

Article

An Interface-Fitted Fictitious Domain Finite Element Method for the Simulation of Neutrally Buoyant Particles in Plane Shear Flow

Yi Liang ¹ , Cheng Wang ¹ and Pengtao Sun ^{2,*}

¹ School of Mathematical Sciences, Tongji University, Shanghai 200092, China; 2310285@tongji.edu.cn (Y.L.); wangcheng@tongji.edu.cn (C.W.)

² Department of Mathematical Sciences, University of Nevada Las Vegas, Las Vegas, NV 89154, USA

* Correspondence: pengtao.sun@unlv.edu

Abstract: In this paper, an interface-fitted fictitious domain finite element method is developed for the simulation of fluid–rigid particle interaction problems in cases of rotated particles with small displacement, where an interface-fitted mesh is employed for the discrete scheme to capture the fluid–rigid particle interface accurately, thereby improving the solution accuracy near the interface. Moreover, a linearization and decoupling process is presented to release the constraint between velocities of fluid and rigid particles in the finite element space, and to make the developed numerical method easy to be implemented. Our numerical experiments are carried out using two different moving interface-fitted meshes; one is obtained by a rotational arbitrary Lagrangian–Eulerian (ALE) mapping, and the other one through a local smoothing process among interface-cut elements. A unified velocity is defined in the entire domain based on the fictitious domain method, making it easier to develop an interface-fitted mesh generation algorithm in a fixed domain. Both show that the proposed method has a good performance in accuracy for simulating a neutrally buoyant particle in plane shear flow. This approach can be easily extended to fluid–structure interaction problems involving fluids in different states and structures in different shapes with large displacements or deformations.

Keywords: fictitious domain method; interface-fitted mesh; fluid–rigid particle interaction; arbitrary Lagrangian–Eulerian (ALE) method

MSC: 65M06; 76D07



Citation: Liang, Y.; Wang, C.; Sun, P. An Interface-Fitted Fictitious Domain Finite Element Method for the Simulation of Neutrally Buoyant Particles in Plane Shear Flow. *Fluids* **2023**, *8*, 229. <https://doi.org/10.3390/fluids8080229>

Academic Editors: Fang-Bao Tian, Ricardo Ruiz Baier and Pengtao Yue

Received: 11 July 2023

Revised: 10 August 2023

Accepted: 10 August 2023

Published: 12 August 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Fluid–structure interaction (FSI) is a crucial research field in current engineering applications, see for example [1–7]. Particle motion in shear flow is a typical representative and has been a hot topic among scholars, with widespread applications in slurry processing [8], colloid separation [9], and biological cell motion [10]. As early as 1922, Jeffery [11] studied the motion of elliptical particles in creeping flow and provided an analytical expression for the relationship between the angular velocity and the angle. Ding and Aidun [12] performed direct numerical simulations using the discrete Boltzmann method to study the motion of elliptical particles in simple shear flow and found that the behavior of particles at low Reynolds numbers agrees well with periodic solutions of Jeffery. Lundell et al. [13] also conducted numerical studies on the motion of an inertial ellipsoid in a creeping linear shear flow of a Newtonian fluid and found that the particle motion is similar to Jeffery’s solution but with the addition of an orbit drift. Furthermore, Pasquino et al. [14] used both experimental and numerical approaches to study the migration and chaining behavior of noncolloidal spheres in a worm-like micellar, viscoelastic solution within the shear flow.

The fictitious domain method is an effective approach for solving FSI problems, which extends the fictitious fluid into solid particles and imposes constraints on their motions in the fluid. Initially proposed by Saul'ev in 1963 [15], the fictitious domain method was used to extend partial differential equations from complex domains to regular ones. Glowinski et al. [16] analyzed the flow around rigid bodies using the fictitious domain method with the Lagrange multiplier in the case of known solid motions. Furthermore, under the assumption that rigid body motions are not known in advance, they directly simulated the flow around moving rigid bodies in two-dimensional and three-dimensional Newtonian and non-Newtonian viscous fluids using the fictitious domain method with the distributed Lagrange multipliers in order to impose constraints on the motion of solid bodies [17]. The distributed Lagrange multipliers-based fictitious domain method for solving fluid–structure interaction problems can also be found in [3,18]. Hwang et al. [19] proposed a novel finite element scheme for direct simulation of the simple shear flow of an inertia-free particle suspension in a Newtonian fluid, applying rigid body conditions on particle boundaries as traction forces through Lagrange multipliers. Yu et al. [4] use the distributed Lagrange multipliers-based fictitious domain method in the simulation of the motion of a spherical particle in a deterministic lateral displacement device. Wang et al. [20] proposed a one-field fictitious domain method for solving general FSI problems, where only fluid variables (velocity and pressure) are defined in the whole domain, while structural variables are no longer defined. This approach not only improves computational efficiency but also maintains the generality and robustness of the distributed Lagrange multiplier-based fictitious domain method.

Nevertheless, the fictitious domain method typically employs a uniformly fixed grid in the entire domain while a moving structural grid on the top of it, which can lead to computational errors near the moving interface as it cannot effectively track the immersed boundary. For example, Auricchio et al. [21] have mentioned that the lack of global regularity in the solution of moving interface problems would cause suboptimal convergence of standard finite element formulation if the computational mesh near the interface is not appropriately modified. Considering the inherent limitations of the fictitious domain method on fixed grids, it seems necessary to introduce grid modification around the interfaces to accurately track moving interfaces and improve computational accuracy. For instance, Wan et al. [22] presented a fictitious boundary method for the simulation of particulate flows, which incorporates the motion of solids and interfaces as additional constraints to the governing Navier–Stokes equations, thereby extending the fluid domain to the entire region. In order to address the issue of low accuracy in boundary approximation, they dynamically relocated the mesh through a special partial differential equation to capture the region near the surface of the moving particles with high accuracy.

In this paper, we develop an interface-fitted fictitious domain method for the simulation of fluid-particle interaction problems. On the one hand, compared with the classical fictitious domain method, our solution accuracy is improved since an interface-fitted mesh is used to track the motion of particles. We note that additional computational efforts to generate the body-fitted mesh are required in this proposed method, and two interface-fitted mesh generation algorithms are given in cases of rotated particles with small displacement. On the other hand, compared to the classical arbitrary Lagrangian–Eulerian (ALE) method, we eliminate the requirement of a fixed number of vertices/elements in the fluid mesh by defining a unified unknown velocity over the entire domain, and instead need to generate an interface-fitted grid in the fixed entire domain. We note that both the classical ALE method and the proposed method can deal with FSI problems in cases of particles with small displacement. But due to the elimination of the requirement of a fixed number of vertices/elements in the fluid, the proposed method is expected to be capable of dealing with FSI problems in cases of particles with large displacement without any remeshing and interpolation.

The rest of this paper is structured as follows. In Section 2, we introduce the model problem of neutral buoyancy particles moving in a planar shear flow. Then, we present the

weak formulation based on the fictitious domain method and its ALE description, followed by temporal and spatial discretization, as detailed in Section 3. Numerical results of circular and elliptical particles in planar shear flows are presented in Section 4. The conclusions are reported in Section 5.

2. Model Problems

Let $\Omega \in \mathbb{R}^2$ be a rectangular domain filled with a Newtonian viscous incompressible fluid and a rigid particle, as depicted in Figure 1. The bottom, right, top and left boundary of Ω are denoted by $\Gamma_1, \Gamma_2, \Gamma_3$ and Γ_4 , respectively. The distance between Γ_2 and Γ_4 (resp. Γ_1 and Γ_3) is denoted by L (resp. D). For any time $t \in [0, T]$, the interface between fluid and rigid particle is denoted by Γ^t , which divide the domain Ω into two parts: the fluid domain Ω_f^t and the particle domain Ω_s^t .

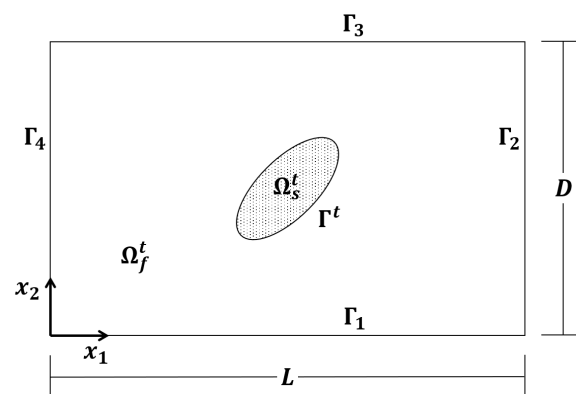


Figure 1. An example of a fluid domain with a particle.

For the fluid, the governing equations are given by the following Navier–Stokes equations [17],

$$\rho_f \frac{du_f}{dt} = \rho_f g + \nabla \cdot \sigma_f, \quad \text{in } \Omega_f^t \times (0, T], \quad (1)$$

$$\nabla \cdot u_f = 0, \quad \text{in } \Omega_f^t \times (0, T], \quad (2)$$

where $\frac{du_f}{dt} = \frac{\partial u_f}{\partial t} + (u_f \cdot \nabla)u_f$, u_f is the fluid velocity, g is the gravity, $\sigma_f = -pI + 2\mu_f D(u_f)$ is the stress tensor with $D(u_f) = \frac{1}{2}(\nabla u_f + (\nabla u_f)^T)$, p is the fluid pressure, ρ_f and μ_f are the fluid density and viscosity, respectively. The initial condition is defined as

$$u_f(x, 0) = u_f^0(x), \quad \text{in } \Omega_f^0, \quad (3)$$

while the boundary conditions are given as follows,

$$u_f(x, t) = -g_0, \quad \text{on } \Gamma_1 \times (0, T], \quad (4)$$

$$u_f(x, t) = g_0, \quad \text{on } \Gamma_3 \times (0, T], \quad (5)$$

and

$$u_f([0, x_2], t) = u_f([L, x_2], t), \quad \forall x_2 \in [0, D], \quad t \in (0, T], \quad (6)$$

$$\sigma_f n_f([0, x_2], t) = -\sigma_f n_f([L, x_2], t), \quad \forall x_2 \in [0, D], \quad t \in (0, T], \quad (7)$$

where $g_0 = (U_0/2, 0)^T$, n_f is the outward unit normal vector to the boundary of the fluid domain.

