viscoelastic fluids and complex geometries. *J. Comput. Phys.* **2020**, *415*, 109487. [CrossRef]
4. Liu, F.; Liu, G.; Shu, C. Fluid–structure interaction simulation based on immersed boundary-lattice Boltzmann flux solver and absolute nodal coordinate formula. *Phys. Fluids* **2020**, *32*, 047109. [CrossRef]
5. Karimnejad, S.; Delouei, A.A.; Başağaoğlu, H.; Nazari, M.; Shahmardan, M.; Falcucci, G.; Succi, S. A Review on Contact and Collision Methods for Multi-Body Hydrodynamic Problems in Complex Flows. *Commun. Comput. Phys.* **2022**, *32*, 899–950. [CrossRef]
6. Le Tallec, P.; Mouro, J. Fluid structure interaction with large structural displacements. *Comput. Methods Appl. Mech. Eng.* **2001**, *190*, 3039–3067. [CrossRef]
7. Hirt, C.W.; Amsden, A.A. An arbitrary Lagrangian-Eulerian computing method for all flow speeds. *J. Comput. Phys.* **1974**, *14*, 227–253. [CrossRef]
8. Souli, M.; Ouahsine, A. ALE formulation for fluid–structure interaction problems. *Comput. Methods Appl. Mech. Eng.* **2000**, *190*, 659–675. [CrossRef]
9. Peskin, C.S. Flow patterns around heart valves: A numerical method. *J. Comput. Phys.* **1972**, *10*, 252–271. [CrossRef]
10. Sotiropoulos, F.; Yang, X. Immersed boundary methods for simulating fluid–structure interaction. *Prog. Aerosp. Sci.* **2014**, *65*, 7821–7836. [CrossRef]
11. Yang, X.; Zhang, X.; Li, Z. A smoothing technique for discrete delta functions with application to immersed boundary method in moving boundary simulations. *J. Comput. Phys.* **2009**, *228*, 7821–7836. [CrossRef]
12. Bao, Y.; Kaye, J.; Peskin, C.S. A Gaussian-like immersed-boundary kernel with three continuous derivatives and improved translational invariance. *J. Comput. Phys.* **2016**, *316*, 139–144. [CrossRef]
13. Fadlun, E.A.; Verzicco, R.; Orlandi, P.; Mohd-Yusof, J.C.D. Combined immersed-boundary finite-difference methods for three-dimensional complex flow simulations. *J. Comput. Phys.* **2000**, *161*, 35–60. [CrossRef]
14. Yang, J.; Balaras, E. An embedded-boundary formulation for large-eddy simulation of turbulent flows interacting with moving boundaries. *J. Comput. Phys.* **2006**, *215*, 12–40. [CrossRef]
15. Griffith, B.E.; Patankar, N.A. Immersed methods for fluid–structure interaction. *Annu. Rev. Fluid Mech.* **2020**, *52*, 421–448. [CrossRef] [PubMed]
16. McNamara, G.R.; Zanetti, G. Use of the Boltzmann equation to simulate lattice-gas automata. *Phys. Rev. Lett.* **1988**, *61*, 2332. [CrossRef] [PubMed]
17. Benzi, R.; Succi, S.; Vergassola, M. The lattice Boltzmann equation: Theory and applications. *Phys. Rep.* **1992**, *222*, 145–197. [CrossRef]
18. Succi, S. *The Lattice Boltzmann Equation: For Fluid Dynamics and Beyond*, 1st ed.; Oxford University Press: New York, NY, USA, 2001; pp. 10–255. ISBN 0-19-850398-9.
19. Krüger, T.; Kusumaatmaja, H.; Kuzmin, A.; Shardt, O. *The Lattice Boltzmann Method: Principles and Practice*, 1st ed.; Springer International Publishing: Cham, Switzerland, 2017; pp. 22–684. ISBN 978-3-319-44649-3.
