Published in final edited form as:
Comput Methods Appl Mech Eng. 2013 December 1; 267: . doi:10.1016/j.cma.2013.07.019.

Modified Immersed Finite Element Method For Fully-Coupled Fluid-Structure Interations

Xingshi Wang and **Lucy T. Zhang**[1]
JEC 2049, Department of Mechanical, Aerospace, and Nuclear Engineering, Rensselaer Polytechnic Institute, Troy, NY 12180

Abstract

In this paper, we develop a "modified" immersed finite element method (mIFEM), a non-boundary-fitted numerical technique, to study fluid-structure interactions. Using this method, we can more precisely capture the solid dynamics by *solving* the solid governing equation instead of imposing it based on the fluid velocity field as in the original immersed finite element (IFEM). Using the IFEM may lead to severe solid mesh distortion because the solid deformation is been over-estimated, especially for high Reynolds number flows. In the mIFEM, the solid dynamics is solved using appropriate boundary conditions generated from the surrounding fluid, therefore produces more accurate and realistic coupled solutions. We show several 2-D and 3-D testing cases where the mIFEM has a noticeable advantage in handling complicated fluid-structure interactions when the solid behavior dominates the fluid flow.

Keywords

Immersed Finite Element Method; fluid-structure interaction; compressibility

1 Introduction

Analyzing and understanding complex physical phenomena involving fluid-structure interactions require reliable, robust, and efficient modeling techniques. In the past few decades, numerous research efforts have been directed toward methods development of fluid-structure interactions.

In general, the numerical approaches to model fluid-structure interactions can be classified into two categories: boundary-fitted mesh and non-boundary-fitted mesh. One of the most well-known and popularly used boundary-fitted mesh methods is the Arbitrary Lagrangian Eulerian (ALE) [1–6]. It can handle complicated fluid-solid interface, but it also requires a process of re-meshing or mesh-updating, in which the whole domain or part of the domain is re-discretized spatially. This process can be computationally expensive, and more importantly might reduce the accuracy at the interface due to the transferring of the solutions from the degenerated mesh to the new mesh. Recent advances have been made toward resolving the interfacial solution accuracies by using non-matching interfaces [7,8]. Research done by Tezduyar and Takizawa [9–11] demonstrated that a robust and efficient

[1]zhangLucy@rpi.edu, Tel: (518) 276-6914, Fax: (518) 276-2623 .

solutions can be obtained using a re-meshing and solution projection technique with a space-time body-fitted approach.

To avoid such re-meshing or mesh-updating procedure, so-called non-boundary-fitted mesh methods are developed, in which the solid object and the background fluid grid are meshed independently. The solid described using a Lagrangian frame of reference can move freely on top of a fixed fluid mesh that is described using Eulerian frame of reference. However, interpolations are required to couple these two independent meshes. The interpolation process inevitably introduces a "layer" of interface instead of a distinct boundary interface. Some sharp interface treatments are proposed [12,13], which allow the local hydrodynamics at the interface can be more precisely resolved. Among the non-boundary-fitted mesh methods, one of the most noteworthy efforts is the immersed boundary method [10], which was initially proposed by Peskin to study the blood flow around heart valves [14–20]. The immersed boundary method inspired researchers around the world to further develop and enhance the accuracy and efficiency of the method. One of them is the immersed interface method [21,22] where LeVeque and Li obtained second-order accuracy in Peskin's immersed boundary method by imposing a derived second-order solution for Stokes fluid at the interface. This method was later extended to solve more complicated Navier-Stokes fluid flows [23–27]. Other methods include the extended immersed boundary method [28] which uses the finite element approach to calculate solid boundary nodal forces and the immersed boundary finite element method [29,30] which describes an immersed solid boundary in a fluid domain that is modeled using finite elements.

The immersed finite element method [31–38] is another method that extends the immersed boundary method to represent the background viscous fluid with an unstructured finite element mesh and nonlinear finite elements for the immersed deformable solid. Similar to the immersed boundary method, the fluid domain is defined on a fixed Eulerian grid and the solid domain is constructed independently with a Lagrangian mesh. The major difference between these two immersed approaches, however, is that with the immersed finite element method, the solid material can be described with a detailed constitutive model such as the linear elastic, hyperelastic, or viscoelastic. It is no longer limited to just a boundary layer, instead, it can occupy a volume space in the entire computational domain. Such realistic material representations are particularly useful when large deformation or movement of the material must be realized and its affected hydrodynamics are altered through interactions with the surrounding fluid. A semi-implicit two-way coupling algorithm [39] is later developed. As a result of the semi-implicit coupling, the numerical difficulty dealing with large fluid-solid density difference is resolved. The coupled solution converges better, faster, and larger time step size can be used for significantly higher computational efficiency.

Even though the IFEM algorithm has been improved over time to resolve its numerical stability and convergence issues, problems still remain when the solid dynamics needs to be captured accurately. In the case involving large solid deformation or high Reynolds number flows, the solid may behave unrealistically when its velocity and displacement are predicted based on the overlapping fluid velocity field through interpolation. Given that the fluid and the solid dynamics are inherently different from one another, it is not surprising a slight mismatch would result at each time step. However, this over-estimated solution accumulates over time causing severe solid mesh distortion. Moreover, if the background fluid is incompressible, the solid that follows the fluid is also constrained to be incompressible or nearly incompressible, where the volume change must be neglected. It further decreases the accuracy of solid solution and limits the numerical applications to which the method can apply. It is particularly critical if wave propagation in the solid is important.

In this paper, we propose a newly developed "modified immersed finite element method" (mIFEM) to resolve the above mentioned issues and to achieve more accurate coupled solutions. The key idea of the mIFEM algorithm is that it lets the solid lead the fluid-structure interaction and makes the fluid underneath to follow the solid motion. The internal solid deformation are accurately *solved* by a set of solid governing equation with appropriate boundary conditions instead of being *evaluated* using the interpolated fluid velocity field. Doing so also allows the solid to have its natural compressibility instead of obeying the fluid compressibility condition.

The paper is organized as follows. We first briefly review the immersed finite element method in Section 2. The key assumptions and their corresponding restrictions will also be discussed. In Section 3, we will present the mIFEM algorithm and the detailed derivations. To demonstrate the capabilities of this newly developed mIFEM algorithm, three testing cases are studied in Section 4. Finally, the conclusions are drawn in Section 5.

2 Review of the Immersed Finite Element Method

In the IFEM, the fluid exists everywhere in the computational domain. One or more deformable solid structures are immersed in the fluid, which occupy a finite domain, Ω^s, as illustrated in Fig. 1. The part of the fluid volume that overlaps with the solid domain is called the "artificial" fluid, $\bar{\Omega}$, since it does not physically exist. The fluid outside the solid domain is the real fluid, Ω^f. The fluid domain and the solid domain together construct the entire computational domain, $\Omega = \Omega^f \cup \Omega^s$. The solid boundary Γ^s is also the common interface intersecting with the fluid domain, called the fluid-structure interface $\Gamma^{FSI} \equiv \Gamma^s$. Three major assumptions are made in the IFEM algorithm: 1) the fluid (real and artificial) fills the entire computational domain; 2) both the real and artificial fluids are incompressible; 3) the entire solid and the artificial fluid obey no-slip condition where the solid motion follows that of the artificial fluid.

For completeness, here we briefly review the original IFEM algorithm with semi-implicit coupling scheme (hereon will be referred to as IFEM). The detailed derivations can be further found in our pervious papers [36,39].

The governing equations for the entire computational domain Ω which includes the real Ω^f and the artificial fluid domains $\bar{\Omega}$, are the Navier-Stokes continuity and momentum equations for incompressible flows:

$$\nabla \cdot \mathbf{v}^f = 0 \quad (1)$$

$$\bar{\rho}\left(\mathbf{v}^f_{,t} + \mathbf{v}^f \cdot \nabla \mathbf{v}^f = -\nabla p^f + \mu \nabla^2 \mathbf{v}^f + \mathbf{f}^{FSI,f} + \bar{\rho}\,\mathbf{g} \quad \text{in} \quad \Omega, \quad (2)$$

where $\bar{\Omega}$ is defined as:

$$\bar{\rho} = \rho^f + \left(\rho^s - \rho^f\right) I(\mathbf{x}). \quad (3)$$

The $\mathbf{v}$ and p are the state variables: velocity and pressure, respectively. $\mathbf{f}^{FSI,f}$ is the fluid-structure interaction force. $\mathbf{g}$ is the external body force. ρ^f and ρ^s are the densities of the fluid and the solid, respectively. Superscript f represents fluid variables and s for solid variables. The indicator function I is used to identify the artificial fluid from the real fluid. It is set to 1 in the artificial fluid domain and 0 in the real fluid domain; and varies from 0 to 1 at and near the fluid-structure interface. Since the artificial fluid overlaps with the solid domain, the

indicator field must be updated along with the motion of the solid. More information of the indicator function and its detailed calculation can be found in [39-41].

The $\mathbf{f}^{FSI,f}$ in the momentum equation (2) is defined as the fluid-structure interaction force that represents the viscous effects due to the existence of the solid in the fluid domain. This fluid-structure interaction force is first evaluated in the solid domain, $\mathbf{f}^{FSI,s}$ as,

$$\mathbf{f}^{FSI,s} = \nabla \cdot \sigma^s - \nabla \cdot \sigma^f \quad \text{in} \quad \Omega^s. \quad (4)$$

Here, σ^s is the solid stress evaluated based the solid constitutive law as a function of the solid deformation; σ^f is the fluid stress interpolated onto the solid domain from the previous time solution. This interaction force is to effectively subtract the artificial fluid's viscous force out of the solid's internal force. Once $\mathbf{f}^{FSI,s}$ is calculated, it is then distributed onto the fluid domain as $\mathbf{f}^{FSI,f}$ using an interpolation function. Noting that $\mathbf{f}^{FSI,f}$ is a local force that only exists in the artificial fluid domain $\bar{\Omega}$. It is easy to notice that this equation only involves the solid stress, where the *dynamics* of the solid is entirely "controlled" by the fluid.

Once the state variables, $\mathbf{v}^f$ and p are solved from Eqs. (1) and (2) for the entire computational domain, the solid velocity field $\mathbf{v}^s$ is directly mapped from the fluid velocity field $\mathbf{v}^f$ using either 1st order forward Euler scheme or 3rd order Runge-Kutta scheme [42,43]. The algorithm flowchart of IFEM is shown in Fig. 2(a).

3 Modified Immersed Finite Element Method

3.1 Rationale of the mIFEM algorithm

The purpose of developing and implementing the modified immersed finite element method (mIFEM) is to remove some constraints imposed by the assumptions made in the IFEM algorithm. As mentioned previously, in the IFEM, the solid deforms along with the artificial fluid. The velocity of every single nodal point in the solid domain is interpolated directly from the velocity of the artificial fluid, i.e. $\mathbf{v}^s = \mathbf{v}^f$ in Ω^s. The underlying assumption here is that the solid behavior is predicted by the fluid's equations, whereas the effects from the solid are not accounted for until the next time step through the interaction force. This scheme relies heavily on the fluid's equations yielding a fairly reasonable solid solution and a small time step is often required. It would not cause any fundamental difference if the solid behavior is similar to the fluid. However, in cases where the solid behavior is more dominant, it is not appropriate to approximate the internal nodal solid movement based on the fluid velocity. The fluid velocity often leads to unrealistic solid deformation. Sometimes, it even causes severe distortion of the solid mesh. Similarly, if the fluid continuity equation governs the artificial fluid, then it also governs the solid. The compressibility constraint of the solid is forced to be the same as the fluid, which is often unrealistic.

In the mIFEM, we reverse the assumption made before where the solid follows the artificial fluid, instead, the solid motion or the dynamics is *solved* and we let the artificial fluid to follow the solid, i.e. $\mathbf{v}^f = \mathbf{v}^s$ in Ω^s. Doing so allows the solid motion and deformation to be precisely controlled by the solid governing equations, constitutive laws, and the boundary conditions derived from the fluid solution. To ensure the artificial fluid moves together with the solid in order to still satisfy the no-slip boundary condition, it is set to have properties that are the same as the solid so that when the fluid-structure interaction force is applied on the artificial fluid domain, the velocity of the solid is reproduced in the artificial fluid region. One can think that the artificial fluid is now governed by the solid equation. In this context, it is not an unreasonable assumption because the artificial fluid physically does not exist. Therefore it can move wherever it needs to as long as it satisfies the equilibrium of the

system. In this case, the solid sees the fluid only through the actual interface or the solid boundary. All the interior deformation is solved by the solid governing equation. This would yield a much more accurate solid solution. In the meantime, the artificial fluid can be manipulated to have the same compressibility as the solid and the solid compressibility is no longer compromised.

Based on the above logic, a few key features must be represented in the formulation:

Given the fluid velocity and pressure fields from the initial or previous time step, interpolate the velocity or pressure data onto the solid boundary or the interface Γ^{FSI}. Use the boundary condition and the solid governing equation to solve for the solid acceleration, velocity, and displacement.

Evaluate the fluid-structure interaction force $\mathbf{f}^{FSI}$ using the fluid viscous stress and internal solid stress.