Equations (4) and (5) indicate that the fluid flow field to be studied exhibits characteristics of a shear flow. Such a characteristic arises when velocities of adjacent fluid layers that are parallel to each other are different, which can be attributed to the reinforced boundary conditions. Equations (6) and (7) imply that the fluid velocity is periodic in the x_1 direction with the period L , which serves to restore the scenario of unbounded shear flow in the horizontal direction as described in Jeffery's solution.

The rigid particle motion is defined by the following Euler–Newton's equations [17] for any $t \in (0, T]$,

$$M \frac{d\mathbf{U}}{dt} = M\mathbf{g} + \mathbf{F}(\mathbf{u}_f, p_f), \quad (8)$$

$$I_G \frac{d\omega}{dt} = T(\mathbf{u}_f, p_f), \quad (9)$$

$$\frac{d\mathbf{G}}{dt} = \mathbf{U}, \quad (10)$$

$$\frac{d\theta}{dt} = \omega, \quad (11)$$

where $M = \int_{\Omega_s^0} \rho_s dx$, ρ_s and $I_G = \int_{\Omega_s^0} \rho_s ((x_1 - G_1)^2 + (x_2 - G_2)^2) dx$, are the mass, density, and moment of inertia of the rigid particle, respectively. $\mathbf{G} = (G_1, G_2)^T$ is the center of mass, $\mathbf{U}$ is the velocity of the center of mass, and, ω and θ are the angular velocity and inclination angle of the particle, respectively. Moreover, the hydrodynamical force and torque of the immersed rigid particle are defined as

$$\mathbf{F}(\mathbf{u}_f, p_f) = - \int_{\Gamma^t} \sigma_f \mathbf{n}_f ds, \quad (12)$$

$$T(\mathbf{u}_f, p_f) = - \int_{\Gamma^t} \overrightarrow{\mathbf{G}\mathbf{x}^\perp} \cdot (\sigma_f \mathbf{n}_f) ds, \quad (13)$$

where $\overrightarrow{\mathbf{G}(t)\mathbf{x}^\perp} = (-(x_2 - G_2), x_1 - G_1)^T$.

The initial conditions for (8)–(11) are defined as

$$\mathbf{U}(0) = \mathbf{U}^0, \quad \omega(0) = \omega^0, \quad \mathbf{G}(0) = \mathbf{G}^0, \quad \theta(0) = \theta^0, \quad (14)$$

where $\mathbf{U}^0$, ω^0 , $\mathbf{G}^0$ and θ^0 are given initial values. On the fluid-particle interface Γ^t , the following interface condition is imposed,

$$\mathbf{u}_f(\mathbf{x}, t) = \mathbf{U}(t) + \omega(t) \overrightarrow{\mathbf{G}(t)\mathbf{x}^\perp}, \quad \text{on } \Gamma^t \times (0, T], \quad (15)$$

which indicates that there is no relative sliding between the fluid and the rigid body.

In summary, all symbols involved in the above model descriptions and their SI units are also listed in Table 1.

Table 1. Symbol descriptions and their SI units.

Symbols	Description	Units
L	Length of domain	m
D	Width of domain	m
$\mathbf{u}_f$	Fluid velocity	$\text{m} \cdot \text{s}^{-1}$
p	Fluid pressure	Pa
$\mathbf{g}$	Gravitational acceleration	$\text{m} \cdot \text{s}^{-2}$
ρ_f	Density of fluid	$\text{kg} \cdot \text{m}^{-3}$
μ_f	Viscosity of fluid	$\text{Pa} \cdot \text{s}$
ρ_s	Density of particle	$\text{kg} \cdot \text{m}^{-3}$

Table 1. *Cont.*

Symbols	Description	Units
M	Mass of particle	kg
$\mathbf{U}$	Velocity of center of mass	$\text{m} \cdot \text{s}^{-1}$
$\mathbf{G}$	Position of center of mass	m
I_G	Moment of inertia of particle	$\text{kg} \cdot \text{m}^2$
ω	Angular velocity of particle	$\text{rad} \cdot \text{s}^{-1}$
θ	Inclination angle of particle	rad
$\mathbf{F}$	Hydrodynamic force of particle	N
T	Hydrodynamic torque of particle	$\text{N} \cdot \text{m}$

3. An Interface-Fitted Fictitious Domain Finite Element Method

3.1. Weak Formulation Based on the Fictitious Domain Method

In this section, we introduce a unified velocity field and present a weak formulation of the aforementioned model problem based on the fictitious domain method.

Introduce Sobolev spaces

$$\tilde{\mathbb{V}}_f = \left\{ (v_f, \mathbf{V}, \xi) \left| v_f \in \left(H^1(\Omega_f^t) \right)^2, \mathbf{V} \in \mathbb{R}^2, \xi \in \mathbb{R}, v_f = \mathbf{V} + \xi \overrightarrow{\mathbf{G}\mathbf{x}}^\perp \text{ on } \Gamma^t, \right. \right. \\ \left. \left. v_f = 0 \text{ on } \Gamma_1 \cup \Gamma_3, v_f \text{ is periodic on } \Gamma_2 \text{ and } \Gamma_4 \right\}, \quad (16)$$

$$\mathcal{Q}_f = \left\{ q_f \left| q_f \in L^2(\Omega_f^t), \int_{\Omega_f^t} q_f d\mathbf{x} = 0 \right. \right\}. \quad (17)$$

For any $(v_f, \mathbf{V}, \xi) \in \tilde{\mathbb{V}}_f$, multiplying both sides of (1) by v_f , and conducting integration by parts on the stiffness term, we have

$$\rho_f \int_{\Omega_f^t} \frac{d\mathbf{u}_f}{dt} \cdot v_f d\mathbf{x} + 2\mu_f \int_{\Omega_f^t} \mathbf{D}(\mathbf{u}_f) : \mathbf{D}(v_f) d\mathbf{x} - \int_{\Omega_f^t} p_f (\nabla \cdot v_f) d\mathbf{x} \\ - \int_{\partial\Omega_f^t} v_f \cdot (\sigma_f \mathbf{n}_f) ds = \rho_f \int_{\Omega_f^t} \mathbf{g} \cdot v_f d\mathbf{x}, \quad (18)$$

where we can rewrite the above boundary integral term as follows by using (8), (9), (12) and (13),

$$\int_{\partial\Omega_f^t} v_f \cdot (\sigma_f \mathbf{n}_f) ds = \int_{\Gamma_1 \cup \Gamma_3} v_f \cdot (\sigma_f \mathbf{n}_f) ds + \int_{\Gamma_2 \cup \Gamma_4} v_f \cdot (\sigma_f \mathbf{n}_f) ds + \int_{\Gamma^t} v_f \cdot (\sigma_f \mathbf{n}_f) ds \\ = \int_{\Gamma^t} (\mathbf{V} + \xi \overrightarrow{\mathbf{G}\mathbf{x}}^\perp) \cdot (\sigma_f \mathbf{n}_f) ds = -M \left(\frac{d\mathbf{U}}{dt} - \mathbf{g} \right) \cdot \mathbf{V} - I_G \frac{d\omega}{dt} \xi.$$

Thus, for any $(v_f, \mathbf{V}, \xi) \in \tilde{\mathbb{V}}_f$ and $q_f \in \mathcal{Q}_f$, we obtain the following weak formulation of fluid part,

$$\rho_f \int_{\Omega_f^t} \frac{d\mathbf{u}_f}{dt} \cdot v_f d\mathbf{x} + 2\mu_f \int_{\Omega_f^t} \mathbf{D}(\mathbf{u}_f) : \mathbf{D}(v_f) d\mathbf{x} - \int_{\Omega_f^t} p_f (\nabla \cdot v_f) d\mathbf{x} \\ + M \left(\frac{d\mathbf{U}}{dt} - \mathbf{g} \right) \cdot \mathbf{V} + I_G \frac{d\omega}{dt} \xi = \rho_f \int_{\Omega_f^t} \mathbf{g} \cdot v_f d\mathbf{x}, \quad (19)$$

$$\int_{\Omega_f^t} q_f (\nabla \cdot \mathbf{u}_f) d\mathbf{x} = 0. \quad (20)$$

Next, we introduce the particle velocity $\mathbf{u}_s$, which is defined by

$$\mathbf{u}_s = \mathbf{U} + \omega \overrightarrow{\mathbf{G}\mathbf{x}}^\perp. \quad (21)$$

It is easy to check that

$$\frac{d\mathbf{u}_s}{dt} = \frac{d\mathbf{U}}{dt} + \frac{d\omega}{dt} \overrightarrow{\mathbf{G}\mathbf{x}}^\perp + \omega^2 \overrightarrow{\mathbf{G}\mathbf{x}} \quad (22)$$

and

$$\int_{\Omega_s^t} \overrightarrow{\mathbf{G}\mathbf{x}} d\mathbf{x} = \int_{\Omega_s^t} (\mathbf{x} - \mathbf{G}) d\mathbf{x} = \int_{\Omega_s^t} \left(\mathbf{x} - \frac{\int_{\Omega_s^t} \mathbf{x} d\mathbf{x}}{\int_{\Omega_s^t} d\mathbf{x}} \right) d\mathbf{x} = \mathbf{0}, \quad (23)$$

$$\begin{aligned} \int_{\Omega_s^t} \overrightarrow{\mathbf{G}\mathbf{x}}^\perp d\mathbf{x} &= \int_{\Omega_s^t} (-(x_2 - G_2), x_1 - G_1)^T d\mathbf{x} \\ &= \int_{\Omega_s^t} \left(-\left(x_2 - \frac{\int_{\Omega_s^t} x_2 d\mathbf{x}}{\int_{\Omega_s^t} d\mathbf{x}} \right), x_1 - \frac{\int_{\Omega_s^t} x_1 d\mathbf{x}}{\int_{\Omega_s^t} d\mathbf{x}} \right)^T d\mathbf{x} = \mathbf{0}. \end{aligned} \quad (24)$$