20. Bouzidi, M.H.; Firdaouss, M.; Lallemand, P. Momentum transfer of a Boltzmann-lattice fluid with boundaries. *Phys. Fluids* **2001**, *13*, 3452–3459. [CrossRef]
21. Guo, Z.; Zheng, C.; Shi, B. An extrapolation method for boundary conditions in lattice Boltzmann method. *Phys. Fluids* **2002**, *14*, 2007–2010. [CrossRef]
22. Ladd, A.J. Numerical simulations of particulate suspensions via a discretized Boltzmann equation. Part 1. Theoretical foundation. *J. Fluid Mech.* **1994**, *271*, 285–309. [CrossRef]
23. Nguyen, N.Q.; Ladd, A.J.C. Lubrication corrections for lattice-Boltzmann simulations of particle suspensions. *Phys. Rev. E* **2002**, *66*, 046708. [CrossRef] [PubMed]
24. Ding, E.-J.; Aidun, C.K. Extension of the lattice-Boltzmann method for direct simulation of suspended particles near contact. *J. Stat. Phys.* **2003**, *112*, 685–708. [CrossRef]
25. Başağaoğlu, H.; Succi, S.; Wyrick, D.; Blount, J. Particle shape influences settling and sorting behavior in microfluidic domains. *Sci. Rep.* **2018**, *8*, 8583. [CrossRef]
26. Motta, J.B.; Costa, P.; Derksen, J.J.; Peng, C.; Wang, L.P.; Breugem, W.P.; Renon, N. Assessment of numerical methods for fully resolved simulations of particle-laden turbulent flows. *Comput. Fluids* **2019**, *179*, 1–14. [CrossRef]
27. Peng, Y.; Luo, L.S. A comparative study of immersed-boundary and interpolated bounce-back methods in LBE. *Prog. Comput. Fluid Dyn. Int. J.* **2008**, *8*, 156–167. [CrossRef]
28. Saurabh, K.; Solovchuk, M.; Sheu, T.W.H. Investigating ion transport inside the pentameric ion channel encoded in COVID-19 E protein. *Phys. Rev. E* **2020**, *102*, 052408. [CrossRef] [PubMed]
29. Afra, B.; Delouei, A.A.; Mostafavi, M.; Tarokh, A. Fluid-structure interaction for the flexible filament's propulsion hanging in the free stream. *J. Mol. Liq.* **2021**, *323*, 114941. [CrossRef]

30. Afra, B.; Karimnejad, S.; Delouei, A.A.; Tarokh, A. Flow control of two tandem cylinders by a highly flexible filament: Lattice spring IB-LBM. *Ocean Eng.* **2022**, *250*, 111025. [CrossRef]
31. Cong, L.; Teng, B.; Bai, W.; Chen, B. A VOS based Immersed Boundary-Lattice Boltzmann method for incompressible fluid flows with complex and moving boundaries. *Comput. Fluids* **2023**, *255*, 105832. [CrossRef]
32. Feng, Z.G.; Michaelides, E.E. The immersed boundary-lattice Boltzmann method for solving fluid–particles interaction problems. *J. Comput. Phys.* **2004**, *195*, 602–628. [CrossRef]
33. Feng, Z.G.; Michaelides, E.E. Proteus: A direct forcing method in the simulations of particulate flows. *J. Comput. Phys.* **2005**, *202*, 20–51. [CrossRef]
34. Feng, Z.G.; Michaelides, E.E. Robust treatment of no-slip boundary condition and velocity updating for the lattice-Boltzmann simulation of particulate flows. *Comput. Fluids* **2009**, *38*, 370–381. [CrossRef]
35. Wu, J.; Shu, C. Implicit velocity correction-based immersed boundary-lattice Boltzmann method and its applications. *J. Comput. Phys.* **2009**, *228*, 1963–1979. [CrossRef]
36. Wu, J.; Shu, C.; Zhang, Y.H. Simulation of incompressible viscous flows around moving objects by a variant of immersed boundary-lattice Boltzmann method. *Int. J. Numer. Methods Fluids* **2010**, *62*, 327–354. [CrossRef]