With the known $\mathbf{f}^{FSI}$, proper boundary conditions, and properties applied onto the real and artificial fluid, the Navier-Stokes equations are solved in the entire fluid domain.

In the section below, we will show the detailed formulation and derivation of the governing equation for each of the domains: solid domain, artificial fluid domain, and the entire fluid domain.

3.2 Derivation of the mIFEM algorithm

3.2.1 Solid domain—One of the major changes in the mIFEM is that the solid motion and deformation are solved base on the solid governing equation with the solid constitutive law and boundary conditions. A dynamic solid equation is:

$$\rho^s \ddot{u}_i^s = \sigma_{ij,j}^s \quad \text{in} \quad \Omega^s, \quad (5)$$

where $\mathbf{u}^s$ is the solid displacement,$\ddot{u}^s$ is the solid acceleration, $\sigma^s$is the solid stress. The solid stress σ^s is described based on the constitutive law as a function of the solid strain tensor ε^s and material properties:

$$\sigma_{kl}^s = c_{ijkl}\varepsilon_{ij}^s + \eta_{ijkl}\varepsilon_{ij,t}^s \quad \text{in} \quad \Omega^s, \quad (6)$$

where $\varepsilon_{ij}^s = \frac{1}{2}\left(u_{i,j}^s + u_{j,i}^s\right)$. Different combinations of c_{ijkl} and η_{ijkl} provide various choices of solid material constitutive laws such as linear elastic, visco-elastic, and hyper-elastic, etc. In this algorithm, we implemented linear elastic and visco-elastic materials for small strains and hyperelastic materials with Mooney-Rivlin model for large strains. The details of the constitutive laws for these particular materials can be found in [44] and [31].

The boundary condition of the solid domain is the actual interface that the solid 'sees' the fluid. The solid boundary can be applied as either the Dirichlet boundary condition on Γ^{sq} and/or Neumann boundary condition on Γ^{sh}:

$$u_i^s = q_i \quad \text{on} \quad \Gamma^{sq}, \quad (7)$$

$$\sigma_{ij}^s n_j = h_i \quad \text{on} \quad \Gamma^{sh}. \quad (8)$$

To ensure the existence and the uniqueness of the solid solution, we further require $\Gamma^s = \Gamma^{sq} \bigcup \Gamma^{sh}$ and $\Gamma^{sq} \neq \emptyset$.

Based on the no-slip and traction boundary conditions imposed at the fluid-solid interface, the boundary conditions can be calculated through the interpolation function using the known fluid velocity and pressure fields from the previous time step:

$$q_i = \left[\int_\Omega v_i^f \phi\left(\mathrm{x} - \mathrm{x}^s\right) d\Omega\right] \Delta t \quad \text{on} \quad \Gamma^{sq}, \quad (9)$$

$$h_i = \left[\int_\Omega \sigma_{ij}^f \phi\left(\mathrm{x} - \mathrm{x}^s\right) d\Omega\right] n_j \quad \text{on} \quad \Gamma^{sh}. \quad (10)$$

Here, ϕ is the interpolation function that is a function of the distance of a fluid grid point x and a solid point x^s; Δt is the time step size and **n** is the outward normal of the fluid-structure interface.

For dynamic systems, the initial condition should also be known in order to solve for the solid equation in Eq. (5), the initial displacement and the velocity conditions are:

$$u_i^s\left(\mathrm{x}^s\right) = u_{0i}^s\left(\mathrm{x}^s\right) \quad \text{in} \quad \Omega^s, \quad (11)$$

and

$$u_{i,t}^s\left(\mathrm{x}^s\right) = \dot{u}_{0i}^s\left(\mathrm{x}^s\right) \quad \text{in} \quad \Omega^s. \quad (12)$$

If the solid starts from a resting position, the solid nodal velocity and displacement are initially set to zero.

This set of solid equations described above is discretized using finite elements and solved numerically using α-method which is an unconditionally stable scheme with second-order accuracy [45]. Because the solid domain is defined and discretized on a Lagrangian mesh, no continuity equation is specifically required and the mass balance is satisfied automatically.

3.2.2 Artificial fluid domain—The goal of the mIFEM is to let the artificial fluid domain to behave like the solid as much as possible because the artificial fluid physically does not exist, so it should mimic the solid. The first thing is to enforce $\mathbf{v}^f = \mathbf{v}^s$ in $\bar{\Omega}$ so that the artificial fluid domain can produce the same velocity as the solid. Again, no-slip boundary condition is applied at the fluid-structure interface Γ^{FSI} and we also assume that there does not exist any mass exchange between the fluid and solid domain.

First, we write the continuity equation for the artificial fluid, which should allow compressibility. Here, we let the fluid to be pseudo-compressible so that a proper compressibility is introduced. Letting the artificial fluid domain to be pseudo-compressible is necessary in two aspects: 1) to ensure the artificial fluid velocity to be the same as the solid velocity and 2) when any volume change in the solid has to be considered. In fact, all solid is compressible or at least nearly incompressible. The compressibility of the material is usually represented by the bulk modulus, κ, where a large value of κ means it is hard to be compressed. The appoximated bulk modulus is $160 \times 10^9 Pa$ for steel, $10^7 \sim 10^8$ for rubber, and $10^4 \sim {}^5 Pa$ for certain biological tissues. On the other hand, the bulk modulus is around $2 \sim 10^9 Pa$ for water and $10^5 Pa$ for air. From this wide range of values, we can see that the volume change of the solid domain must be considered under many circumstances. Even if the solid material can be considered as nearly incompressible, a volume correction algorithm must be introduced in order to ensure its compatibility with the incompressible fluid [46]. Using such a volume correction algorithm helps in limiting the amount of volume change,

but errors are inevitably introduced into the solid displacement field, which causes unrealistic solid behaviors [47]. Furthermore, even if the solid volume change can be neglected globally, it is often necessary to consider the local changes of the solid volume, such as in the case of wave propagation in the solid.

Here, we assume that the density in the artificial fluid domain is only a function of time and the continuity equation can be written as:

$$\frac{1}{\kappa}\frac{\partial p}{\partial t}+v^f_{i,i}=0 \quad \text{in} \quad \bar{\Omega}\,. \quad (13)$$

where κ is the bulk modulus. Since we want the compressibility of the artificial fluid to be the same as the solid, we replace κ with κ^s, which is the bulk modulus of the solid. Then the continuity equation becomes:

$$\frac{1}{\kappa^s}\frac{\partial p}{\partial t}+v^f_{i,i}=0 \quad \text{in} \quad \bar{\Omega}\,. \quad (14)$$

Letting the solid and fluid domains share the same bulk modulus ensures the velocity in the artificial fluid domain to be the same or at least very similar to the solid velocity.

The momentum Navier-Stokes equation is:

$$\rho\frac{\partial v^f_i}{\partial t}+\rho v^f_j v^f_{i,j}=\sigma^f_{ij,j}+f^{FSI,f}_i \quad \text{in} \quad \Omega. \quad (15)$$

In the artificial fluid domain, we want the velocity in this domain to be the same as the solid in $\bar{\Omega}$. Therefore, we will replace ρ by ρ^s so that it has the same property as the solid:

$$\rho^s\frac{\partial v^f_i}{\partial t}+\rho^s v^f_j v^f_{i,j}=\sigma^f_{ij,j}+f^{FSI,f}_i \quad \text{in} \quad \bar{\Omega}\,. \quad (16)$$

Here, the velocity v^f is an independent variable, the interaction force $\mathbf{f}^{FSI,f}$ is there to further enforce $\mathbf{v}^f = \mathbf{v}^s$ in $\bar{\Omega}$. This interaction force is first evaluated in the solid domain and distributed onto the fluid domain through the interpolation function as,

$$\mathbf{f}^{FSI,f}=\int_{\Omega^s}\mathbf{f}^{FSI,s}\phi\left(\mathrm{x}-\mathrm{x}^s\right)d\Omega^s. \quad (17)$$

The definition of the fluid-structure interaction force in the solid domain was shown previously in Eq. (4),

$$f^{FSI,s}_i=\sigma^s_{ij,j}-\sigma^f_{ij,j} \quad \text{in} \quad \Omega^s, \quad (18)$$

in which σ^f is the fluid stress.

With the fluid-structure interaction force and appropriate material properties, the artificial fluid domain can produce nearly the same velocity as the solid domain. However, it is still possible that the artificial fluid may not be exactly the same as the solid velocity in the entire artificial fluid domain due to the following reasons:

The Dirichlet boundary condition Eq. (7) is not necessarily applied onto the entire solid boundary, because in some cases Neumann boundary condition may be more appropriate to use as a part of the solid boundary condition. Therefore, the no-slip boundary condition is not always strictly enforced on the entire fluid-structure

interaction surface, which may cause minor differences between the artificial fluid and solid velocities.

Although the artificial fluid domain and the solid domain share similar governing equations, they are solved on the Eulerian and Lagrangian meshes respectively, which inevitably introduces error between the artificial fluid velocity and the solid velocity.

The coupling between the solid and fluid domains involves a pair of interpolation and distribution procedures which may also cause small discrepancies between the artificial fluid and solid velocities.

To compensate the possible accumulative error in enforcing $\mathbf{v}^f = \mathbf{v}^s$ in the artificial fluid domain, we further introduce a correction force, $\mathbf{f}^{\Delta \mathbf{v}}$,

$$\mathbf{f}^{\Delta \mathrm{v}} = \rho^s \left(\frac{D\mathrm{v}^s}{Dt} - \frac{D\mathrm{v}^f}{Dt} \right) \quad \text{in} \quad \Omega^s. \quad (19)$$

The correction force is effectively the difference between the material derivatives of the solid velocity and the artificial fluid velocity. Both errors introduced in the inertial and convective acceleration forces are included. The reason to introduce this correction force is purely an extra enforcement so that $\mathbf{v}^f = \mathbf{v}^s$ is satisfied. It would be zero if the artificial fluid follows the solid exactly. The correction force is added into the fluid structure interaction force as,

$$\mathbf{f}^{FSI,s} = \nabla \cdot \sigma^s - \nabla \cdot \sigma + \mathbf{f}^{\boldsymbol{\Delta} \mathrm{v}} \quad \text{in} \quad \boldsymbol{\Omega}^{\mathbf{s}}. \quad (20)$$

This correction can be done once or iteratively depending on how precisely the solid velocities are reproduced in the artificial fluid domain.

3.2.3 Fluid domain—The fluid domain here includes both the real fluid, Ω^f and the artificial fluid domain $\bar{\Omega}$. As shown in Sec. 3.2.2, for accuracy the artificial fluid is considered as pseudo-compressible, but the real fluid can be either incompressible or compressible. For simplicity, we first assume the real fluid is incompressible, the continuity equation can be written as,

$$\begin{cases} v^f_{i,i} & = 0, \quad \text{in} \quad \Omega^f \\ \frac{1}{\kappa^s} \frac{\partial p^f}{\partial t} + v^f_{i,i} & = 0, \quad \text{in} \quad \bar{\Omega}. \end{cases} \quad (21)$$

And the momentum equations in the fluid domain are,

$$\begin{cases} \rho^f \frac{\partial v^f_i}{\partial t} + \rho^f v^f_j v^f_{i,j} & = \sigma^f_{ij.j}, \quad \text{in} \quad \Omega^f \\ \rho^s \frac{\partial v^f_i}{\partial t} + \rho^s v^f_j v^f_{i,j} & = \sigma^f_{ij,j} + f^{FSI,f}_i, \quad \text{in} \quad \bar{\Omega}. \end{cases} \quad (22)$$

The fluid stress is,

$$\sigma^f_{ij} = -p^f \delta_{ij} + \tau^f_{ij}, \quad (23)$$

and τ^f_{ij} is the viscous stress,

$$\tau^f_{ij} = \mu \left(v^f_{i,j} + v^f_{j,i} \right). \quad (24)$$

Because the entire fluid domain is considered as a background and defined on a fixed Eulerian mesh, the real fluid domain and artificial fluid domain can be combined together and solved simultaneously. The indicator function is used so that we have a general form of the governing equations defined on the entire computational domain. The indicator function, $I(\mathbf{x})$, is a delta function equals 0 in the real fluid domain and 1 in the artificial fluid domain,

$$I(\mathrm{x}) = \begin{cases} 0 & \text{in } \Omega^f \\ 1 & \text{in } \bar{\Omega} \end{cases}. \quad (25)$$

The continuity equation for the entire fluid domain in Eq. (21) can be expressed using the indicator function,

$$\frac{1}{\kappa^s}\frac{\partial p^f}{\partial t} I(\mathrm{x}) + v^f_{i,i} = 0 \quad \text{in} \quad \Omega. \quad (26)$$

Similarly, the momentum equations in both real fluid domain and artificial fluid domain in Eq. (22) are combined together by the indicator function as well,

$$\bar{\rho}\frac{\partial v^f_i}{\partial t} + \bar{\rho}\, v^f_j v^f_{i,j} = \sigma^f_{ij,j} + f^{FSI,f}_i \quad \text{in} \quad \Omega, \quad (27)$$

where $\bar{\rho}$ is defined as

$$\bar{\rho} = \rho^f + \left(\rho^s - \rho^f\right) I(\mathrm{x}) \quad \text{in} \quad \Omega. \quad (28)$$

The velocity and pressure for both the real and the artificial fluids are obtained simultaneously by solving the governing equations Eq. (26) and Eq. (27) at every time step. The governing equations in the entire fluid domain are also discretized by finite element method and solved by Matrix-free Newton Krylov method [48].

