

On Explicit/Implicit and Incompressible/Compressible Issues of Immersed Boundary/Continuum Methods

by

Xiaodong (Sheldon) Wang

Department of Mathematical Sciences

New Jersey Institute of Technology

Newark, NJ 07102

Abstract

In addition to an overview of the immersed boundary/continuum methods and their finite element formulations, explicit *vs.* implicit and incompressible *vs.* compressible issues are discussed. The recent finite element formulations retain the same strategies employed in the original immersed boundary method, namely, the independent Lagrangian solid mesh moves on top of a fixed or prescribed background Eulerian fluid mesh. The added features in recent finite element formulations are the generality of the immersed solid which can occupy a finite volume in the fluid and be impermeable, compressible, and highly deformable. Furthermore, a matrix-free Newton-Krylov iterative solution technique also resolves the time step limitation issues related to stiff spring supports from the boundary and the high elasticity moduli of the immersed solid. This implicit iterative approach enables the application of immersed methods to many engineering problems some of which are documented here for illustrative purposes.

Keywords: Immersed Boundary Method, Kernel, Meshless, Fluid-Structure Interaction, Compressible, Explicit, Implicit, and Mixed Finite Element Formulation.

Submitted to: AMS Contemporary Mathematics Series

1 Fluid-Structure Interactions

The rapid developments of computational mechanics and computer technology during the last two decades have brought immensely powerful computational tools to the disposal of scientists and engineers. With the recent advent of tera-flop and parallel computers, simulation-based engineering processes are gaining more acceptance and rapidly becoming viable design alternatives. Furthermore, the traditional classification of various disciplines of mechanics has been blurred, and interaction problems involving two or more disciplines are often encountered in ever more challenging engineering practices. One of these interactive disciplines is fluid-structure interaction (FSI), which is at the intersection of the disciplines of fluid mechanics, solid mechanics, and dynamics. In fact, the study of FSI systems is becoming a bridge between complex dynamical systems and computational mechanics.

A typical FSI system is shown in Fig. 1. The fluid domain of interest is denoted as Ω^f , its boundary $\partial\Omega^f$ consists of three types of mutually exclusive boundaries, namely, the Dirichlet boundary $\Gamma^{f,v}$, the Neumann boundary $\Gamma^{f,f}$, and the Cauchy boundary for fluid-structure interfaces Γ^{fsi} . Likewise, the solid domain of interest is Ω^s , its boundary $\partial\Omega^s$ consists of three types of mutually

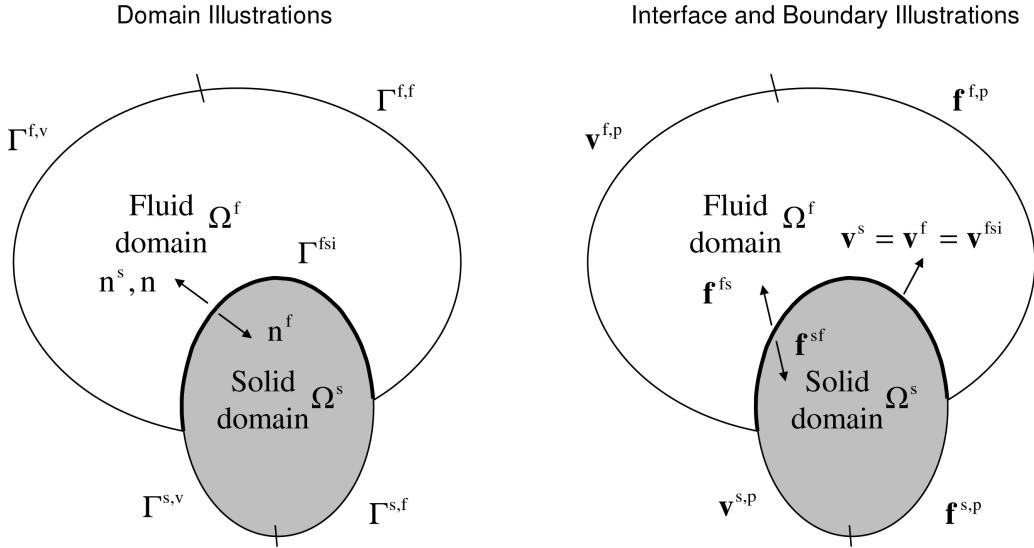


Figure 1: Illustration of a typical fluid-structure interaction system.

exclusive boundaries, namely, the Dirichlet boundary $\Gamma^{s,v}$, the Neumann boundary $\Gamma^{s,f}$, and the Cauchy boundary for fluid-structure interfaces Γ^{fsi} .