37. Wu, J.; Shu, C. Simulation of three-dimensional flows over moving objects by an improved immersed boundary–lattice Boltzmann method. *Int. J. Numer. Methods Fluids* **2012**, *68*, 977–1004. [CrossRef]
38. Kang, S.K.; Hassan, Y.A. A comparative study of direct-forcing immersed boundary-lattice Boltzmann methods for stationary complex boundaries. *Int. J. Numer. Methods Fluids* **2011**, *66*, 1132–1158. [CrossRef]
39. Seta, T.; Rojas, R.; Hayashi, K.; Tomiyama, A. Implicit-correction-based immersed boundary–lattice Boltzmann method with two relaxation times. *Phys. Rev. E* **2014**, *89*, 023307. [CrossRef]
40. Wang, W.Q.; Yan, Y.; Liu, G.R. An IB-LBM implementation for fluid-solid interactions with an MLS approximation for implicit coupling. *Appl. Math. Model.* **2018**, *62*, 638–653. [CrossRef]
41. Gsell, S.; Favier, J. Direct-forcing immersed-boundary method: A simple correction preventing boundary slip error. *J. Comput. Phys.* **2021**, *435*, 110265. [CrossRef]
42. Peng, C.; Wang, L.P. Force-amplified, single-sided diffused-interface immersed boundary kernel for correct local velocity gradient computation and accurate no-slip boundary enforcement. *Phys. Rev. E* **2020**, *101*, 053305. [CrossRef]
43. Afra, B.; Nazari, M.; Kayhani, M.H.; Delouei, A.A.; Ahmadi, G. An immersed boundary-lattice Boltzmann method combined with a robust lattice spring model for solving flow–structure interaction problems. *Appl. Math. Model.* **2018**, *55*, 502–521. [CrossRef]
44. Niu, X.D.; Shu, C.; Chew, Y.T. A momentum exchange-based immersed boundary-lattice Boltzmann method for simulating incompressible viscous flows. *Phys. Lett. A* **2006**, *354*, 173–182. [CrossRef]
45. Hu, Y.; Yuan, H.; Shu, S.; Niu, X.; Li, M. An improved momentum exchanged-based immersed boundary–lattice Boltzmann method by using an iterative technique. *Comput. Math. Appl.* **2014**, *68*, 140–155. [CrossRef]
46. Yuan, H.Z.; Niu, X.D.; Shu, S.; Li, M.; Yamaguchi, H. A momentum exchange-based immersed boundary-lattice Boltzmann method for simulating a flexible filament in an incompressible flow. *Comput. Math. Appl.* **2014**, *67*, 1039–1056. [CrossRef]
47. Tao, S.; Chen, B.; Yang, X.; Huang, S. Second-order accurate immersed boundary-discrete unified gas kinetic scheme for fluid-particle flows. *Comput. Fluids* **2018**, *165*, 54–63. [CrossRef]
48. Wang, Z.; Wei, Y.; Qian, Y. A bounce back-immersed boundary-lattice Boltzmann model for curved boundary. *Appl. Math. Model.* **2020**, *81*, 428–440. [CrossRef]
49. Peng, C.; Ayala, O.M.; Wang, L.P. A comparative study of immersed boundary method and interpolated bounce-back scheme for no-slip boundary treatment in the lattice Boltzmann method: Part I, laminar flows. *Comput. Fluids* **2019**, *192*, 104233. [CrossRef]
50. Uhlmann, M. An immersed boundary method with direct forcing for the simulation of particulate flows. *J. Comput. Phys.* **2005**, *209*, 448–476. [CrossRef]
51. Bhatnagar, P.L.; Gross, E.P.; Krook, M. A model for collision processes in gases. I. Small amplitude processes in charged and neutral one-component systems. *Phys. Rev.* **1954**, *94*, 511. [CrossRef]