It is generally believed that the fluid flow should be considered as compressible when the Mach number is larger than 0.3. Even when the Mach number is relatively small, the compressibility of the fluid plays an important role in some fluid-structure problems such as air flow compressed by piston, pulse wave of a tissue immersed in fluid, and voice production through vocal folds vibration. Here, another set of equations are implemented for compressible fluid governed by idea gas law. Such option makes the mIFEM suitable for solving more complex fluid-structure interaction problems.

For the real fluid that is compressible, the N-S equations are adjusted so that the density can be a function of time. Here the real fluid is assumed to be isotropic ideal gas and the energy equation is avoided. The continuity equations for the real fluid Ω^f and the artificial fluid $\bar{\Omega}$ are:

$$\begin{cases} \frac{\partial \rho^s}{\partial t} + (\rho^s v_i)_{,i} = 0 & \text{in} \quad \bar{\Omega} \\ \frac{\partial \rho^f}{\partial t} + \left(\rho^f v_i\right)_{,i} = 0 & \text{in} \quad \Omega^f \end{cases} \quad (29)$$

Using the indicator function, the continuity equation for the entire fluid domain is written as,

$$\frac{\partial \bar{\rho}}{\partial t} + \left(\bar{\rho}\, v_i\right)_{,i} = 0 \quad \text{in} \quad \Omega. \quad (30)$$

$\bar{\rho}$ was defined in Eq. (28).

The state equations for the artificial fluid and the ideal gas law for real fluid are,

$$\begin{cases} \frac{1}{\rho^s}\frac{\partial \rho^s}{\partial t}=\frac{1}{\kappa^s}\frac{\partial p}{\partial t} & \text{in} \quad \bar{\Omega} \\ p=\rho^f RT & \text{in} \quad \Omega^f, \end{cases} \quad (31)$$

Here, R is the specific gas constant (R = 286.9 $Jkg^{-1}K^{-1}$ for dry air) and T represents the temperature which is a constant.

The momentum equation remains the same

$$\bar{\rho}\frac{\partial v_i^f}{\partial t}+\bar{\rho}\, v_j^f v_{i,j}^f=\sigma_{ij,j}^f+f_i^{FSI,f} \quad \text{in} \quad \Omega, \quad (32)$$

where the fluid stress becomes,

$$\sigma_{ij}^f=-p\delta_{ij}+\mu\left(v_{i,j}^f+v_{j,i}^f\right)+\frac{2}{3}\mu v_{k,k}^f\delta_{ij}. \quad (33)$$

The algorithm for compressible fluid remains the same as the incompressible one except that the governing equations Eq. (30) and Eq. (32) are used.

3.3 Numerical algorithm

Here is a summary of the assumptions and the mIFEM algorithm. The assumptions are:

The fluid is everywhere in the entire computational domain;

The solid is completely immersed in the fluid;

The artificial fluid is pseudo-compressible to accommodate the compressibility of the solid;

No-slip and traction boundary conditions are applied at the fluid-structure interface.

The numerical algorithm is summarized as follows. A flowchart of the mIFEM algorithm is shown in Fig. 2(b).

Solve solid equation on the Lagrangian mesh to obtain solid motion and deformation using Eq. (5); apply solid boundary conditions Eqs. (7) and (8) derived from fluid solutions from the previous time step;

Calculate the fluid-structure interaction force $\mathbf{f}^{FSI}$ using Eq. (20) in the solid domain;

Distribute the fluid-structure interaction force onto the fluid domain using Eq. (17);

Update the indicator field **I** based on the relative position of the solid in the fluid domain;

Solve fluid equations on the Eulerian mesh to obtain velocity and pressure fields: continuity Eq. (26) and momentum Eq. (27) for incompressible fluid, or Eqs. (30) and (32) for compressible flows;

Interpolate fluid velocity and stress onto the solid boundary as its boundary condition using Eqs. (9) and (10), go back to step (1).

Remarks: The solid domain and fluid domain are solved independently and strongly coupled with each other within each time step. As opposed to the original IFEM, the solid dynamic terms, i.e. the inertial and convective terms, in the mIFEM are solved simultaneously in the

current time step instead of using the previous time step solution. It is still considered as a semi-implicit coupling however, due to the fact that the solid boundary conditions Eq. (7) and Eq. (8) still use the fluid solution at the fluid-structure interface from the previous time step. Therefore, the fluid-structure interaction force is evaluated one time step behind compared to all the other terms in the fluid's momentum equation. Such time step mismatch will cause coupling error proportional to the "stiffness" ratio between the two domains [39]. If the solid is very stiff, extra caution may be required to ensure the stability and convergence of the coupled solution. Small time step may be used, or the coupled solutions can be solved iteratively within each time step until the difference between the solid velocity and the artificial fluid velocity becomes negligible, as explained in the previous section.

4 Numerical examples

Four numerical examples are presented in this section. The first example is a 3-D sphere dropping in a channel due to gravity. This particular problem was previously studied using the IFEM algorithm [39]. The purpose of this study is to demonstrate that such a case with large displacement can be handled very easily without any mesh update or re-meshing as would be required if boundary-fitted mesh were used. Here, we repeated this study using mIFEM to illustrate that the mIFEM algorithm is able to capture very accurately the displacement and velocity compared to the theoretical values. The second example is a 2-D deformable leaflet placed in a fluid channel with a constant inflow velocity. The purpose of this study is to make direct comparisons of the mIFEM and the IFEM. The third example is a 2-D deformable oscillated plate in a fluid. It is a solid-initiated fluid-structure interaction problem dominated by the solid motion and deformation. The fourth example is a 3-D flow past a deformable cylinder that are fixed at two ends. The fluid-structure interaction is initiated by the fluid flow and the contributions from both the fluid and the structure to the drag and lift forces as well as the oscillation frequency of the cylinder are equally important.

4.1 3-D sphere dropping in a free fall

In this example, we study a 3-D ball dropping in free fall. The fluid domain is $2\times2\times4\ cm^3$. All the boundaries are considered as no-slip steady walls. The fluid simulated here is air with a density of $10^{-3}g/cm^3$. The 3-D sphere has a density of $1g/cm^3$ with diameter of $0.5cm$ and is initially located at (1, 0, 0). The movement of the ball is driven by gravity of $g = 980cm/s^2$. The sphere is described with linear elastic material with Young's modulus of $E = 10{,}000dyn/cm^2$, and Poisson's ratio of $\nu = 0.49$. The problem setup is illustrated in Figure 3. The fluid domain is discretized with 16, 000 uniform hexagonal elements and 18, 081 nodes, while the solid domain is discretized with 20, 754 tetrahedral elements and 4, 092 nodes.

The snapshots of the simulation are shown in Figure 4. The small vortices near the ball as it falls are clearly captured and the fluid velocity field is developed based on the movement of the sphere as it falls under gravity. The time history of the sphere displacement and velocity are shown in Figure 5. The displacement matches the theoretical prediction almost perfectly when gravity of $g = 980cm/s^2$ is used. Based on the velocity history plot, we perform a linear fit which yields an acceleration of this free fall to be $972.60cm/s^2$ before it hits the bottom. The no-slip side walls and the sphere rigidity do not seem to have major effects on the solution when comparing to a rigid sphere falling in an open-spaced channel, as also observed in the snapshots in Fig. 4. We acknowledge that the solution of the problem is obvious without much complex phenomena. The purpose of this example is merely to demonstrate the capability of the mIFEM algorithm which can handle large displacement without any re-meshing. This problem had been solved previously in [36] using the original IFEM that has working fluid as water and has a density that is in the same order of magnitude as the solid. It was also solvedusing the semi-implicit algorithm in [39], which produced the acceleration of $968.47cm/s^2$.

4.2 2-D leaflet in a channel

A 2-D deformable leaflet placed in a rectangular fluid channel is studied. The fluid domain is a rectangular channel of size $8cm \times 2cm$. A leaflet of $1.0cm$ in height and $0.5cm$ in width is positioned at $3cm$ from the inflow boundary and it is fixed at the bottom of the channel. The problem setup is illustrated in Figure 6. The results of this particular numerical example was reported in our previous study [39] using the IFEM. Here, several cases are examined using the same geometric setup so that more detailed analysis can be performed.

The fluid is incompressible air with a density of $1.3 \times 10^{-3} g/cm^3$ and a dynamic viscosity of $1.8 \times 10^{-4} g/(cm \cdot s)$. The entire system is driven by a constant inflow velocity, U_0, at the left boundary of the channel. The outlet on the right has stress-free boundary condition, and both top and bottom boundaries have no-slip boundary condition. The leaflet is a linear elastic material with Young's modulus of $E = 1000 dyn/cm^2$ or $100 Pa$ and Poisson's ratio of 0.3. Three sets of parameters with different combinations of solid densities ρ^s and inflow velocity U_0 are studied. These parameters then correspond to different density ratios and Reynolds numbers using the leaflet length as the characteristic length. They are listed in Table 1.

For the first two cases with Re=144.5 and density ratio of 77.92 and 770.2, the overall deformation of the leaflet when reaching the steady state ($\sim 1.4s$) is quite small due to the low Reynolds number flow, see Figures 7(a) and 7(b). The deformed shape from mIFEM yields a slightly smaller deformation comparing to the IFEM. This is coming from the fact that the IFEM overestimates the solid deformation because the solid is set to be the fluid velocity which is usually larger and the dynamics of the solid itself is not being solved.

A quantitative comparison of the two cases can be seen by measuring the x-coordinate at the top left corner of the leaflet, see Figure 8. Both density ratios converge to almost the same solution when reaching steady state, which is expected when using the same Reynolds number where the inertial force takes almost no effect at steady state. It is evident that with a relatively low density (or lighter solid), the leaflet oscillates back and force a few times before finally reaching steady state, while high density leaflet steadily deforms to the steady state. This behavior is also expected as a light solid can be easily swayed by the surrounding fluid.

In the case of a higher Reynolds number of 722.3 (case 3), the leaflet experiences a larger deformation, see Figure 9(a). While the mIFEM captures the deformed leaflet at the steady state, the IFEM algorithm failed to reach the steady state solution due to a highly distorted mesh of the leaflet. An enlarged mesh of the solid at its left corner is shown in Figure 9(b). The velocities near the solid boundary nodes are inaccurately estimated based on the fluid and severe mesh distortion occurs at this point, which eventually results in a total failure of the simulation. This problem cannot be improved even with finer mesh. Figure 10 shows that each increase of the mesh resolution (regular tetrahedral mesh of 6400, 10000, and 256000 elements, respectively) allows the simulation to last slightly longer time, but ultimate cannot reach a steady state. However, using the mIFEM, all mesh resolutions converge to the same solution, as shown in Figure 11. The fluid field solutions (pressure contour and velocity vector) at different time steps are also shown in Figure 12.

This example shows that the mIFEM can more realistically and accurately capture the solid deformation when the IFEM cannot, especially in cases of high Re number flows where the solid dynamics cannot be computed accurately based on the fluid solution.

4.3 2-D oscillating deformable plate in a flow

In this example, we show a solid initiated fluid-structure interaction problem where a de - formable plate is attached to the bottom of a fluid and is prescribed with an oscillatory motion. The fluid domain is a 2-D rectangular channel with $8cm$ in length and $2cm$ in width. The problem setup is shown in Fig. 13. The deformable plate is $0.5cm \times 1cm$, placed in the middle of the channel length. The bottom surfaces of the plate and the fluid domain are overlapped. The bottom surface of the plate is prescribed with an oscillating motion in the horizontal direction using a sinusoidal function $v_x^s = -sin(2\pi t)$ which has a frequency of 1Hz.

The fluid is incompressible air at room temperature with $\rho^f = 1.3\times10^{-3} g/cm^3$ and dynamic viscosity of $\mu = 1.8\times10^{-4} g/(cm\cdot s)$. The plate is a viscoelastic material with Young's modulus of $E = 10Pa$, Poisson's ratio $\nu = 0.3$ and damping factor $\eta = 100$ *poise*. The density of the solid is $\rho^s = 1.001 g/cm^3$. No-slip and no-penetration boundary conditions are assigned to all the fluid boundaries so that the fluid is contained in the box. Periodic boundary condition is used on the bottom surface of the fluid domain. The fluid and the plate are discretized with 6400 and 1250 uniform quadrilateral elements, respectively. The time step size used in the simulation is $1 \times 10^{-2} s$.

The fluid velocity field at four instances during one oscillation cycle is shown in Fig. 14. Because it is incompressible fluid, the same amount of fluid pushed by the moving plate goes back to the other side of the plate and causes a reverse flow. This explains the vortex generated near the top surface of the plate. To clearly show the instantaneous snapshots of the deformation and the location of the plate, we superimpose the plate boundaries at these four particular time steps and its initial undeformed state in Fig. 15.