For any $\mathbf{V} \in \mathbb{R}^2$ and $\zeta \in \mathbb{R}$, let $\mathbf{v}_s = \mathbf{V} + \zeta \overrightarrow{\mathbf{G}\mathbf{x}}^\perp$. We have

$$\rho_s \int_{\Omega_s^t} \frac{d\mathbf{u}_s}{dt} \cdot \mathbf{V} d\mathbf{x} = \rho_s \int_{\Omega_s^t} \left(\frac{d\mathbf{U}}{dt} + \frac{d\omega}{dt} \overrightarrow{\mathbf{G}\mathbf{x}}^\perp + \omega^2 \overrightarrow{\mathbf{G}\mathbf{x}} \right) \cdot \mathbf{V} d\mathbf{x} = M \frac{d\mathbf{U}}{dt} \cdot \mathbf{V}, \quad (25)$$

$$\rho_s \int_{\Omega_s^t} \frac{d\mathbf{u}_s}{dt} \cdot \zeta \overrightarrow{\mathbf{G}\mathbf{x}}^\perp d\mathbf{x} = \rho_s \int_{\Omega_s^t} \left(\frac{d\mathbf{U}}{dt} + \frac{d\omega}{dt} \overrightarrow{\mathbf{G}\mathbf{x}}^\perp + \omega^2 \overrightarrow{\mathbf{G}\mathbf{x}} \right) \cdot \zeta \overrightarrow{\mathbf{G}\mathbf{x}}^\perp d\mathbf{x} = I_G \frac{d\omega}{dt} \zeta, \quad (26)$$

$$\rho_s \int_{\Omega_s^t} \mathbf{g} \cdot \mathbf{v}_s d\mathbf{x} = \rho_s \int_{\Omega_s^t} \mathbf{g} \cdot \mathbf{V} d\mathbf{x} + \rho_s \int_{\Omega_s^t} \mathbf{g} \cdot \zeta \overrightarrow{\mathbf{G}\mathbf{x}}^\perp d\mathbf{x} = M \mathbf{g} \cdot \mathbf{V}. \quad (27)$$

Then for any $\mathbf{V} \in \mathbb{R}^2$ and $\zeta \in \mathbb{R}$, we can attain

$$\rho_f \int_{\Omega_f^t} \frac{d\mathbf{u}_s}{dt} \cdot \mathbf{v}_s d\mathbf{x} - \frac{\rho_f}{\rho_s} \left(M \left(\frac{d\mathbf{U}}{dt} - \mathbf{g} \right) \cdot \mathbf{V} + I_G \frac{d\omega}{dt} \zeta \right) = \rho_f \int_{\Omega_f^t} \mathbf{g} \cdot \mathbf{v}_s d\mathbf{x}. \quad (28)$$

Furthermore, we define a unified velocity

$$\mathbf{u} = \begin{cases} \mathbf{u}_f, & \text{on } \Omega_f^t, \\ \mathbf{u}_s, & \text{on } \Omega_s^t, \end{cases} \quad (29)$$

which is defined over the entire domain Ω .

Introduce Sobolev spaces

$$\begin{aligned} \mathbb{V} = \left\{ (\mathbf{v}, \mathbf{V}, \zeta) \middle| \mathbf{v} \in \left(H^1(\Omega) \right)^2, \mathbf{V} \in \mathbb{R}^2, \zeta \in \mathbb{R}, \mathbf{v} = \mathbf{V} + \zeta \overrightarrow{\mathbf{G}\mathbf{x}}^\perp \text{ on } \Omega_s^t, \right. \\ \left. \mathbf{v} = -\mathbf{g}_0 \text{ on } \Gamma_1, \mathbf{v} = \mathbf{g}_0 \text{ on } \Gamma_3, \mathbf{v} \text{ is periodic on } \Gamma_2 \text{ and } \Gamma_4 \right\}, \end{aligned} \quad (30)$$

$$\begin{aligned} \tilde{\mathbb{V}} = \left\{ (\mathbf{v}, \mathbf{V}, \zeta) \middle| \mathbf{v} \in \left(H^1(\Omega) \right)^2, \mathbf{V} \in \mathbb{R}^2, \zeta \in \mathbb{R}, \mathbf{v} = \mathbf{V} + \zeta \overrightarrow{\mathbf{G}\mathbf{x}}^\perp \text{ on } \Omega_s^t, \right. \\ \left. \mathbf{v} = 0 \text{ on } \Gamma_1 \cup \Gamma_3, \mathbf{v} \text{ is periodic on } \Gamma_2 \text{ and } \Gamma_4 \right\}. \end{aligned} \quad (31)$$

Note that $\frac{d\mathbf{u}}{dt} = \frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u}$. Thus, by combining (19), (20) and (28), we obtain the final weak formulation of the fluid-particle interaction problem: for any $(\mathbf{v}, \mathbf{V}, \xi) \in \tilde{\mathbb{V}}$ and $q_f \in Q_f$, find $(\mathbf{u}, \mathbf{U}, \omega) \in \mathbb{V}$ and $p_f \in Q_f$ such that

$$\begin{aligned} \rho_f \int_{\Omega} \left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} \right) \cdot \mathbf{v} \, dx + 2\mu_f \int_{\Omega_f^t} \mathbf{D}(\mathbf{u}) : \mathbf{D}(\mathbf{v}) \, dx - \int_{\Omega_f^t} p_f (\nabla \cdot \mathbf{v}) \, dx \\ + \left(1 - \frac{\rho_f}{\rho_s} \right) \left[M \left(\frac{d\mathbf{U}}{dt} - \mathbf{g} \right) \cdot \mathbf{V} + I_G \frac{d\omega}{dt} \xi \right] = \rho_f \int_{\Omega} \mathbf{g} \cdot \mathbf{v} \, dx, \end{aligned} \quad (32)$$

$$\int_{\Omega_f^t} q_f (\nabla \cdot \mathbf{u}) \, dx = 0, \quad (33)$$

$$M \frac{d\mathbf{U}}{dt} = M\mathbf{g} + \mathbf{F}(\mathbf{u}, p_f), \quad (34)$$

$$I_G \frac{d\omega}{dt} = T(\mathbf{u}, p_f), \quad (35)$$

subjecting to the following initial conditions

$$\mathbf{u} = \mathbf{u}^0, \quad \mathbf{U} = \mathbf{U}^0, \quad \omega = \omega^0, \quad (36)$$

where

$$\mathbf{u}^0 = \begin{cases} \mathbf{u}_f^0, & \text{in } \Omega_f^0, \\ \mathbf{U}^0 + \omega^0 \mathbf{G}^0 \mathbf{x}^\perp, & \text{in } \Omega_s^0. \end{cases} \quad (37)$$

For the cases of neutrally buoyant particles, i.e., $\rho_s = \rho_f$, Equation (32) can be simply rewritten as

$$\begin{aligned} \rho_f \int_{\Omega} \left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} \right) \cdot \mathbf{v} \, dx + 2\mu_f \int_{\Omega_f^t} \mathbf{D}(\mathbf{u}) : \mathbf{D}(\mathbf{v}) \, dx \\ - \int_{\Omega_f^t} p_f (\nabla \cdot \mathbf{v}) \, dx = \rho_f \int_{\Omega} \mathbf{g} \cdot \mathbf{v} \, dx. \end{aligned} \quad (38)$$

3.2. Interface-Fitted Mesh and ALE Mapping

Although the rectangular domain Ω is independent of time in which the unified velocity $\mathbf{u}$ is defined, the fluid-particle interface moves along in time. Thus, we need to construct an interface-fitted mesh in Ω in order to capture the surface of immersed particle, and therefore improve the accuracy of numerical simulation.

Let N be a positive integer, $\Delta t = T/N$ be the time step and $t^n = n\Delta t$, $n = 0, \dots, N$. Denote by $\mathcal{T}_h^n$ a triangulation of $\Omega^n := \Omega^{t^n}$, where the superscript n indicates the mesh is interface-fitted with Γ^{t^n} . That is, for any edge e in $\mathcal{T}_h^n$, $e \cap \Gamma_h^{t^n} = \emptyset$, where e denotes the interior of e . For any subdomain D , $\mathcal{T}_h^n(D)$ denotes the restriction of $\mathcal{T}_h^n$ on D .

Generally, constructing such interface-fitted meshes and frequent remeshing can lead to prohibitive CPU time in the simulation of FSI problems, especially when the displacement or deformation of the structure is large. In our fictitious domain finite element scheme, the unified unknown velocity is defined in the fixed and connected domain Ω , which releases the requirement that the number of vertices in the fluid subdomain needs to be fixed when an ALE method is used. Although the number of vertices of $\mathcal{T}_h^n$ needs to keep fixed, it is much easier to generate an interface-fitted mesh for the entire fixed domain Ω . In fact, some existing moving mesh strategies, such as those in [23–26], can be employed for generating such interface-fitted background meshes with some slight modifications.

First, in following the idea presented in [26], one can construct an interface-fitted mesh and the corresponding discrete ALE mapping in cases of rotated particles with small

displacement, which is called moving mesh Algorithm 1 in this paper, as illustrated in Figure 2 and described below.

Algorithm 1: Moving mesh by a rotational ALE mapping.