Effective numerical algorithms for large FSI systems are essential to instability analysis of linear non-autonomous systems; the computation of limit sets, which include stationary points, periodic orbits, quasi-periodic orbits, and strange attractors; and the computation of turning points, which include super- and sub-critical saddle-node, pitchfork, and Hopf bifurcations in parameter space. Nevertheless, FSI systems, in particular those with highly deformable immersed structures/solids, still pose unique challenges to computational mechanics communities. Reviews of various finite element formulations for FSI systems are available in Morand and Ohayon, 1995 [24] and Bathe, 1998 [17].

2 Immersed Boundary Method

The immersed boundary method was originally developed by Peskin [5] for the computation of blood flows interacting with the heart and heart valves. It has since then been successfully extended to three-dimensional heart-flow interactions and a variety of other biomechanics problems, which include the design of prosthetic cardiac valves [9], swimming motions of marine worms [11], wood pulp fiber dynamics [26], wave propagation in cochlea [25], and platelet aggregation and biofilm processes [1] [8]. A recent review with comprehensive descriptions of the immersed boundary method and its applications has been presented by Peskin [6]. In this paper, we only summarize the key aspects of the immersed boundary method.

Consider a neutrally buoyant immersed flexible structure contained in a viscous incompressible fluid. Let $\mathbf{F}(s, t)$ denote the elastic fiber point force (force per unit length as in the surface tension definition) at the Lagrangian parametric coordinate $\mathbf{x}^s(s, t) = (x_1(s, t), x_2(s, t), x_3(s, t))$ as a function of the arc length s and the time t ,

$$\mathbf{F}(s, t) = \frac{\partial(T(s, t)\mathbf{t}(s, t))}{\partial s}, \quad (2.1)$$

where T and \mathbf{t} represent the tension within the fiber and the unit tangent vector, respectively.

Denote $\mathbf{f}(s, t)$ as the corresponding effective body force in the fluid domain at the spatial position \mathbf{x} , Ω as the entire fluid domain, the governing equations of the fluid-structure interaction system involving a single neutrally buoyant smooth submerged elastic fiber Γ_s can be stated as

$$\rho\left(\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v}\right) = -\nabla p + \mu \nabla^2 \mathbf{v} + \mathbf{f}, \quad (2.2)$$

$$\nabla \cdot \mathbf{v} = 0, \quad (2.3)$$

$$\mathbf{f}(\mathbf{x}, t) = \int_{\Gamma_s} \mathbf{F}(s, t) \delta(\mathbf{x} - \mathbf{x}^s) d\Gamma, \quad (2.4)$$

$$\mathbf{v}^s = \int_{\Omega} \mathbf{v}(\mathbf{x}, t) \delta(\mathbf{x} - \mathbf{x}^s) d\Omega, \quad (2.5)$$

where $\mathbf{F}(s, t)$ is given in Eq. (2.1); and \mathbf{v}^s , $\mathbf{v}(\mathbf{x}, t)$, $p(\mathbf{x}, t)$, ρ , and μ represent the velocity of the immersed structural/solid point \mathbf{x}^s , the fluid velocity, pressure, density, and dynamic viscosity.

Since the elastic fiber does not occupy any volume, the net effects on the surrounding fluid can be represented by a set of forces associated with the immersed fiber points. Therefore, the fluid-structure interaction system can be simply depicted as a conventional fluid mechanics problem with an Eulerian kinematic description. Of course, all the difficulties are hidden in the representation of the inhomogeneous body force $\mathbf{f}(\mathbf{x}, t)$.

Notice that at a typical fiber point k , the resultant elastic or internal force \mathbf{F}_k is determined by the fiber configuration and material property. Furthermore, if there are I arcs passing through the point k , assuming that each fiber has a tension T_i and an increment of the arc length Δs_i , the j^{th} component of the nodal force $\mathbf{F}_k(s, t)$ is calculated as

$$\sum_{i=1}^I F_{kj}^i(s, t) = \sum_{i=1}^I \frac{\partial(T_i t_{kj}^i)}{\partial s} \Delta s_i, \quad (2.6)$$

where \mathbf{t}_k^i represents the unit tangent vector of the i^{th} arc at the point k .