The plate movement and deformation are dominated by the oscillatory motion prescribed on the bottom surface. The oscillatory wave travels through the plate in the vertical direction causing a slight phase lag between the plate velocity on the bottom and top surfaces. The velocities of the left bottom corner and the left top corner are shown in Fig. 16(a). As shown from the figure, the vibration magnitude is slightly larger on the top surface. The vibrational frequency of the top surface of the plate remains the same as the prescribed oscillatory frequency at $1Hz$. The same frequency and wave form are also found in the volume flow rate computed across the vertical line at $x = 4cm$ of the fluid domain, shown in Fig. 16(b). However, because the plate is much denser than air and no driving force is applied in the fluid domain, the air flow can hardly affect the plate deformation, even though the displacement of the plate is much larger.

4.4 3-D flow past a deformable cylinder

This numerical example is a 3-D fluid flow over a deformable cylinder. The fluid domain has dimensions $8 \times 2 \times 2\ cm^3$ in x, y, z directions, respectively. The fluid is considered as incompressible air at room temperature with a density $\rho^f = 1.3 \times 10^{-3} g/cm^3$ and dynamic viscosity $\mu = 1.8 \times 10^{-4} g/(cm \cdot s)$. The cylinder has a diameter of $d = 0.5cm$ and length of $l = 2cm$ in the z-direction. The center of the cylinder is located at $(1.5cm, 1.0cm)$ in the $x - y$ plane. The two ends of the cylinder are fixed on the front and the back of the fluid channel walls. The problem set up is shown in Fig. 17. The cylinder is a viscoelastic material with density $\rho^s = 1.001 g/cm^3$, Young's modulus $E = 500Pa$, Poisson ratio $\nu = 0.3$ and a damping factor $\eta = 100 poise$.

A constant inflow boundary condition $U_0 = 100 cm/s$ is applied at the channel inlet ($x = 0cm$). The channel exit ($x = 8cm$) is given as the outflow stress-free boundary condition. No-penetration boundary condition is applied at the top and the bottom boundaries of the fluid

channel ($y = 0$ and $2cm$). The front and back surfaces ($z = 0$ and $2cm$) are no-slip and no-penetration steady walls. The Reynolds number is $Re = \rho^f U_0 d/\mu = 361$. For a cylinder that is rigid and stationary, a Strouhal number of $St = fd/U_0 = 0.2$ is expected [49], where f is the vortex shedding frequency.

The fluid domain is discretized using 77, 818 non-uniform hexahedral elements with higher mesh density concentrated around the cylinder surface. The cylinder is discretized with 32, 053 uniform tetrahedral elements. The time step size used in the simulation is $\Delta t = 1 \times 10^{-4} s$.

A few time snapshots of the velocity field in the mid-z x-y plane are shown in Fig. 18. Vortex shedding is clearly observed. To further obtain the quantitative shedding frequency, we examine the lift and drag coefficients from this simulation. The lift coefficient $C_L = L/\left(\frac{1}{2}\rho^f U^2 A\right)$ and drag coefficient $C_D = D/\left(\frac{1}{2}\rho^f U^2 A\right)$ are defined based on the lift and drag forces L and D where A is the flow frontal projected area of the cylinder $1 \times d$. These forces are obtained by integrating the pressure force, P, in the x and y directions such that:

$$L = \int_{\Gamma^s} P n_y d\Gamma, \quad (34)$$

$$D = \int_{\Gamma^s} P n_x d\Gamma. \quad (35)$$

The time history of drag and lift coefficients are plotted in Fig. 19. The time averaged drag coefficient is found to be $< C_D >_t = 1.5$. The time averaged lift coefficient is $< C_L >_t = -3.5 \times 10^{-4}$, which is very close to zero. The oscillatory frequency of C_L and C_D through FFT power spectrum are found to be $f^{CL} = 41.02 Hz$ and $f^{CD} = 82.03 Hz$. The frequency of the drag coefficient is approximately twice as the frequency of the lift coefficient, which is consistent with results found in [49]. Due to the alternating vortex wake the oscillations in lift force occur at the vortex shedding frequency. Based on the oscillation frequency of the lift coefficient, the Strouhal number from our deformable cylinder case is $St = 0.205$. All the values, such as the drag coefficient, lift coefficient and Strouhal number come quite close to the expected values of a flow past a stationary, smooth, infinitely long cylinder case [50]. The lift coefficient fluctuates about a zero mean value and the instantaneous drag coefficient fluctuates about the mean value of the drag coefficient at 1.15, although at a much smaller oscillatory amplitude.

The velocity of a particular point on the cylinder surface at (1.5, 1.25, 1) is examined, shown in Fig. 20. The velocity in the y-direction has the same frequency as the drag coefficient. The wave form of the velocity in the x-direction shows 2 modes of frequencies, the higher frequency mode matches the one of the lift coefficient f^{CL}, which is the same as the vortex shedding frequency. The lower frequency mode is found to be $11.72 Hz$. We believe that this lower frequency mode is related to the natural frequency of the cylinder, which is dependent on the material properties. This cylinder vibration is caused by the fluid generating a pressure difference in the x-direction across the cylinder. This pressure difference has a larger amplitude and smaller frequency than the one that induces drag. It pushes the cylinder in the flow direction and is balanced with the force acting on the two ends of the cylinder. When the vortex shedding frequency and the natural frequency of the cylinder are close to each other, the phenomena of lock-in would happen which can lead to large vibration or even structure damage.

5 Conclusion

In this study, we presented a new algorithm, the mIFEM, that reverses the coupling process of the original IFEM. The method is designed so that the dynamics of the solid is been solved rather than been imposed based on the dynamics of the background fluid. It is particularly important when the solid dynamics plays a major role in the fluid-structure interaction process. Comparing to the IFEM algorithm, the solid behaviors are much more realistically captured, especially in the case with high Reynolds numbers. Instead of having the entire solid (interior and boundary) to follow the overlapping fluid, the solid only "sees" the fluid at the boundary via boundary conditions.

The idea of using indicator function to decompose the fluid domain into real and artificial fluids has several contributions in both physical and mathematical aspects. The real and artificial fluid are identified by the indicator function so the real and artificial fluids can be treated separately. It is important not only because the artificial fluid can be assigned properties close to the solid so that the artificial fluid can follow the solid behavior more easily, but also the artificial fluid can be compressible while the real fluid can be either compressible or incompressible. Accounting for the compressibility of the artificial fluid is important because the volume change of the solid can now be considered. Therefore, the mIFEM is not restricted to only solve fluid-structure interaction problems when the solid is nearly-incompressible. Moreover, because the volume correction algorithm [46] is no longer required to conserve the solid volume, the solid behaviors are more accurate and realistic.

Our 2-D and 3-D testing cases show that the mIFEM captures the dynamics of the solid more precisely comparing to the original IFEM. Even though these examples involve relatively small deformations, it is not a limitation in the algorithm. We choose these specific cases so that the solutions can be compared with IFEM and other techniques more easily. Although the second example with the oscillating leaflet is also small deformation, its displacement is not. We show, based on the solutions of the examples, that the mIFEM is a robust algorithm and is appropriate to be applied in a wide range of fluid-structure interaction problems.

Acknowledgments

This work is partially funded by DTRA (HDTRA1-09-1-0029) and NIH (R01-DC005642-07). We utilized CCNI Bluegene/Q at Rensselaer Polytechnic Institute for the computations performed in this study.

References

[1]. Hughes TJR, Liu WK, Zimmermann TK. Lagrangian-Eulerian Finite Element formulation for incompressible viscous flows. Computer Methods in Applied Mechanics and Engineering. 1981; 29:329–349.

[2]. Liu WK, Ma DC. Computer implementation aspects for fluid-structure interaction problems. Computer Methods in Applied Mechanics and Engineering. 1982; 31:129–148.

[3]. Huerta A, Liu WK. Viscous flow with large free surface motion. Computer Methods in Applied Mechanics and Engineering. 1988; 69:277–324.

[4]. Liu WK, Chang H, Chen J, Belytschko T. Arbitrary Lagrangian-Eulerian Petrov-Galerkin finite elements for nonlinear continua. Computer Methods in Applied Mechanics and Engineering. 1988; 68:259–310.

[5]. Hu HH, Patankar NA, Zhu MY. Direct numerical simulations of fluid-solid systems using the arbitrary Lagrangian-Eulerian technique. Journal of Computational Physics. 2001; 169:427–462.

[6]. Zhang LT, Wagner G, Liu WK. Modeling and simulation of fluid structure interaction by meshfree and FEM. Communications in Numerical Methods in Engineering. 2003; 19:615–621.

[7]. Farhat C, Lesoinne M, Le Tallec P. Load and motion transfer algorithms for fluid/structure interaction problems with non-matching discrete interfaces: momentum and energy conservation, optimal discretization and application to aeroelasticity. Computer Methods in Applied Mechanics and Engineering. 1998; 157:95–114.

[8]. Bazilevs Y, Hsu M-C, Scott MA. Isogeometric fluid-structure interaction analysis with emphasis on non-matching discretizations, and with application to wind turbines. Computer Methods in Applied Mechanics and Engineering. 2012; 249-252:28–41.

[9]. Tezduyar TE, Takizawa K, Moorman C, Wright S, Christopher J. Space-time finite element computation of complex fluid-structure interactions. International Journal for Numerical Methods in Fluids. 2010; 64:1201–1218.

[10]. Takizawa K, Tezduyar TE. Multiscale space-time fluid-structure interaction techniques. Computational Mechanics. 2011; 48:247–267.

[11]. Takizawa K, Henicke B, Puntel A, Spielman T, Tezduyar TE. Space-time computational techniques for the aerodynamics of flapping wings. Journal of Applied Mechanics. 2012; 79:010903.

[12]. Mittal R, Dong H, Bozkurttas M, Najjar FM, Vargas A, von Loebbecke A. A versatile sharp interface immersed boundary method for incompressible flows with complex boundaries. Journal of Computational Physics. 2008; 227(10):4825–4852. [PubMed: 20216919]

[13]. Wang, Xingshi; Zhang, Lucy T. Numerical method for fluid-structure interactions with sharp interfaces: formulation and convergence tests. Computational Mechanics. 2010; 45(4):321–334.

[14]. Peskin CS. Numerical analysis of blood flow in the heart. Journal of Computational Physics. 1977; 25:220–252.

[15]. McCracken MF, Peskin CS. A vortex method for blood flow through heart valves. Journal of Computational Physics. 1980; 35:183–205.

[16]. McQueen DM, Peskin CS. Computer-assisted design of pivoting-disc prosthetic mitral valves. Journal of Computational Physics. 1983; 86:126–135.

[17]. Peskin CS, McQueen DM. A three-dimensional computational method for blood flow in the heart. I. Immersed elastic fibers in a viscous incompressible fluid. Journal of Computational Physics. 1989; 81(2):372–405.

[18]. Peskin CS, McQueen DM. Cardiac fluid dynamics. Critical Reviews in Biomedical Engineering. SIAM Journal on Scientific and Statistical Computing. 1992; 20(6):451–459.

[19]. Peskin CS, McQueen DM. Mechanical equilibrium determines the fractal fiber architecture of aortic heart valve leaflets. American Journal of Physiology. 1994; 266(1):H319–H328. [PubMed: 8304514]

[20]. Peskin, CS.; McQueen, DM. Case Studies in Mathematical Modeling-Ecology, Physiology, and Cell Biology. Prentice-Hall; 1996.

[21]. LeVeque RJ, Li Z. The immersed interface method for elliptic equations with discontinuous coefficients and singular sources. SIAM Journal on Numerical Analysis. 1994; 31(4):1091–1044.

[22]. LeVeque RJ, Li Z. Immersed interface methods for stokes flow with elastic boundries or surface tension. SIAM Journal on Scientific Computing. 1997; 18(3):709–735.

[23]. Fogelson AL, Keener JP. Immersed interface method for Neumann and related problems in two and three dimensions. SIAM Journal on Scientific Computing. 2000; 22(5):1630–1654.

[24]. Lee L, LeVeque RJ. An immersed interface method for incompressible Navier-Stokes equations. SIAM Journal on Scientific Computing. 2003; 25(3):832–856.

[25]. Li Z, Lai MC. The immersed interface mehtods for the Navier-Stokes equations with singular forces. Journal of Computational Physics. 2001; 171(822-842)

[26]. Wiegmann A, Bube KP. The immersed interface method for nonlinear differential equations with discontinuous coefficients and singular sources. SIAM Journal on Numerical Analysis. 1998; 35(1):177–200.

[27]. Wiegmann A, Bube KP. The explicit-jump immersed interface method: finite difference methods for PDEs with piecewise smooth solutions. SIAM Journal on Numerical Analysis. 2000; 37(3): 827–862.

[28]. Wang X, Liu WK. Extended immersed boundary method using FEM and RKPM. Computer Methods in Applied Mechanics and Engineering. 2004; 193:1305–1321.

[29]. Boffi D, Gastaldi L. A finite element approach for the immersed boundary method. Computers and Structures. 2003; 81:491–501.

[30]. Boffi D, Gastaldi L, Heltai L. On the CFL condition for the finite element immersed boundary method. Computers and Structures. 2007; 85:775–783.