1. Draw an artificial circular buffer zone, Ω_{rf}^t , in the fluid domain Ω_f^t to enclose the immersed particle Ω_s^t and share the same axis of rotation. Denote $\Omega_{sf}^t = \Omega_f^t \setminus \Omega_{rf}^t$, and $\Gamma_{rs}^t = \Omega_{rf}^t \cap \Omega_{sf}^t$.
 2. Triangulate Ω_f^0 with $\mathcal{T}_h^0$ that fits Γ^0 and Γ_{rs}^0 , i.e., for any edge $e \in \mathcal{T}_h^0$, let $e \cap \Gamma_{rs}^0 = \emptyset$ and $e \cap \Gamma^0 = \emptyset$. Fix the mesh $\mathcal{T}_h^n(\Omega_{sf}^n)$ as $\mathcal{T}_h^0(\Omega_{sf}^0)$ for $n > 0$.
 3. Rotate $\mathcal{T}_h^{n-1}(\Omega_{rf}^{n-1})$ together with the particle at the same angular velocity and about the same axis of rotation to obtain a rotating mesh $\widehat{\mathcal{T}}_h^n(\Omega_{rf}^n)$.
 4. Locally shift the nodes on the artificial interface Γ_{rs} associated with $\widehat{\mathcal{T}}_h^n(\Omega_{rf}^n)$ in order to find and match with the fixed coordinates of the closest nodes on Γ_{rs} associated with $\mathcal{T}_h^n(\Omega_{sf}^n)$. Thus, the rotating and conforming mesh $\mathcal{T}_h^n(\Omega_{rf}^n)$, further, the total mesh $\mathcal{T}_h^n = \mathcal{T}_h^n(\Omega_{rf}^n) \cup \mathcal{T}_h^n(\Omega_{sf}^n)$ are obtained.
-

We note that the discrete ALE-based mesh velocity, w_h , only needs to be computed in Ω_{rf} , since the mesh $\mathcal{T}_h^n(\Omega_{sf}^n)$ is fixed. Algorithm 1 guarantees a rigid particle's vertex of $\mathcal{T}_h^0$ (resp. $\mathcal{T}_h^0(\Omega_{rf})$) is always the same rigid particle's vertex of $\mathcal{T}_h^n$ (resp. $\mathcal{T}_h^n(\Omega_{rf})$) all the time, establishing a one-to-one mapping (denoted by X_h^n) between vertices of $\mathcal{T}_h^n(\Omega_{rf})$ and of $\mathcal{T}_h^0(\Omega_{rf})$.

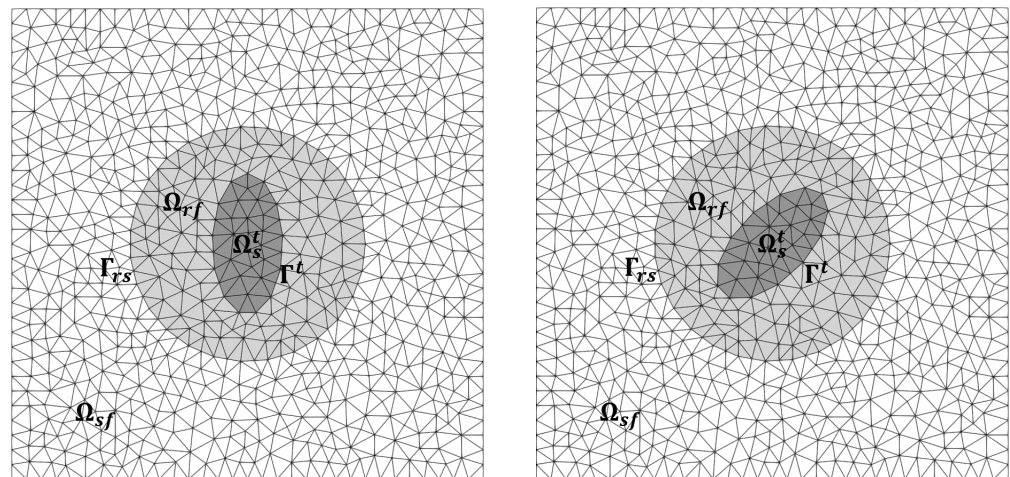


Figure 2. Moving mesh Algorithm 1: interface-fitted meshes at different time steps.

Next, we present another moving mesh algorithm called Algorithm 2, which is similar to the method proposed in [23].

We remark that Algorithm 2 may deliver such a interface-fitted mesh $\mathcal{T}_h^n$ whose rigid particle's (resp. fluid) vertices at one time step may be fluid (resp. rigid particle's) vertices at another time step (e.g., see the bottom-right part of Figure 3), but still form a one-to-one discrete ALE mapping between vertices of $\mathcal{T}_h^n$ and of $\mathcal{T}_h^0$ that is still denoted by X_h^n .

Algorithm 2: Moving mesh by local smoothing among interface-cut elements.

1. Triangulate Ω into a rectangular mesh $\hat{\mathcal{T}}_h$ with a given mesh size h_x (resp. h_y) in x -axis (resp. y -axis) direction that is independent of time t .
2. Inspect all vertices of $\hat{\mathcal{T}}_h$ in the column-wise lexicographic order, and divide them into fluid- (black), interface- (red), and rigid particle's (blue) vertex sets associated with the current position of fluid-particle interface Γ^{t_n} , where the red interface vertices are those vertices closest to Γ^{t_n} , as shown in the top-left part of Figure 3.
3. Move all red interface vertices onto the fluid-particle interface to set them as the closest intersection points between Γ^{t_n} and $\hat{\mathcal{T}}_h$, respectively.
4. Cut each quadrilateral of $\hat{\mathcal{T}}_h$ with the slash diagonal to obtain a triangular mesh of Ω , $\tilde{\mathcal{T}}_h$, where some edges of $\tilde{\mathcal{T}}_h$ may intersect with the interface, as shown in the top-right part of Figure 3.
5. Locally adjust interface-cut elements of $\tilde{\mathcal{T}}_h$ using vertex smoothing and edge swapping techniques [24] to enhance the mesh quality, simultaneously, ensure the edges of each element are all aligned with the fluid-particle interface Γ^{t_n} . Thus, an interface-fitted mesh $\mathcal{T}_h^n$ is generated, as depicted in the bottom-left part of Figure 3.

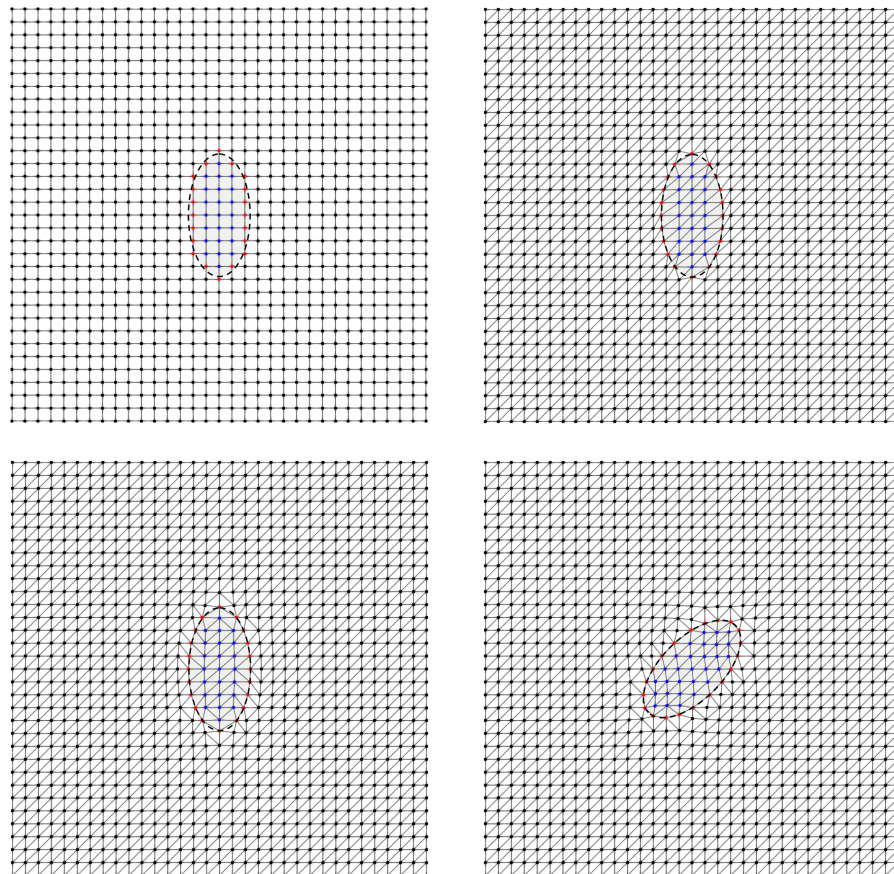


Figure 3. Moving mesh Algorithm 2: interface-fitted meshes at different time steps.

Once the interface-fitted mesh $\mathcal{T}_h^n$ is obtained, the discrete ALE velocity w_h^n , which is a piecewise linear function associated with $\mathcal{T}_h^n$, can be computed by

$$w_h^n(P) = \frac{\mathbf{x}^n(P) - \mathbf{x}^{n-1} \left(X_h^{n-1} \circ (X_h^n)^{-1}(P) \right)}{\Delta t} \quad (39)$$

for any vertex P of $\mathcal{T}_h^n$. Moreover, the discrete ALE material derivative is defined as

$$\left. \frac{\partial \mathbf{v}_h}{\partial t} \right|_{\hat{\mathbf{x}}}^n : \Omega^n \rightarrow \mathbb{R}^2 \quad (40)$$

$$(\mathbf{x}, t^n) \mapsto \left. \frac{\partial \mathbf{v}_h}{\partial t} \right|_{\hat{\mathbf{x}}}^n(\mathbf{x}) = \left(\frac{\partial \mathbf{v}_h}{\partial t} \right)^n(\mathbf{x}) + (\mathbf{w}_h^n(\mathbf{x}) \cdot \nabla) \mathbf{v}_h^n(\mathbf{x}). \quad (41)$$

Associated with the interface-fitted mesh $\mathcal{T}_h^n$, the subdomains Ω_f^n and Ω_s^n are approximated by $\Omega_{f,h}^n$ and $\Omega_{s,h}^n$, respectively.