Let us consider a typical point k as illustrated in Fig. 2, the nodal force at the point k , denoted as \mathbf{F}_k , is determined by tensions within the two adjacent fibers. If the immersed boundary has the same mass density as the surrounding fluid, i.e., is neutrally buoyant, at the submerged point k , there is no attached additional mass, and the resultant fluid force vector \mathbf{R}_k^f and the external solid force vector \mathbf{R}_k^e must equal the resultant elastic or internal nodal force \mathbf{F}_k . However, if the mass density of the elastic structure ρ_s is different from the surrounding fluid density ρ_f , \mathbf{R}_k^f and \mathbf{R}_k^e must achieve a dynamic equilibrium with the resultant elastic nodal force and the inertial or D'Alembert's force. Denote M and \mathbf{a}_k as the lumped mass and the acceleration at the point k , we have

$$\mathbf{R}_k^f + \mathbf{R}_k^e = \begin{cases} \mathbf{F}_k, & \rho_s = \rho_f, \\ \mathbf{F}_k + M\mathbf{a}_k, & \rho_s \neq \rho_f. \end{cases} \quad (2.7)$$

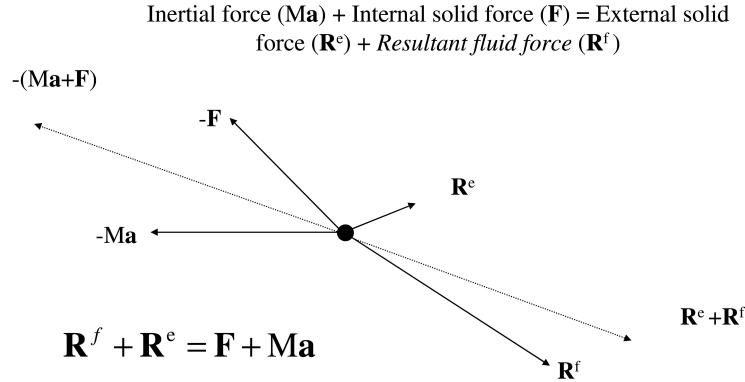


Figure 2: Force balance at a typical submerged point.

Again, this equilibrium depicted by Eq. (2.7) applies for *both* the boundary and the interior material points. Moreover, employing the discretized delta function, we can write the discretized form of Eqs. (2.4) and (2.5) as

$$\mathbf{f} = \sum_{k=1}^K \delta_h(\mathbf{x} - \mathbf{x}_k^s)(-\mathbf{R}_k^f), \quad (2.8)$$

$$\mathbf{v}_k^s = \sum_{\mathbf{x}_i \in \Omega_o^k} \delta_h(\mathbf{x}_i - \mathbf{x}_k^s) \mathbf{v}_i h^3, \quad (2.9)$$

where K is the total number of submerged points, Ω_o^k stands for the finite support domain of the discretized delta function surrounding and centered at the k^{th} submerged point, and h is the grid size of the uniform background grid for the entire fluid domain.

As shown in Fig. 3, various flexible structures as well as immersed points can be modeled by the immersed boundary method. To better illustrate why the immersed boundary method works, we employ the familiar variational principles. Consider a fluid domain Ω enclosed with a sufficiently smooth boundary, $\partial\Omega = \Gamma_v \cup \Gamma_f$, as depicted in Fig. 4, where Γ_v and Γ_f stand for the Dirichlet and Neumann boundaries, respectively. Suppose there exists an enclosed elastic boundary Γ_s (a line for two-dimensional cases and a surface for three-dimensional cases) as the generic representation of the immersed boundary, the fluid domain Ω is subdivided into two regions, namely, the interior region Ω_i and the exterior region Ω_e . Therefore, the boundaries of the interior and the exterior regions can be simply expressed as $\partial\Omega_i = \Gamma_s$ and $\partial\Omega_e = \Gamma_s \cup \Gamma_v \cup \Gamma_f$.

Denote $\boldsymbol{\sigma}$ as the stress tensor, \mathbf{v} as the velocity vector, and ρ as the density in the fluid domain, we establish the following governing set of equations (strong form):

$$\rho \dot{v}_i = \sigma_{ij,j} + f_i^e, \text{ in } \Omega_i (\text{or } \Omega \setminus \Omega_e), \quad (2.10)$$

$$\rho \dot{v}_i = \sigma_{ij,j} + f_i^e, \text{ in } \Omega_e, \quad (2.11)$$

$$[v_i] = 0, \text{ on } \Gamma_s, \text{ kinematic matching}, \quad (2.12)$$

$$[\sigma_{ij} n_j] = f_i^s + m \ddot{u}_i^s, \text{ on } \Gamma_s, \text{ dynamic matching}, \quad (2.13)$$