[31]. Zhang LT, Gerstenberger A, Wang X, Liu WK. Immersed finite element method. Computer Methods in Applied Mechanics and Engineering. 2004; 193:2051–2067.

[32]. Liu WK, Liu Y, Zhang LT, Wang X, Gerstenberger A, Farrell D, International Center for Numerical Methods and Engineering. Immersed finite element method and applications to biological systems. Finite Element Methods: 1970′s and Beyond. 2004

[33]. Liu Y, Liu WK. Rheology of red blood cell aggregation in capillary by computer simulation. Journal of Computational Physics. 2006; 220:139–154.

[34]. Gay M, Zhang LT, Liu WK. Stent modeling using immersed finite element method. Computer Methods in Applied Mechanics and Engineering. 2006; 195:4358–4370.

[35]. Liu WK, Liu Y, Farrell D, Zhang LT, Wang S, Fukui Y, Patankar N, Zhang Y, Bajaj C, Lee J, Hong J, Chen X, Hsu H. Immersed finite element method and its applications to biological systems. Computer Methods in Applied Mechanics and Engineering. 2006; 195:1722–1749. [PubMed: 20200602]

[36]. Zhang LT, Gay M. Immersed finite element method for fluid-structuure interactions. Journal of Fluids and Structures. 2007; 23:839–857.

[37]. Zhang, Lucy T.; Gay, Mickael. Imposing rigidity constraints on immersed objects in unsteady fluid flows. Computational Mechanicsv. 2008; 42:357–370.

[38]. Wang, Sheldon X.; Zhang, Lucy T.; Liu, Wing K. Finite element formulations for immersed methods: explicit and implicit approaches. Journal of Computational Physics. 2009; 228(7): 2535–2551.

[39]. Wang, Xingshi; Wang, Chu; Zhang, Lucy. Semi-implicit Formulation of the Immersed Finite Element Method. Computational Mechanics. 2012; 49:421–430.

[40]. Unverdi SO, Tryggvason G. A Front-Tracking Method for Viscous, Incompressible Multifluid flows. Journal of Computational Physics. 1992; 100:25–37.

[41]. Torres DJ, Brackbill JU. The Point-Set Method: Front-Tracking without Connectivity. Journal of Computational Physics. 2000; 165:620–644.

[42]. Gottlieb, Sigal; Chi-Wang, Shu. Total variation diminishing runge-kutta schemes. Mathematics of Computation. 1998; 67(221):73–85.

[43]. Terashima H, Tryggvason G. A front-tracking/ghost-fluid method for fluid interfaces in compressible flows. Journal of Computational Physics. 2009; 228:4012–4037.

[44]. Belytschko, T.; Liu, WK.; Moran, B. Nonlinear Finite Elements for Continua and Structures. John Wiley and Sons, Inc.; New York: 2000.

[45]. Hughes, TJ. The Finite Element Method: Linear Static and Dynamic Finite Element Analysis. Dover Publications, Inc.; 2000.

[46]. Zhao H, Freund BJ, Moser DB. A fixed-mesh method for incompressible flow-structure systems with finite solid deformations. Journal of Computational Physics. 2008; 227:3114–3140.

[47]. Wang X, Zhang LT. Interpolation functions in the immersed boundary and finite element methods. Computational Mechanics. 45(4):321–334. 20010.

[48]. Knoll DA, Keyes DE. Jacobian-free newton-krylov methods: a survey of approaches and applications. Journal of Computational Physics. 2004; 193(2):357–397.

[49]. Blevins, RD. Flow-Induced Vibration. Van Nostrand Reinhold; New York: 1990.

[50]. Anderson, John D., Jr. Fundamentals of Aerodynamics. McGraw-Hill; 2010.

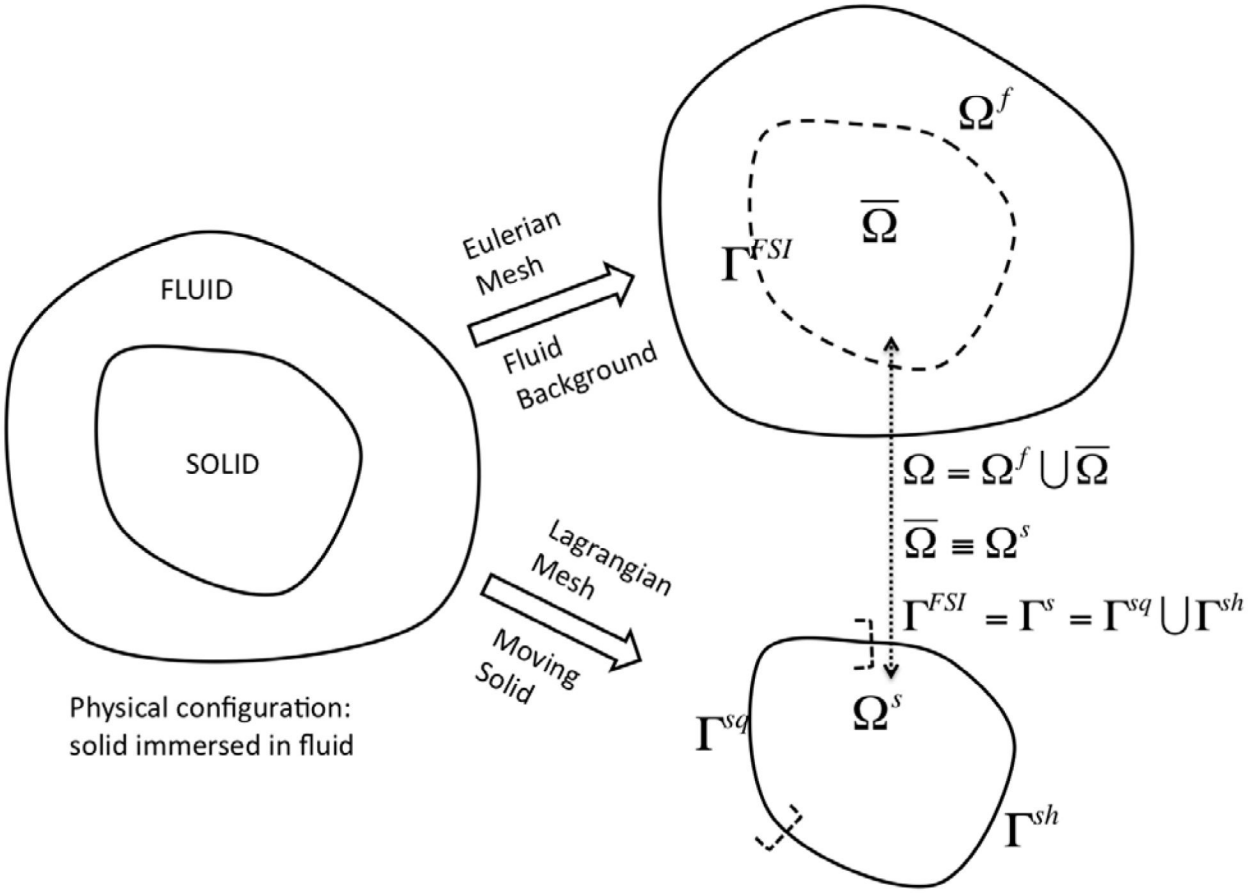
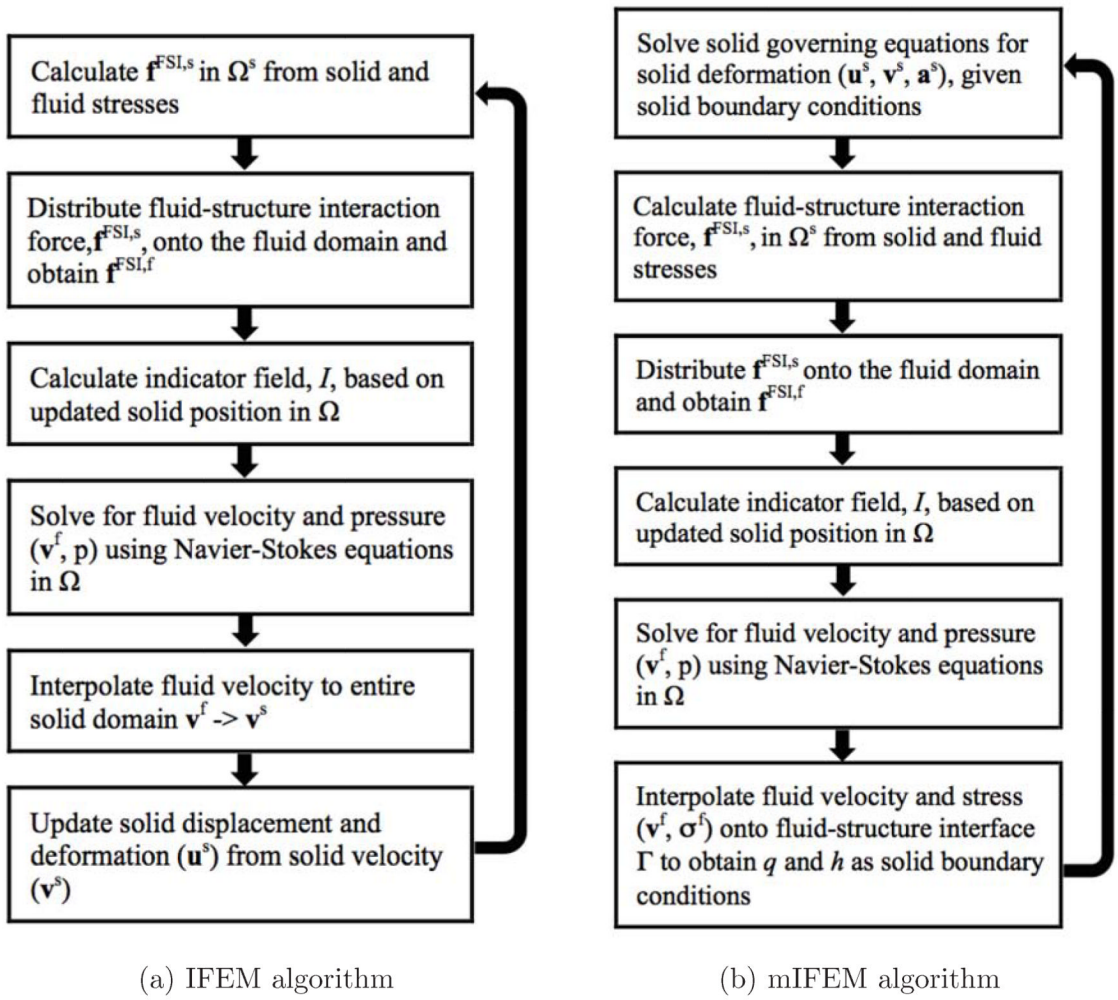


Fig. 1.
Computational domain decomposition.



(a) IFEM algorithm (b) mIFEM algorithm

Fig. 2.
Algorithm flowcharts for IFEM and mIFEM.

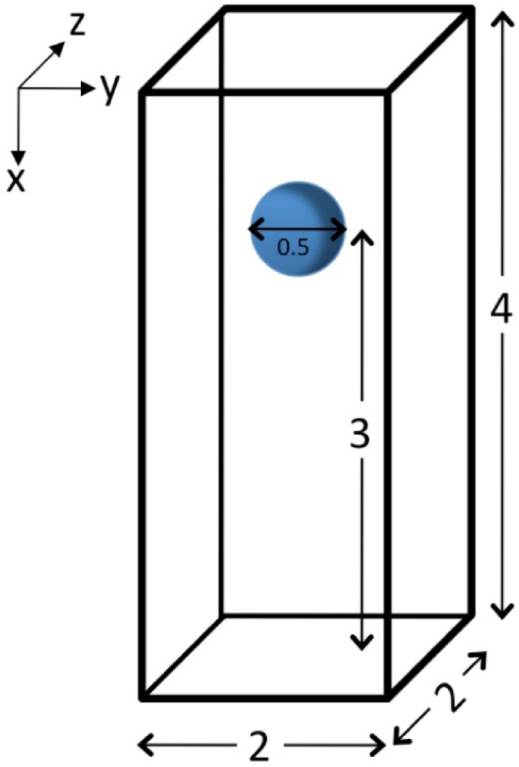


Fig. 3.
Problem statement of a 3-D sphere dropping in a free fall.