3.3. A Interface-Fitted Fictitious Domain Finite Element Method

Define the following finite element spaces

$$\begin{aligned} \mathbb{V}_h^n &= \left\{ (\mathbf{v}_h, \mathbf{V}_h, \xi_h) \mid \mathbf{v}_h|_K \in (P_2)^2, \forall K \in \mathcal{T}_h^n, \mathbf{V}_h \in \mathbb{R}^2, \xi_h \in \mathbb{R}, \mathbf{v}_h = \mathbf{V}_h + \xi_h \overrightarrow{\mathbf{G}_h^n}^\perp \text{ on } \Omega_{s,h}^n, \right. \\ &\quad \left. \mathbf{v}_h = -\mathbf{g}_0 \text{ on } \Gamma_1, \mathbf{v}_h = \mathbf{g}_0 \text{ on } \Gamma_3, \mathbf{v}_h \text{ is periodic on } \Gamma_2 \text{ and } \Gamma_4 \right\}, \\ \tilde{\mathbb{V}}_h^n &= \left\{ (\mathbf{v}_h, \mathbf{V}_h, \xi_h) \mid \mathbf{v}_h|_K \in (P_2)^2, \forall K \in \mathcal{T}_h^n, \mathbf{V}_h \in \mathbb{R}^2, \xi_h \in \mathbb{R}, \mathbf{v}_h = \mathbf{V}_h + \xi_h \overrightarrow{\mathbf{G}_h^n}^\perp \text{ on } \Omega_{s,h}^n, \right. \\ &\quad \left. \mathbf{v}_h = 0 \text{ on } \Gamma_1 \cup \Gamma_3, \mathbf{v}_h \text{ is periodic on } \Gamma_2 \text{ and } \Gamma_4 \right\}, \\ \mathcal{Q}_{f,h}^n &= \left\{ q_{f,h} \mid q_{f,h} \in L^2(\Omega_f^n), \int_{\Omega_{f,h}^n} q_{f,h} d\mathbf{x} = 0, q_{f,h}|_T \in P_1, \forall T \in \mathcal{T}_h^n(\Omega_{f,h}^n) \right\}, \end{aligned}$$

where P_k denotes the set of polynomials with order less than or equal to k .

In Algorithm 1, since only the mesh $\mathcal{T}_h^n(\Omega_{rf}^n)$ is updated at each time step, we define $\mathbf{u}_h^{n,*}$ to be a piecewise P_2 finite element interpolation of $\mathbf{u}_h^n$ in Ω_{rf}^n by letting

$$\mathbf{u}_h^{n,*}(P) = \mathbf{u}_h^n \circ \mathbf{X}_h^n \circ (\mathbf{X}_h^{n+1})^{-1}(P) \quad (42)$$

for any vertex P in $\mathcal{T}_h^{n+1}(\Omega_{rf}^{n+1})$. The discrete ALE velocity $\mathbf{w}_h^{n+1}$ is calculated by (39) only in Ω_{rf}^{n+1} . In order to describe the method in a unified formulation, we extend $\mathbf{w}_h^{n+1}$ from Ω_{rf}^n to the whole domain with zero, and let $\mathbf{u}_h^{n,*}|_{\Omega_{sf}^n} = \mathbf{u}_h^n$.

In Algorithm 2, we define $\mathbf{u}_h^{n,*}$ to be a piecewise P_2 finite element interpolation of $\mathbf{u}_h^n$ in terms of the same formula (42) but now for each vertex $P \in \mathcal{T}_h^{n+1}$. The discrete ALE velocity $\mathbf{w}_h^{n+1}$ is defined in the whole domain and can be calculated by (39).

Then, using the backward Euler scheme, an interface-fitted fictitious domain finite element method for solving the fluid-particle interaction problem can be written as follows: for $n = 0, \dots, N-1$, find $(\mathbf{u}_h^{n+1}, \mathbf{u}_h^{n+1}, \omega_h^{n+1}) \in \mathbb{V}_h^{n+1}$ and $p_{f,h}^{n+1} \in \mathcal{Q}_{f,h}^{n+1}$ such that

$$\begin{aligned} \rho_f \sum_{K \in \mathcal{T}_h^{n+1}} \int_K \left(\frac{\mathbf{u}_h^{n+1} - \mathbf{u}_h^{n,*}}{\Delta t} + (\mathbf{u}_h^{n+1} \cdot \nabla) \mathbf{u}_h^{n+1} \right) \cdot \mathbf{v}_h^{n+1} d\mathbf{x} \\ + 2\mu_f \int_{\Omega_f^{n+1}} \mathbf{D}(\mathbf{u}_h^{n+1}) : \mathbf{D}(\mathbf{v}_h^{n+1}) d\mathbf{x} - \rho_f \sum_{K \in \mathcal{T}_h^{n+1}} \int_K (\mathbf{w}_h^{n+1} \cdot \nabla) \mathbf{u}_h^{n+1} \cdot \mathbf{v}_h^{n+1} d\mathbf{x} \quad (43) \end{aligned}$$

$$- \int_{\Omega_f^{n+1}} p_{f,h}^{n+1} (\nabla \cdot \mathbf{v}_h^{n+1}) d\mathbf{x} = \rho_f \int_{\Omega^{n+1}} \mathbf{g} \cdot \mathbf{v}_h^{n+1} d\mathbf{x},$$

$$\int_{\Omega_f^{n+1}} q_{f,h}^{n+1} (\nabla \cdot \mathbf{u}_h^{n+1}) d\mathbf{x} = 0, \quad (44)$$

$$M \frac{\mathbf{U}_h^{n+1} - \mathbf{U}_h^n}{\Delta t} = M\mathbf{g} + \mathbf{F}(\mathbf{u}_h^{n+1}, p_{f,h}^{n+1}), \quad (45)$$

$$I_G \frac{\omega_h^{n+1} - \omega_h^n}{\Delta t} = T(\mathbf{u}_h^{n+1}, p_{f,h}^{n+1}), \quad (46)$$

for any $(\mathbf{v}_h^{n+1}, \mathbf{V}_h^{n+1}, \zeta_h^{n+1}) \in \tilde{\mathbb{V}}_h^{n+1}$ and $q_{f,h}^{n+1} \in Q_{f,h}^{n+1}$, subjecting to initial values

$$\mathbf{u}_h^0 = \mathbf{u}_h^0, \quad \mathbf{U}_h^0 = \mathbf{U}^0, \quad \omega_h^0 = \omega^0, \quad (47)$$

where $\mathbf{u}_h^0$ is a P_2 finite element approximation of $\mathbf{u}^0$.

Note that the mesh $\mathcal{T}_h^{n+1}$ depends on $\mathbf{u}_h^{n+1}$. To release the constraint between $\mathbf{v}_h$ and $(\mathbf{V}_h, \zeta_h)$ in the definition of $\mathbb{V}_h^n$, we employ the following Newton's linearization and decoupling process.

For given $\mathbf{U}_h^{n+1,m} \in \mathbb{R}^2$, $\omega_h^{n+1,m} \in \mathbb{R}$, $\overline{\Omega}^{n+1,m} := \overline{\Omega}_{s,h}^{n+1,m} \cup \overline{\Omega}_{f,h}^{n+1,m}$ and $\mathcal{T}_h^{n+1,m}$, we define

$$\mathbb{U}_h^{n+1,m+1} = \left\{ \mathbf{v}_h \mid \mathbf{v}_h|_K \in (P_2)^2, \forall K \in \mathcal{T}_h^{n+1,m}, \mathbf{v}_h = \mathbf{U}_h^{n+1,m} + \omega_h^{n+1,m} \overrightarrow{\mathbf{G}_h^{n+1,m}} \mathbf{x}^\perp \right. \quad (48)$$

$$\left. \text{on } \Omega_{s,h}^{n+1,m}, \mathbf{v}_h = -\mathbf{g}_0 \text{ on } \Gamma_1, \mathbf{v}_h = \mathbf{g}_0 \text{ on } \Gamma_3, \mathbf{v}_h \text{ is periodic on } \Gamma_2 \text{ and } \Gamma_4 \right\}, \quad (49)$$

$$\tilde{\mathbb{U}}_h^{n+1,m+1} = \left\{ \mathbf{v}_h \mid \mathbf{v}_h|_K \in (P_2)^2, \forall K \in \mathcal{T}_h^{n+1,m}, \mathbf{v}_h = \mathbf{U}_h^{n+1,m} + \omega_h^{n+1,m} \overrightarrow{\mathbf{G}_h^{n+1,m}} \mathbf{x}^\perp \right. \quad (50)$$

$$\left. \text{on } \Omega_{s,h}^{n+1,m}, \mathbf{v}_h = 0 \text{ on } \Gamma_1 \cup \Gamma_3, \mathbf{v}_h \text{ is periodic on } \Gamma_2 \text{ and } \Gamma_4 \right\}, \quad (51)$$

$$Q_{f,h}^{n+1,m+1} = \left\{ q_{f,h} \in L^2(\Omega_{f,h}^{n+1,m}) \mid \int_{\Omega_{f,h}^{n+1,m}} q_{f,h} \, d\mathbf{x} = 0, q_{f,h}|_K \in P_1, \forall K \in \mathcal{T}_h^{n+1,m}(\Omega_{f,h}^{n+1,m}) \right\}. \quad (52)$$