52. He, X.; Luo, L.S. Theory of the lattice Boltzmann method: From the Boltzmann equation to the lattice Boltzmann equation. *Phys. Rev. E* **1997**, *56*, 6811. [CrossRef]
53. Guo, Z.; Zheng, C. Discrete lattice effects on the forcing term in the lattice Boltzmann method. *Phys. Rev. E* **2002**, *65*, 046308. [CrossRef] [PubMed]
54. Qian, Y.H.; d'Humières, D.; Lallemand, P. Lattice BGK models for Navier-Stokes equation. *Europhys. Lett.* **1992**, *17*, 479. [CrossRef]
55. Peskin, C.S. Numerical analysis of blood flow in the heart. *J. Comput. Phys.* **1972**, *25*, 220–252. [CrossRef]
56. Peskin, C.S. The immersed boundary method. *Acta Numer.* **2002**, *11*, 479–517. [CrossRef]
57. Cao, N.; Chen, S.; Jin, S.; Martinez, D. Physical symmetry and lattice symmetry in the lattice Boltzmann method. *Phys. Rev. E* **1997**, *55*, R21. [CrossRef]
58. He, X.; Luo, L.S.; Dembo, M. Some progress in lattice Boltzmann method. Part I. Nonuniform mesh grids. *J. Comput. Phys.* **1996**, *129*, 357–363. [CrossRef]
59. Stuart, J.T. On the non-linear mechanics of wave disturbances in stable and unstable parallel flows Part 1. The basic behaviour in plane Poiseuille flow. *J. Fluid Mech.* **1960**, *9*, 353–370. [CrossRef]
60. Su, S.W.; Lai, M.C.; Lin, C.A. An immersed boundary technique for simulating complex flows with rigid boundary. *Comput. Fluids* **2007**, *36*, 313–324. [CrossRef]

61. Zhao-Li, G.; Chu-Guang, Z.; Bao-Chang, S. Non-equilibrium extrapolation method for velocity and pressure boundary conditions in the lattice Boltzmann method. *Chin. Phys.* **2002**, *11*, 366. [CrossRef]
62. Tritton, D.J. Experiments on the flow past a circular cylinder at low Reynolds numbers. *J. Fluid Mech.* **1959**, *9*, 547–567. [CrossRef]
63. Calhoun, D.A. Cartesian grid method for solving the two-dimensional stream function-vorticity equations in irregular regions. *J. Comput. Phys.* **2002**, *176*, 231–275. [CrossRef]
64. Wu, X.; Zhang, X.; Tian, X.; Li, X.; Lu, W. A review on fluid dynamics of flapping foils. *Ocean Eng.* **2020**, *195*, 106712. [CrossRef]
65. Verma, S.; Hemmati, A. Characterization of bifurcated dual vortex streets in the wake of an oscillating foil. *J. Fluid Mech.* **2022**, *945*, A7. [CrossRef]
66. Imamura, T.; Suzuki, K.; Nakamura, T.; Yoshida, M. Flow simulation around an airfoil by lattice Boltzmann method on generalized coordinates. *AIAA J.* **2005**, *43*, 1968–1973. [CrossRef]
67. Johnson, A.A.; Tezduyar, T.E. Mesh update strategies in parallel finite element computations of flow problems with moving boundaries and interfaces. *Comput. Methods Appl. Mech. Eng.* **1994**, *119*, 73–94. [CrossRef]
68. Hellström, J.G.I.; Frishfelds, V.; Lundström, T.S. Mechanisms of flow-induced deformation of porous media. *J. Fluid Mech.* **2010**, *664*, 220–237. [CrossRef]
69. Wang, R.; Wei, Y.; Wu, C.; Sun, L.; Zheng, W. Numerical simulations of the motion and deformation of three RBCs during poiseuille flow through a constricted vessel using IB-LBM. *Comput. Math. Methods Med.* **2018**, *2018*, 9425375. [CrossRef]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.