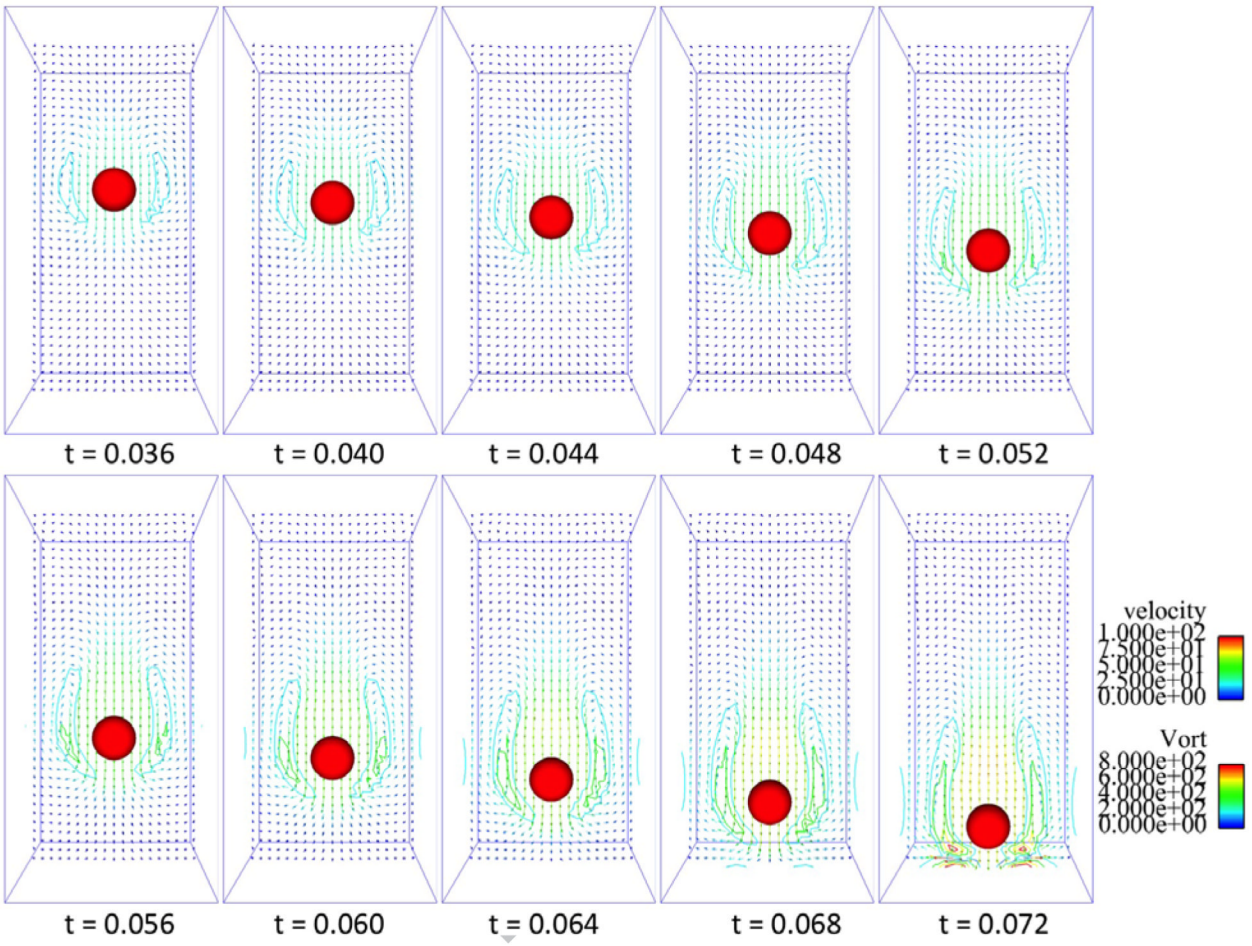
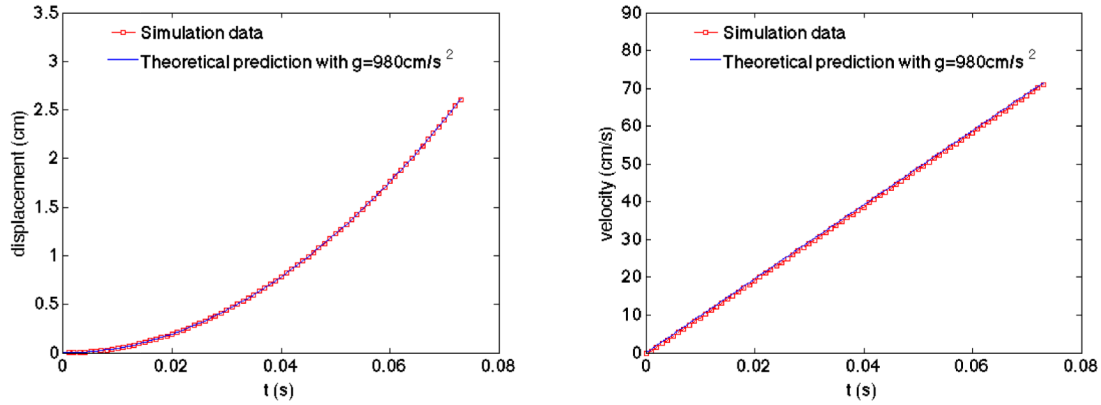


Fig. 4.
3-D sphere dropping in a free fall in a channel at different time steps.



(a) Displacement (b) Velocity

Fig. 5.
Time history of the displacement and velocity at the center of mass as the sphere drops in a free fall.

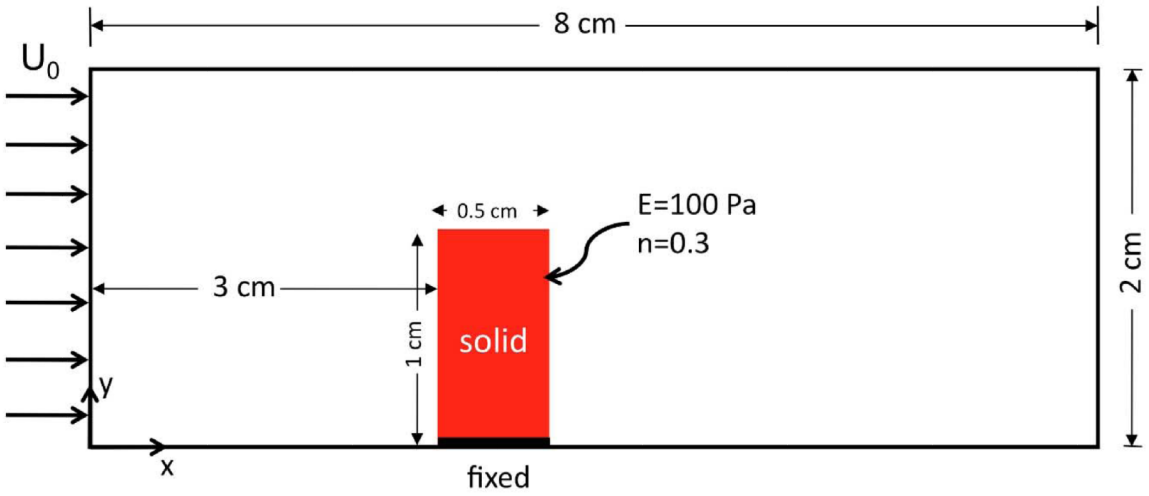
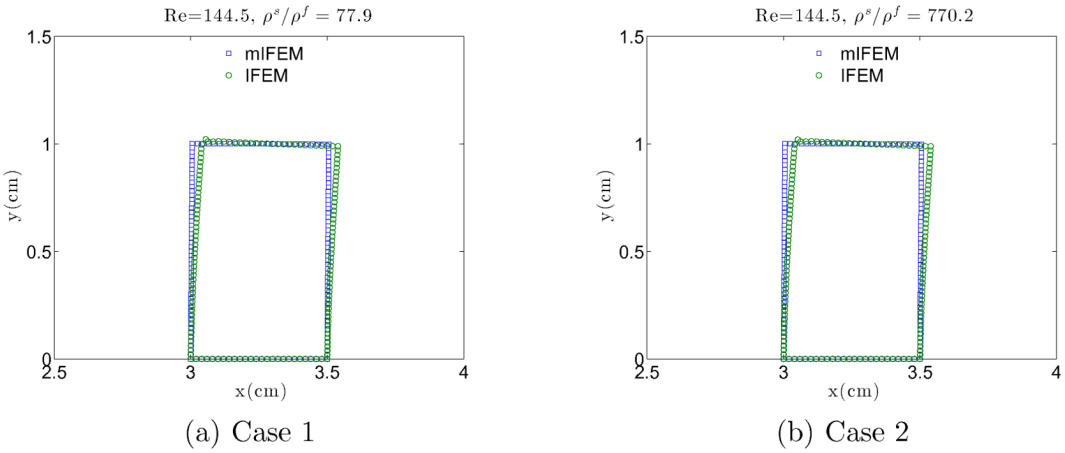


Fig. 6.
Flow over a deformable leaflet in a channel.



(a) Case 1 (b) Case 2

Fig. 7.
Deformed leaflet at steady states.

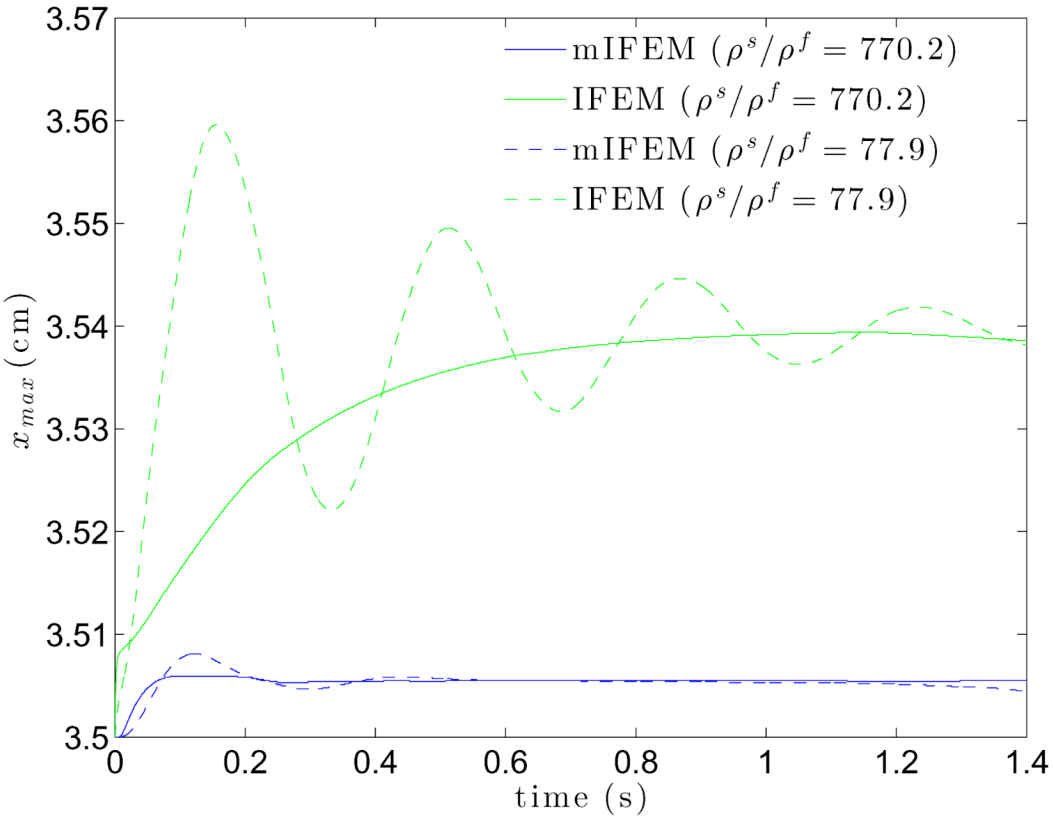
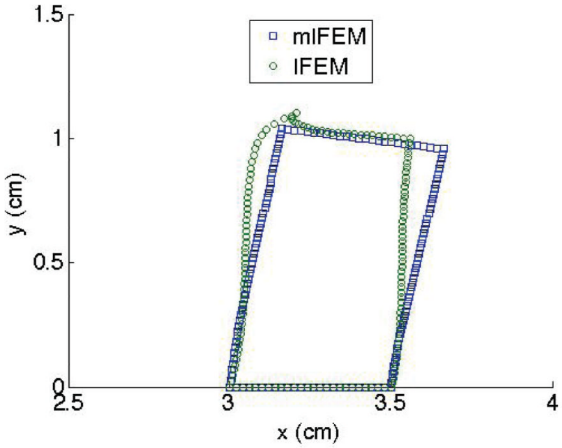
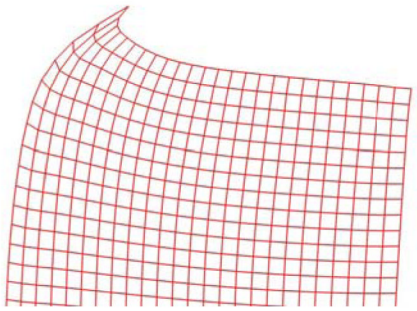


Fig. 8.
Time history of the x-position of the top left corner of the deformed leaflet in a channel at Re=144.5.



(a) Deformed leaflet



(b) Mesh distortion of the left corner of the leaflet when using IFEM.