Then we can obtain the solution of problems (43)–(46) by alternatively solving the discrete problem for the fluid part; find $\mathbf{u}_h^{n+1,m+1} \in \mathbb{U}_h^{n+1,m+1}$ and $p_{f,h}^{n+1,m+1} \in Q_{f,h}^{n+1,m+1}$ such that

$$\begin{aligned} & \rho_f \sum_{K \in \mathcal{T}_h^{n+1,m}} \int_K \left(\frac{\mathbf{u}_h^{n+1,m+1} - \mathbf{u}_h^{n,*}}{\Delta t} + (\mathbf{u}_h^{n+1,m+1} \cdot \nabla) \mathbf{u}_h^{n+1,m} + (\mathbf{u}_h^{n+1,m} \cdot \nabla) \mathbf{u}_h^{n+1,m+1} \right) \cdot \mathbf{v}_h^{n+1} \, d\mathbf{x} \\ & + 2\mu_f \int_{\Omega_{f,h}^{n+1,m}} \mathbf{D}(\mathbf{u}_h^{n+1,m+1}) : \mathbf{D}(\mathbf{v}_h^{n+1}) \, d\mathbf{x} - \rho_f \sum_{K \in \mathcal{T}_h^{n+1,m}} \int_K (\mathbf{w}_h^{n+1,m} \cdot \nabla) \mathbf{u}_h^{n+1,m+1} \cdot \mathbf{v}_h^{n+1} \, d\mathbf{x} \\ & - \int_{\Omega_{f,h}^{n+1,m}} p_{f,h}^{n+1,m+1} (\nabla \cdot \mathbf{v}_h^{n+1}) \, d\mathbf{x} = \rho_f \int_{\Omega_{n+1,m}} \mathbf{g} \cdot \mathbf{v}_h^{n+1} \, d\mathbf{x} \\ & + \rho_f \int_{\Omega_{n+1,m}} (\mathbf{u}_h^{n+1,m} \cdot \nabla) \mathbf{u}_h^{n+1,m} \cdot \mathbf{v}_h^{n+1} \, d\mathbf{x}, \quad \forall \mathbf{v}_h^{n+1} \in \tilde{\mathbb{U}}_h^{n+1,m+1}, \end{aligned} \quad (53)$$

$$\int_{\Omega_{f,h}^{n+1,m}} q_{f,h}^{n+1} (\nabla \cdot \mathbf{u}_h^{n+1,m+1}) \, d\mathbf{x} = 0, \quad \forall q_{f,h}^{n+1} \in Q_{f,h}^{n+1,m+1}, \quad (54)$$

and the discrete problem for the rigid particle part: find $\mathbf{U}_h^{n+1,m+1} \in \mathbb{R}^2$ and $\omega_h^{n+1,m+1} \in \mathbb{R}$ such that

$$M \frac{\mathbf{U}_h^{n+1,m+1} - \mathbf{U}_h^n}{\Delta t} = M\mathbf{g} + \mathbf{F}(\mathbf{u}_h^{n+1,m+1}, p_{f,h}^{n+1,m+1}), \quad (55)$$

$$I_G \frac{\omega_h^{n+1,m+1} - \omega_h^n}{\Delta t} = T(\mathbf{u}_h^{n+1,m+1}, p_{f,h}^{n+1,m+1}), \quad (56)$$

then updating $\Omega^{n+1,m+1}$, $\mathcal{T}_h^{n+1,m+1}$ and $\mathbf{w}_h^{n+1,m+1}$ using the velocity $\mathbf{u}_h^{n+1,m+1}$.

4. Numerical Experiments

In this section, we shall validate the performance of the proposed interface-fitted fictitious domain finite element method by numerically solving three Stokes flow-rigid

particle interaction problems with different particle shapes. Two moving mesh algorithms given in Section 3.2 are both tested. We remark that no remeshing is performed in all tests, instead, we only need to move some vertices to their new positions in order to construct an interface-fitted mesh at each time step. On the other hand, since the Stokes flow is governed by the transient Stokes equations that is obtained by simply dropping the convection term, $(\mathbf{u}_f \cdot \nabla)\mathbf{u}_f$, from Navier–Stokes Equations (1) and (2), its corresponding interface-fitted fictitious domain finite element approximation in the sense of decoupled linearization can be easily developed from (53)–(56) by simply dropping convection terms, $(\mathbf{u}_h^{n+1,m+1} \cdot \nabla)\mathbf{u}_h^{n+1,m}$, $(\mathbf{u}_h^{n+1,m} \cdot \nabla)\mathbf{u}_h^{n+1,m+1}$ and $(\mathbf{u}_h^{n+1,m} \cdot \nabla)\mathbf{u}_h^{n+1,m}$, from (53).

In all numerical tests, the computational domain $\Omega = [0, 2] \times [0, 2]$ (i.e., $L = 2$ and $D = 2$), densities of the fluid and rigid particle are the same as $\rho_f = \rho_s = 1$ for cases of neutrally buoyant particles. Moreover, the fluid viscosity $\mu_f = 1$, and the shear rate of the fluid $G_s = 1$. Therefore, the velocity on Γ_1 and Γ_3 , defined as boundary conditions in (4) and (5), satisfies $U_0/2 = G_s D/2 = 1$.

4.1. Circular Particle

First of all, we consider the case of a circular particle with radius $r = 0.15$. The mass center of the circular particle is initially located at $(1, 1)$. The spatial mesh size is $h = 1/40$ and $h = 1/60$, and the time step is $\Delta t = 0.05$.

It is known that the relationship between the angular speed and the rotating angle of elliptical particles in an unbounded shear Stokes flow is as follows [11]

$$\omega = -G_s \frac{r_a^2 \sin^2 \theta + r_b^2 \cos^2 \theta}{r_a^2 + r_b^2}. \quad (57)$$

The theoretical angular velocity given by the above equation is related to the ratio of r_a and r_b , independent of their specific values. Nevertheless, the values of r_a and r_b used in the numerical experiment should be as small as possible as Jeffery's solution is derived under the assumption that the range of the domain is much larger than the particle size so that to minimize the influence of solid particles on the fluid. Thus, the angular speed of a circular particle is about $-G_s/2 = -0.5$ in the case of $r_a = r_b = 0.15$.

Figure 4 shows snapshots of the velocity field and the pressure field surrounding the circular particle obtained by moving-mesh Algorithms 1 and 2 with $h = 1/40$ and $h = 1/60$, respectively. It can be observed that the pressure near the fluid-particle interface is higher on the upstream side and lower on the downstream side, since the particle locates at the center of the plane shear flow.

The plot of the angular speed versus the angular displacement is reported in Figure 5. It can be observed that the circular particle inside the shear flow eventually reaches a steady state. The obtained stable angular velocity is -0.4948 with $h = 1/40$ and -0.4945 with $h = 1/60$ by Algorithm 1, and -0.4946 with $h = 1/40$ and -0.4947 with $h = 1/60$ by Algorithm 2, respectively. They are all in good agreement with Jeffery's solution (57).

Table 2 and Figure 6 present angular speeds obtained by two moving mesh algorithms with different radii of circular particles when $h = 1/60$. Observations indicate that as the radius of circular particles decreases, the angular velocity tends to approach -0.5 . The inverse process of this behavior can be ascribed to the increasingly subtle influence of particles on the surrounding fluid due to the increasing particle size.

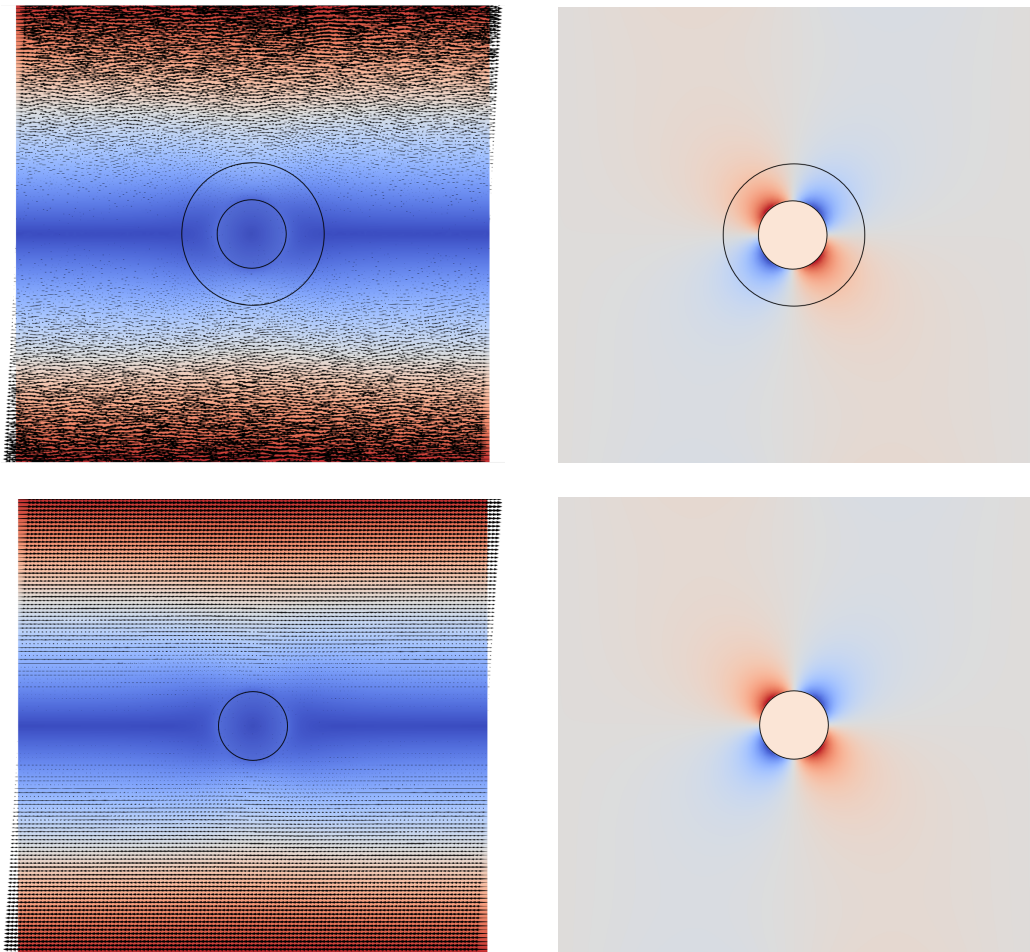


Figure 4. Circular particle: the velocity field (Algorithm 1: **top-left**; Algorithm 2: **bottom-left**) and pressure field (Algorithm 1: **top-right**; Algorithm 2: **bottom-right**).