Fig. 9.
Deformed leaflet for Case 3: Re=722.3, $\rho^s/\rho^f = 77.92$

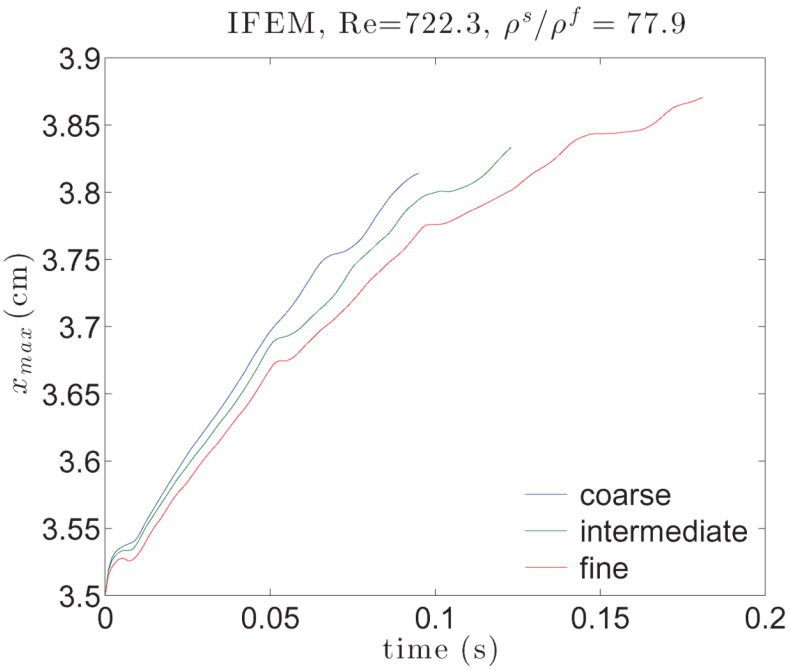


Fig. 10.
Time history of the x-position at the top left corner of the leaflet for different mesh resolution.

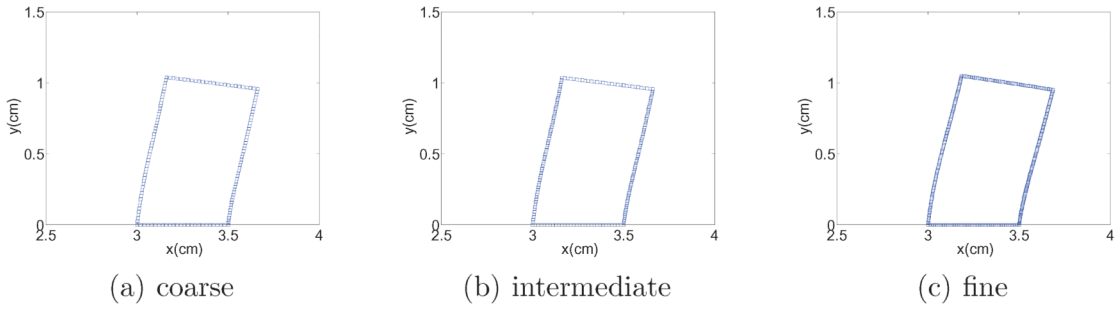


Fig. 11.
Deformed leaflet at steady state using different mesh resolutions.

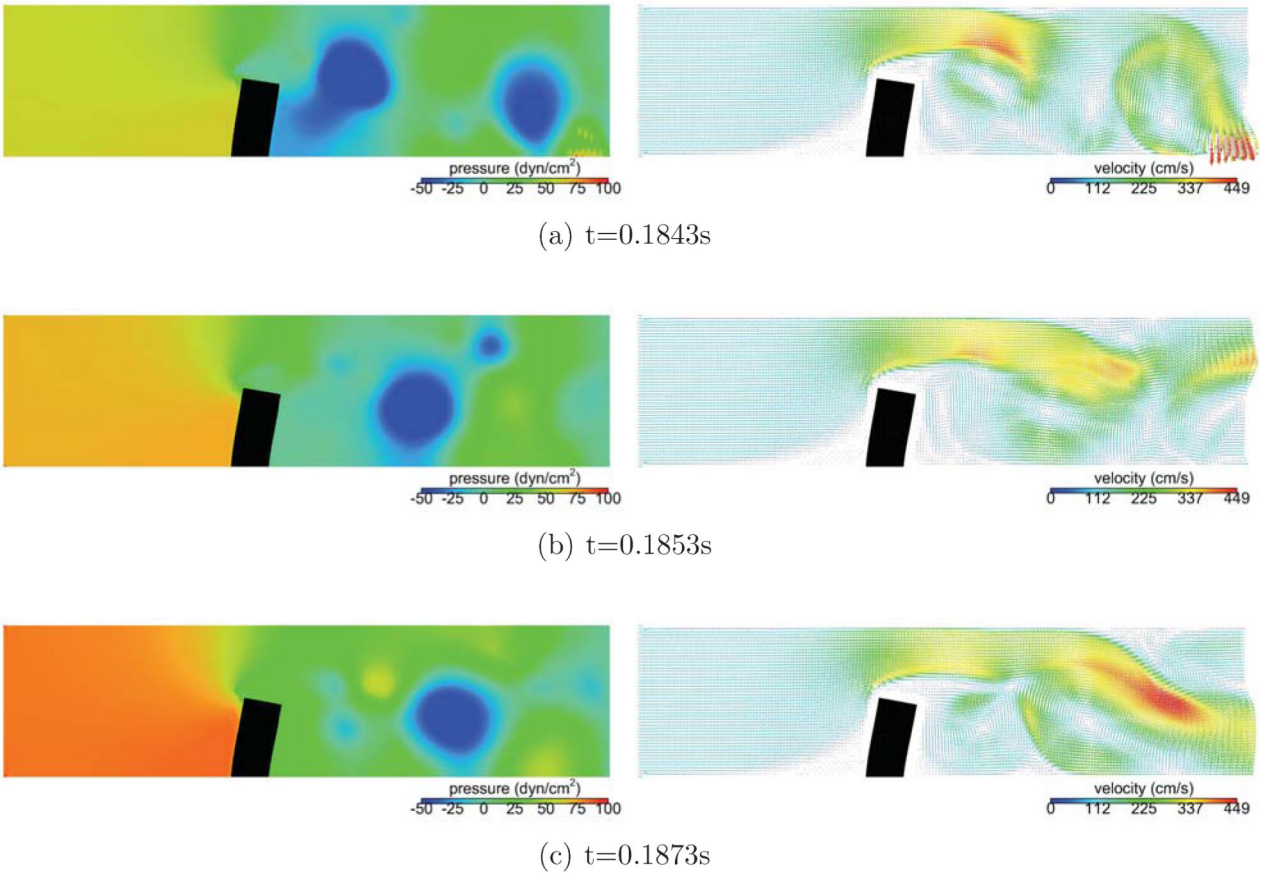


Fig. 12.
Pressure and velocity fields at different time steps towards steady state.

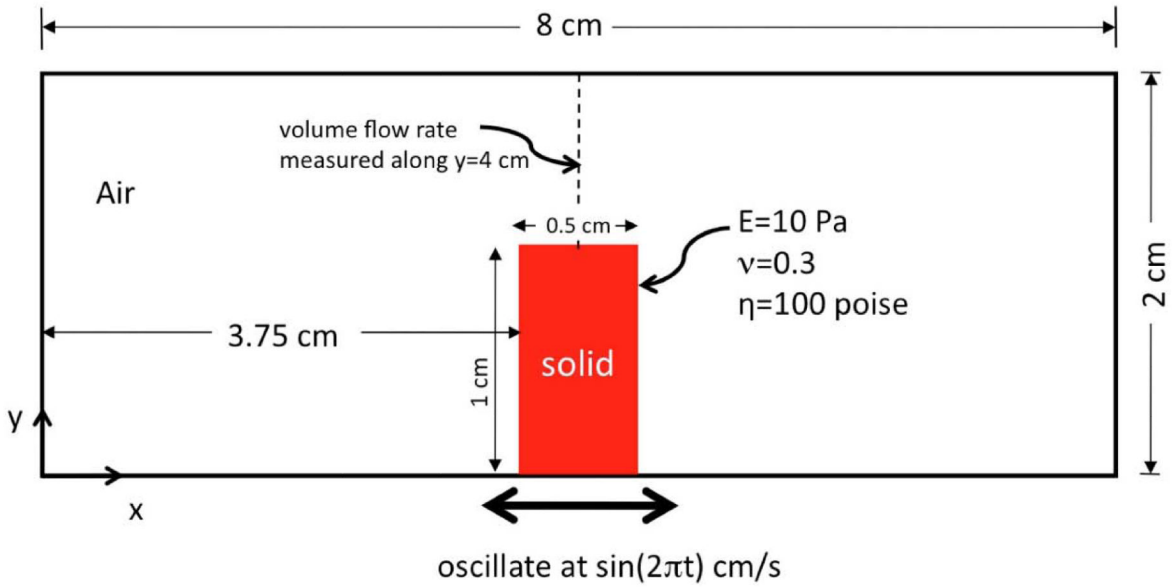


Fig. 13.
2-D deformable plate with prescribed oscillatory motion in a fluid domain.

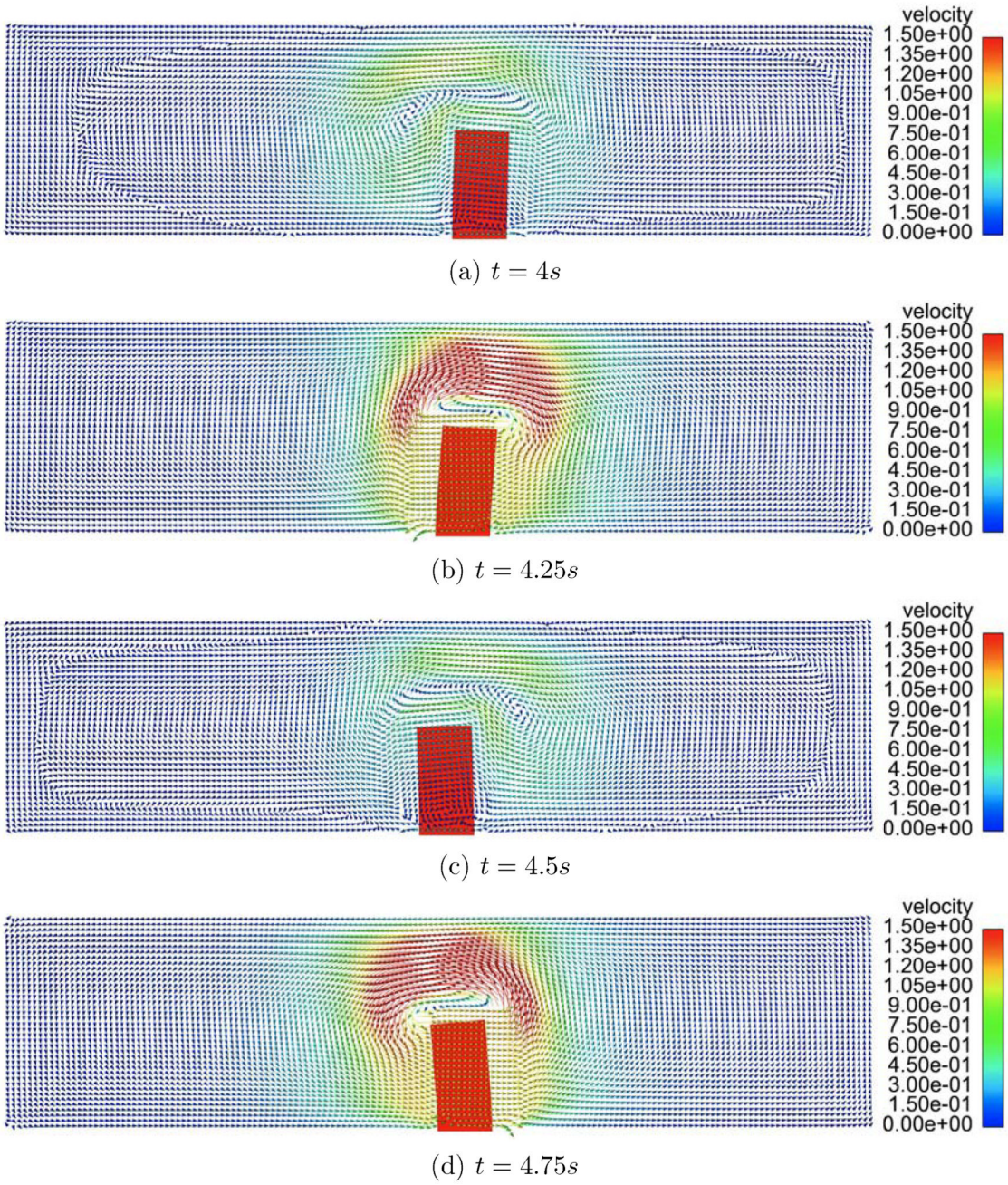


Fig. 14.
Fluid velocity vector field during one cycle.

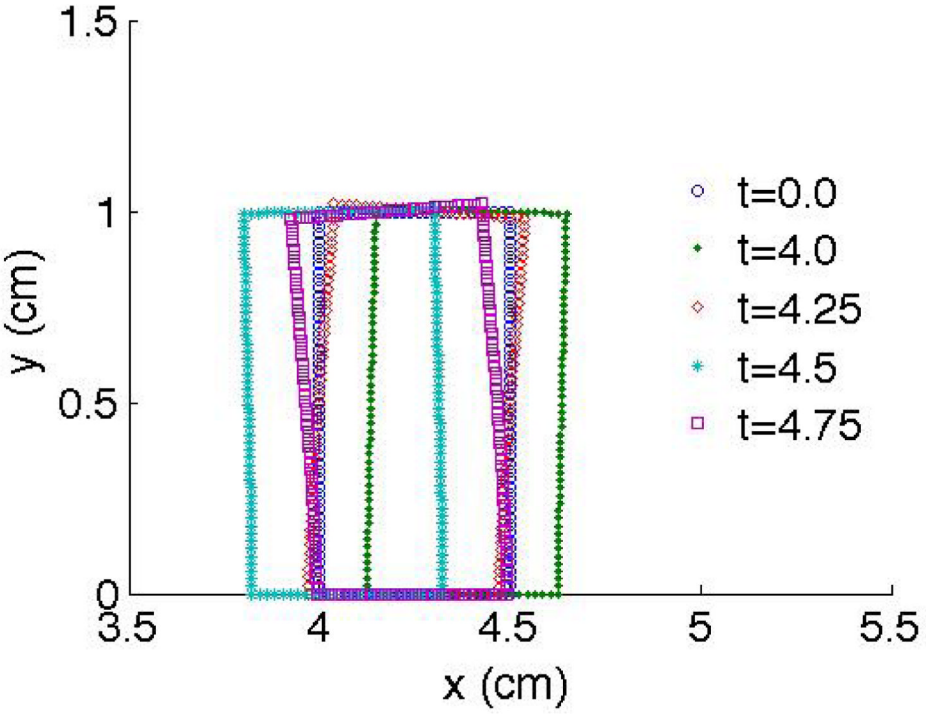


Fig. 15.
Plate boundaries during one oscillation cycle.

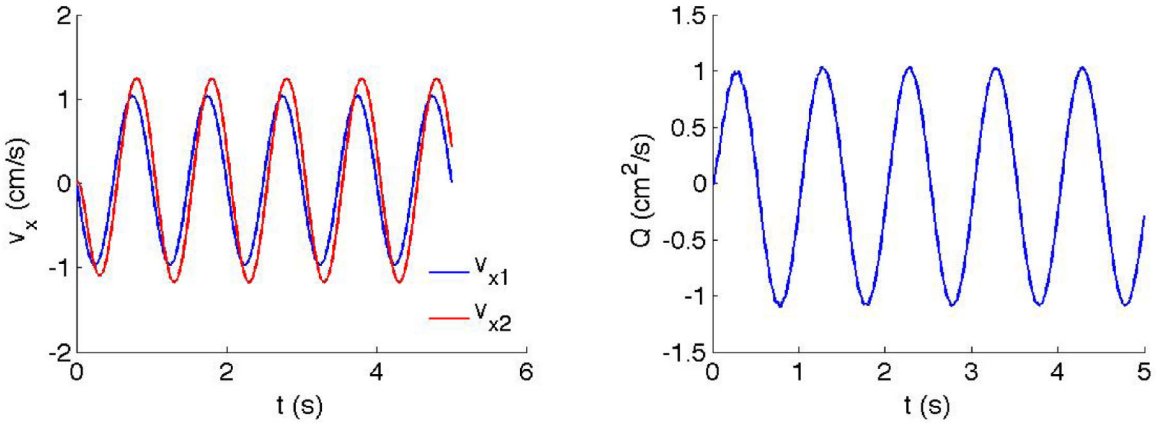


Fig. 16.

The history of plate velocity and fluid volume flow rate.

(a) x-velocity on the left bottom corner (v_{x1}) (b) Volume flow rate measured across $x = 4cm$. and the left top corner (v_{x2}).

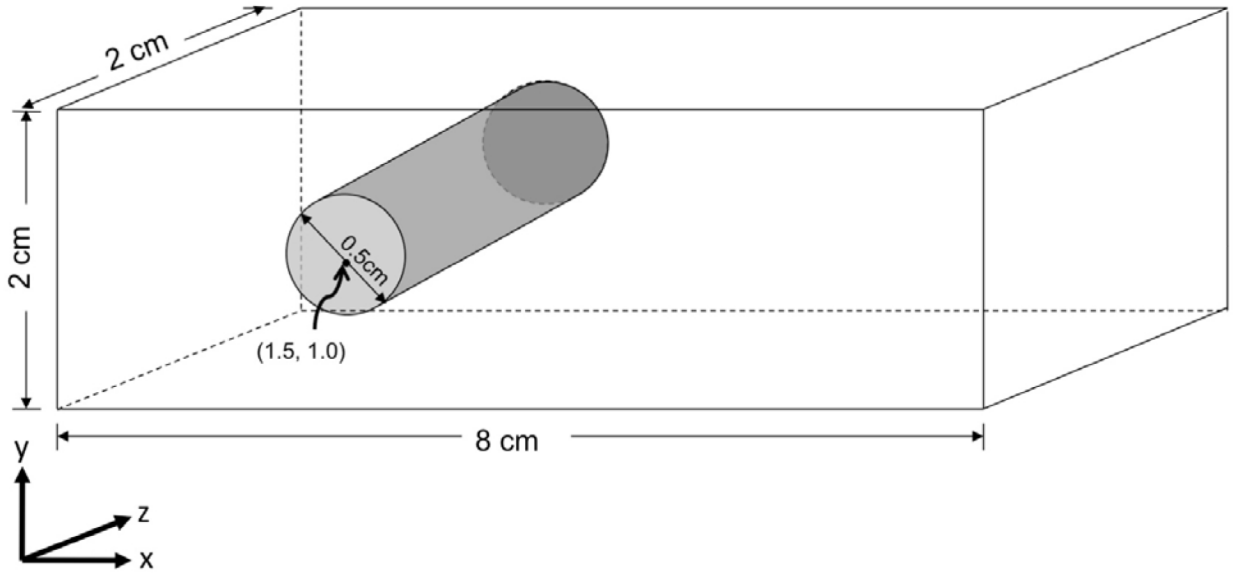
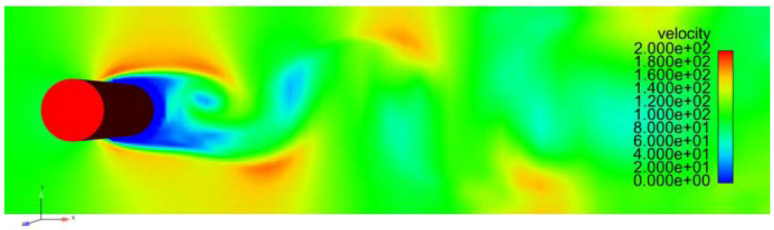
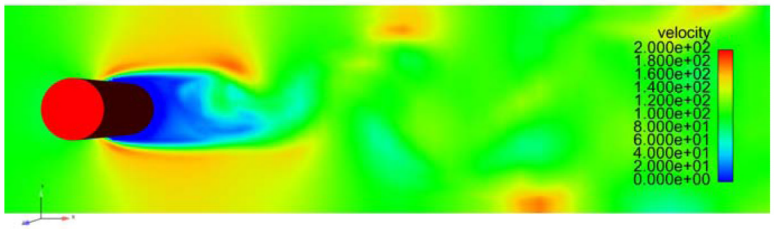


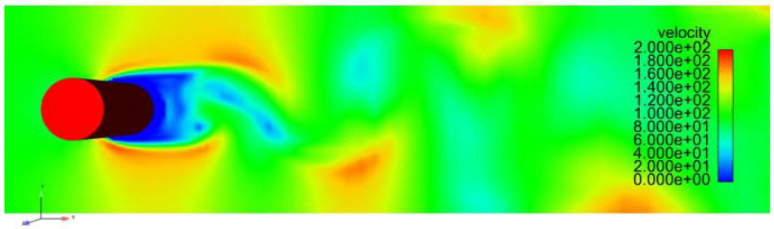
Fig. 17.
Numerical example II: Flow over 3-D deformable cylinder.



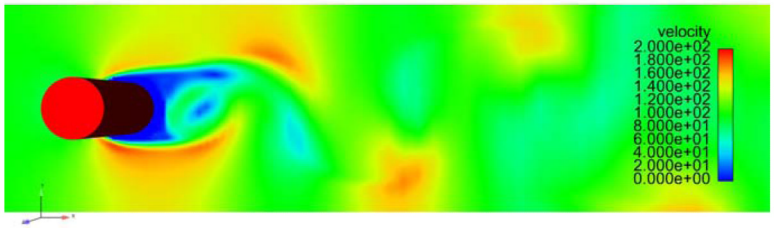
(a) $t = 0.5s$



(b) $t = 0.6s$



(c) $t = 0.7s$



(d) $t = 0.8s$

Fig. 18.
Velocity magnitude contour on the mid-plane ($y = 1cm$) at different time steps.

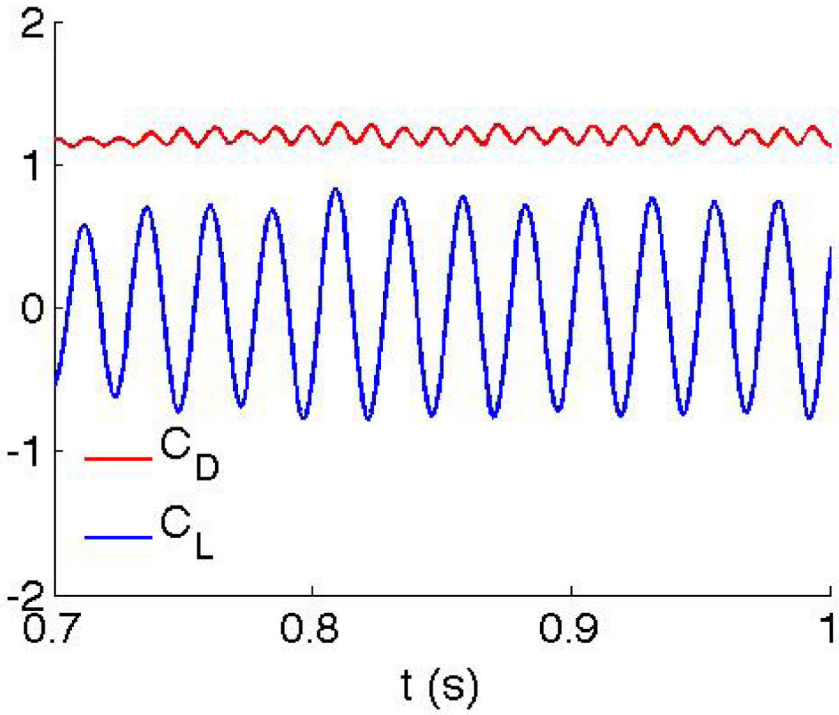


Fig. 19.
Drag coefficient C_D and lift coefficient C_L.

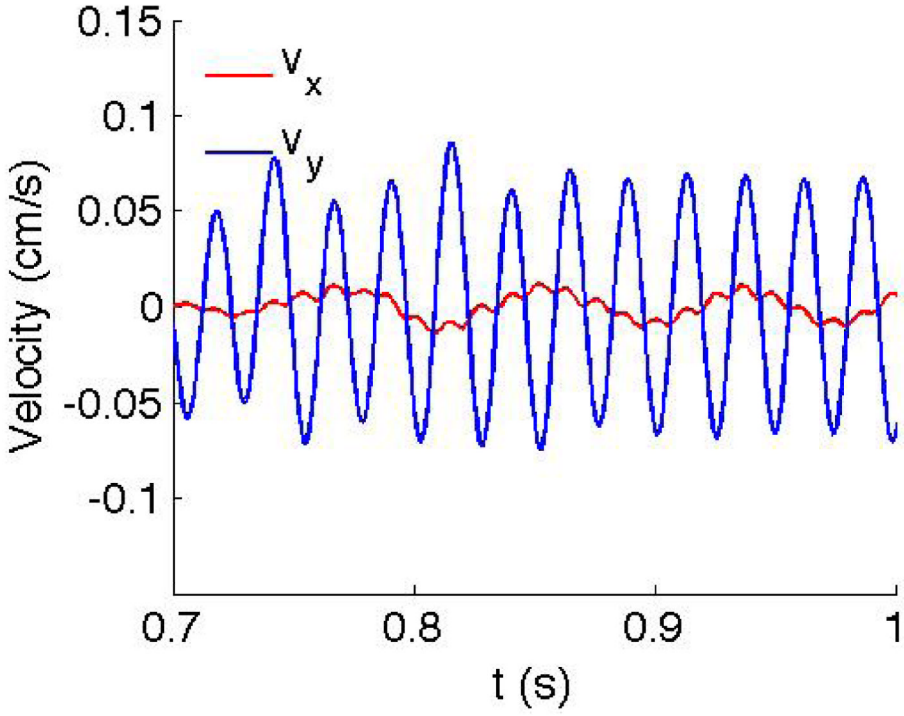


Fig. 20.
Velocities in x- and y- directions on the cylinder surface.

Table 1

Cases with different density ratios (ρ^s/ρ^f) and Re studied for the 2-D leaflet example.

Case	$\rho^s(g/cm^3)$	$U_0(cm/s)$	ρ^s/ρ^f	Re
1	0.1013	20.00	77.92	144.5
2	1.001	20.00	770.2	144.5
3	0.1013	100.0	77.92	722.3