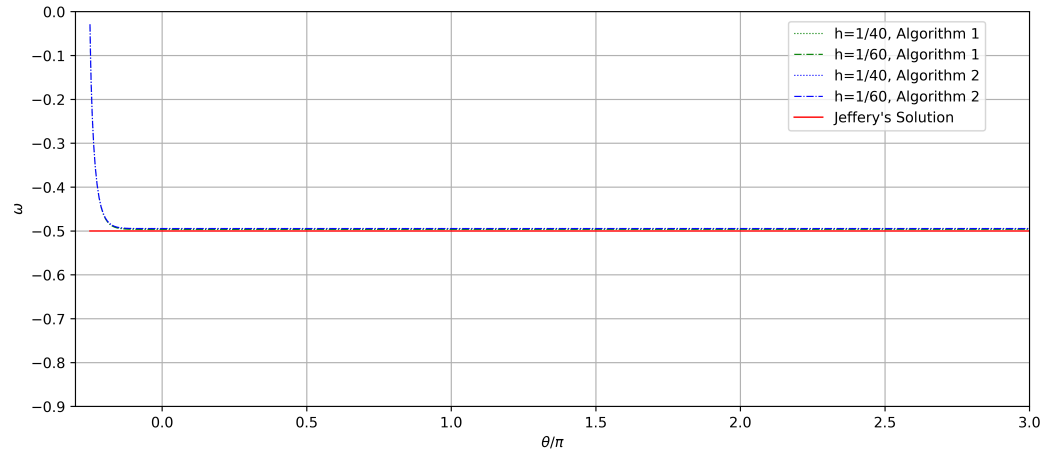


Figure 5. Circular particle: the angular speed versus the angular displacement.

Table 2. Circular particle: angular speeds obtained by two moving mesh algorithms with different radii of circular particles.

Radius r	0.100	0.125	0.150
Algorithm 1	−0.4971	−0.4967	−0.4945
Algorithm 2	−0.4975	−0.4963	−0.4947

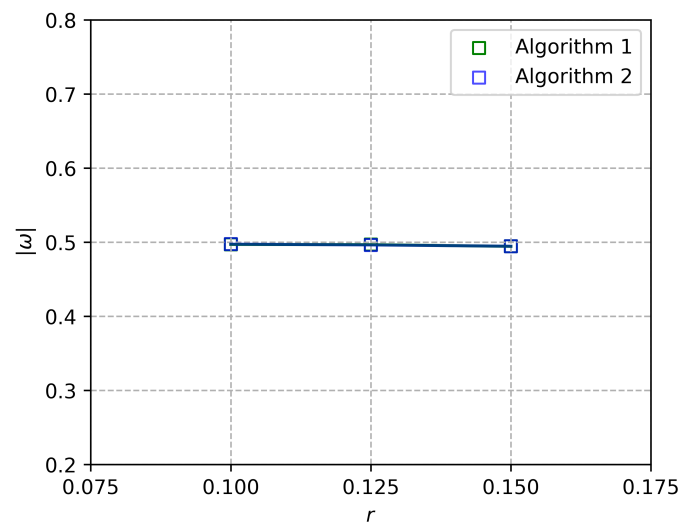


Figure 6. Circular particle: angular speeds obtained by two moving mesh algorithms with different radii of circular particles.

4.2. Elliptical Particle

Next, we consider an elliptical particle with semi-major axes $r_a = 0.2$ and $r_b = 0.1$. The particle's mass center is initially located at $(1, 1)$ with the inclination angle $\theta = \pi/4$ toward the positive x_1 -axis direction. The other parameters are the same as those used in the case of circular particles.

According to (57), while aligning along the x_1 axis, the elliptical particle reaches the minimum angular velocity of -0.2 since it is almost parallel to the direction of fluid velocity, resulting in the minimum hydrodynamic torque experienced. On the other hand, while aligning with the x_2 axis, the particle attains the maximum angular velocity of -0.8 since it is nearly perpendicular to the direction of fluid velocity, leading to the maximum undergoing hydrodynamic torque.

Figure 7 shows snapshots of the velocity field and the pressure field surrounding elliptical particles obtained by moving-mesh Algorithms 1 and 2 with $h = 1/40$ and $h = 1/60$, respectively. The same phenomenon, i.e., the pressure near the fluid-particle interface being higher on the upstream side and lower on the downstream side, can also be observed in this case.

The angular speeds obtained by the proposed numerical method are also compared with Jeffery's solution in Figure 8, illustrating that as the angle changes from $k\pi$ to $k\pi + \pi/2$, the major axis of the elliptical particle progressively aligns perpendicular to the direction of fluid velocity. This alignment leads to an augmentation in hydrodynamic torque, concurrently increasing the angular velocity. Conversely, as the angle changes from $k\pi + \pi/2$ to $k\pi + \pi$, the major axis of the elliptical particle gradually aligns parallel to the direction of fluid velocity. As a consequence, there is a reduction in hydrodynamic torque, causing a gradual decrease in the angular velocity. The angular velocities obtained by Algorithm 1 lie in the range of $[-0.8001, -0.1917]$ when $h = 1/40$ and of $[-0.7958, -0.1932]$ when $h = 1/60$, while ranging from -0.7899 to -0.2000 when $h = 1/40$ and from -0.7933 to -0.1962 when $h = 1/60$ if Algorithm 2 is applied. They all agree well with Jeffery's solution (57).

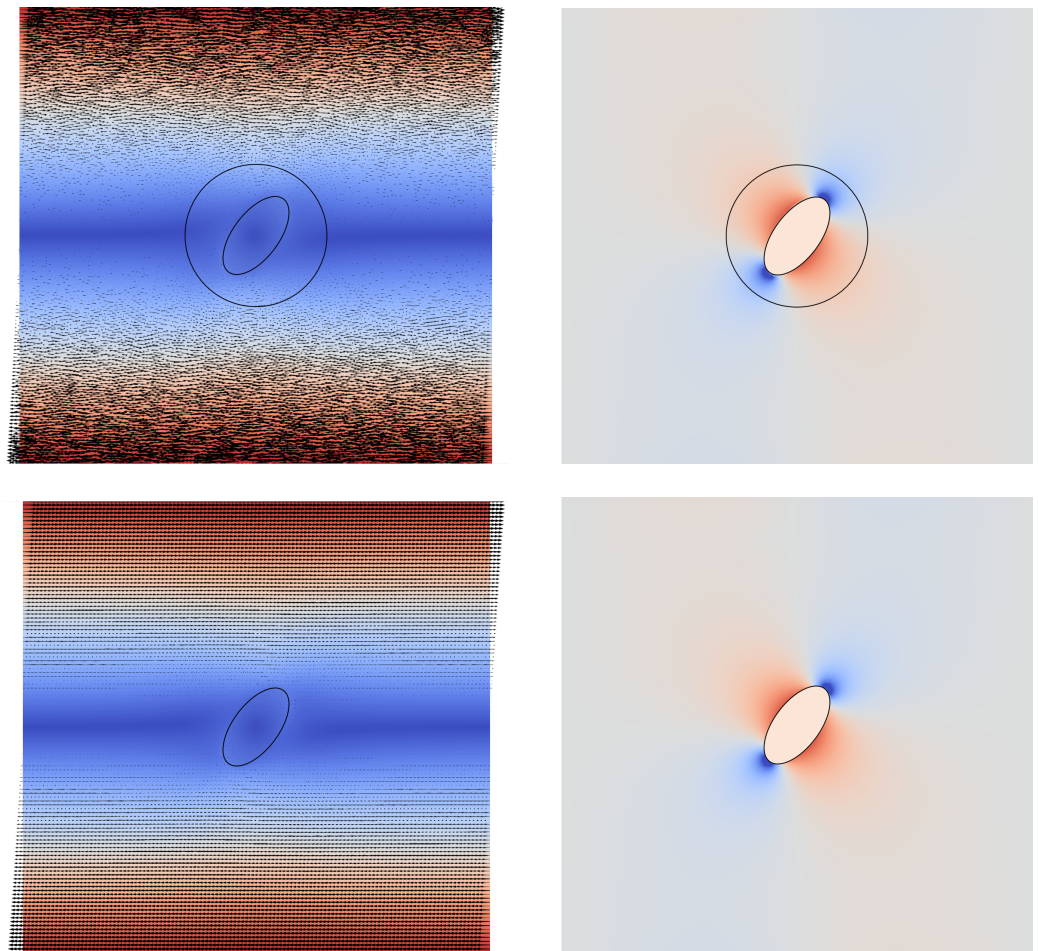


Figure 7. Elliptical particle: the velocity field (Algorithm 1: **top-left**; Algorithm 2: **bottom-left**) and the pressure field (Algorithm 1: **top-right**; Algorithm 2: **bottom-right**), where the color change from red to blue denotes a decreasing magnitude of the shown quantity.

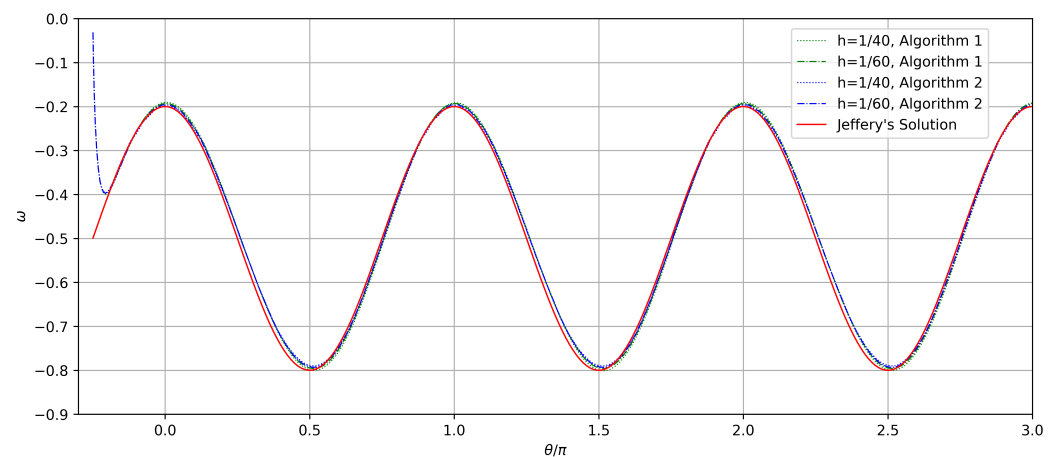


Figure 8. Elliptical particle: the angular speed versus the angular displacement.

5. Conclusions and Future Work

In this paper, we propose an interface-fitted fictitious domain finite element method for fluid–rigid particle interaction problems. A main ingredient of the approach is to introduce a unified velocity defined in the entire domain based on the fictitious domain method to eliminate the requirement of a fixed number of nodes in the fluid subdomain that is usually required by the ALE method. Another main ingredient is to use an interface-fitted mesh to

capture the interface of fluid and rigid particles to improve the accuracy of the numerical simulation. Both ingredients are based upon ideas that an interface-fitted mesh may lead to a more accurate numerical solution, and, an interface-fitted mesh generation algorithm in a fixed domain is easier to develop in contrast to that in a time-dependent domain.

For interface-fitted mesh algorithms, the classical ALE method usually makes the fluid mesh move by solving a PDE-type of ALE mapping that subjects to the Dirichlet boundary condition given by the structural displacement on the interface, which does not support structural motions with large displacement/deformation, as they require a fixed number of mesh nodes/elements in the fluid domain. Otherwise, if the fluid mesh quality becomes too poor due to a large structural motion, then the fluid domain has to be re-meshed at each time step to resume the ALE-like computations on the newly generated mesh which, however, no longer holds a fixed number of mesh nodes in the fluid domain, resulting in additional interpolation errors. But there is no such obstacle in our developed numerical method. In the future, more attempts at simulating realistic FSI problems with large structural displacement or deformation will be studied by following similar ideas presented in this paper.

Author Contributions: Conceptualization, C.W. and P.S.; Data curation, Y.L.; Investigation, Y.L.; Methodology, C.W. and P.S.; Project administration, P.S.; Software, Y.L.; Supervision, P.S.; Validation, Y.L.; Visualization, C.W.; Writing—original draft, C.W.; Writing—review & editing, P.S. All authors have read and agreed to the published version of the manuscript.

Funding: C. Wang and Y. Liang were partially supported by the National Natural Science Foundation of China grant 12171366, P. Sun was partially supported by a grant from the Simons Foundation (MPS-706640, PS).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data sharing not applicable. No new data were created or analyzed in this study. Data sharing is not applicable to this article.

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

References

1. Hron, J.; Turek, S. *A Monolithic FEM/Multigrid Solver for an ALE Formulation of Fluid-Structure Interaction with Applications in Biomechanics*; Springer: Berlin/Heidelberg, Germany, 2006.
2. Barker, A.; Cai, X.C. Scalable parallel methods for monolithic coupling in fluid–structure interaction with application to blood flow modeling. *J. Comput. Phys.* **2010**, *229*, 642–659. [\[CrossRef\]](#)
3. Yu, Z. A DLM/FD method for fluid/flexible-body interactions. *J. Comput. Phys.* **2005**, *207*, 1–27. [\[CrossRef\]](#)
4. Yu, Z.; Yang, Y.; Lin, J. Lubrication Force Saturation Matters for the Critical Separation Size of the Non-Colloidal Spherical Particle in the Deterministic Lateral Displacement Device. *Appl. Sci.* **2022**, *12*, 2733. [\[CrossRef\]](#)
5. Lou, Q.; Ali, B.; Rehman, S.U.; Habib, D.; Abdal, S.; Shah, N.A.; Chung, J.D. Micropolar Dusty Fluid: Coriolis Force Effects on Dynamics of MHD Rotating Fluid When Lorentz Force Is Significant. *Mathematics* **2022**, *10*, 2630. [\[CrossRef\]](#)
6. Rehman, S.; Muhammad, N.; Alshehri, M.; Alkarni, S.; Eldin, S.M.; Shah, N.A. Analysis of a viscoelastic fluid flow with Cattaneo–Christov heat flux and Soret–Dufour effects. *Case Stud. Therm. Eng.* **2023**, *49*, 103223. [\[CrossRef\]](#)
7. Qureshi, M.Z.A.; Faisal, M.; Raza, Q.; Ali, B.; Botmart, T.; Shah, N.A. Morphological nanolayer impact on hybrid nanofluids flow due to dispersion of polymer/CNT matrix nanocomposite material. *AIMS Math.* **2023**, *8*, 633–656. [\[CrossRef\]](#)
8. Tao, C.; Kutcho, B.G.; Rosenbaum, E.; Wu, W.T.; Massoudi, M. Steady Flow of a Cement Slurry. *Energies* **2019**, *12*, 2604. [\[CrossRef\]](#)
9. Slanina, F. Movement of spherical colloid particles carried by flow in tubes of periodically varying diameter. *Phys. Rev. E* **2019**, *99*, 012604. [\[CrossRef\]](#)
10. Pan, T.W.; Zhao, S.; Niu, X.; Glowinski, R. A DLM/FD/IB method for simulating compound vesicle motion under creeping flow condition. *J. Comput. Phys.* **2015**, *300*, 241–253. [\[CrossRef\]](#)
11. Jeffery, G.B. The motion of ellipsoidal particles immersed in a viscous fluid. *Proc. R. Soc. Lond. Ser. A Contain. Pap. Math. Phys. Character* **1922**, *102*, 161–179.
12. Ding, E.; Aidun, C. The dynamics and scaling law for particles suspended in shear flow with inertia. *J. Fluid Mech.* **2000**, *423*, 317–344. [\[CrossRef\]](#)

13. Lundell, F.; Carlsson, A. Heavy ellipsoids in creeping shear flow: Transitions of the particle rotation rate and orbit shape. *Phys. Rev. E* **2010**, *81*, 016323. [[CrossRef](#)]
14. Pasquino, R.; D'Avino, G.; Maffettone, P.L.; Greco, F.; Grizzuti, N. Migration and chaining of noncolloidal spheres suspended in a sheared viscoelastic medium. Experiments and numerical simulations. *J.-Non-Newton. Fluid Mech.* **2014**, *203*, 1–8. [[CrossRef](#)]
15. Marchuk, G.I.; Brown, A.A. *Methods of Numerical Mathematics*; Springer: Berlin/Heidelberg, Germany, 1982; Volume 2, pp. 116–121.
16. Glowinski, R.; Pan, T.; Periaux, J. A Lagrange multiplier fictitious domain method for the numerical simulation of incompressible viscous flow around moving rigid bodies: (1) case where the rigid body motions are known a priori. *C. R. Acad. Des Sci. Ser.-Math.* **1997**, *324*, 361–369. [[CrossRef](#)]
17. Glowinski, R.; Pan, T.; Hesla, T.; Joseph, D.; Periaux, J. A fictitious domain approach to the direct numerical simulation of incompressible viscous flow past moving rigid bodies: Application to particulate flow. *J. Comput. Phys.* **2001**, *169*, 363–426. [[CrossRef](#)]
18. Boffi, D.; Gastaldi, L. A fictitious domain approach with Lagrange multiplier for fluid–structure interactions. *Numer. Math.* **2017**, *135*, 711–732. [[CrossRef](#)]
19. Hwang, W.; Hulsen, M.; Meijer, H. Direct simulation of particle suspensions in sliding bi-periodic frames. *J. Comput. Phys.* **2004**, *194*, 742–772. [[CrossRef](#)]
20. Wang, Y.; Jimack, P.K.; Walkley, M.A. A one-field monolithic fictitious domain method for fluid–structure interactions. *Comput. Methods Appl. Mech. Eng.* **2017**, *317*, 1146–1168. [[CrossRef](#)]
21. Auricchio, F.; Boffi, D.; Gastaldi, L.; Lefieux, A.; Reali, A. On a fictitious domain method with distributed Lagrange multiplier for interface problems. *Appl. Numer. Math.* **2015**, *95*, 36–50. [[CrossRef](#)]
22. Wan, D.; Turek, S. Fictitious boundary and moving mesh methods for the numerical simulation of rigid particulate flows. *J. Comput. Phys.* **2007**, *222*, 28–56. [[CrossRef](#)]
23. Brgers, C. A Triangulation Algorithm for Fast Elliptic Solvers Based on Domain Imbedding. *SIAM J. Numer. Anal.* **1990**, *27*, 1187–1196. [[CrossRef](#)]
24. Alauzet, F. A changing-topology moving mesh technique for large displacements. *Eng. Comput.* **2014**, *30*, 175–200. [[CrossRef](#)]
25. Takizawa, K.; Bazilevs, Y.; Tezduyar, T.E. Mesh Moving Methods in Flow Computations with the Space-Time and Arbitrary Lagrangian–Eulerian Methods. *J. Adv. Eng. Comput.* **2022**, *6*, 85. [[CrossRef](#)]
26. Yang, K.; Sun, P.; Wang, L.; Xu, J.; Zhang, L. Modeling and simulations for fluid and rotating structure interactions. *Comput. Methods Appl. Mech. Eng.* **2016**, *311*, 788–814. [[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.