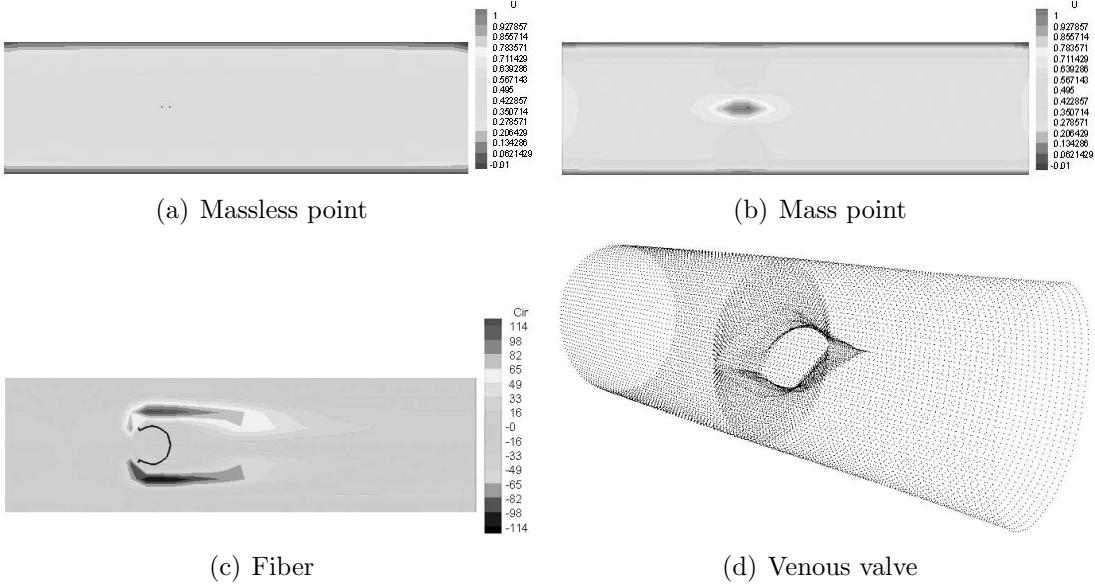


Figure 3: Illustration of the construction of simple and complicated immersed structures with immersed points.

where the external body force \mathbf{f}^e will be replaced by $\rho\mathbf{g}$, with \mathbf{g} as the gravitational acceleration; \mathbf{f}^s and m stand for the elastic force and the mass density of the immersed boundary Γ_s (per unit length for two-dimensional cases and per unit area for three-dimensional cases); \mathbf{u}^s denotes the interface displacement; and the surface normal vector \mathbf{n} is aligned with that of the interior fluid domain \mathbf{n}^i and opposite to that of the exterior fluid domain \mathbf{n}^e .

At this point, we can derive a number of numerical approaches to solve Eqs. (2.10) to (2.13). A straightforward approach is to represent the exterior and the interior fluid domains with different meshes and to match them accordingly at the interface Γ_s . This approach represents the traditional treatment of fluid-structure interaction problems, in which the solid mesh is coupled with the fluid mesh around the fluid-structure interface [19] [30].

Define the Sobolev space $[H_{0,\Gamma_v}^1(\Omega)]^d = \{\mathbf{w} \mid \mathbf{w} \in [H^1(\Omega)]^d, \mathbf{w}|_{\Gamma_v} = \mathbf{0}\}$, where d represents the spatial dimensions, we express Eqs. (2.10) to (2.13) in the variational form (weak form): $\forall \mathbf{w} \in [H_{0,\Gamma_v}^1(\Omega)]^d$

$$\int_{\Omega_i} w_i [\rho(\dot{v}_i - g_i) - \sigma_{ij,j}] d\Omega + \int_{\Omega_e} w_i [\rho(\dot{v}_i - g_i) - \sigma_{ij,j}] d\Omega = 0. \quad (2.14)$$

Remark 2.1 In the variational form, $\mathbf{w} \in [H_{0,\Gamma_v}^1(\Omega)]^d$ implies that the kinematic matching at the interface Γ_s written as Eq. (2.12) is satisfied for all \mathbf{w} .

Furthermore, using integration by parts and the divergence theorem, introducing dynamic matching at the interface Γ_s , and combining the interior and exterior fluid domains with $\Omega_e \cup \Omega_i = \Omega$, Eq. (2.14) can be rewritten as: $\forall \mathbf{w} \in [H_{0,\Gamma_v}^1(\Omega)]^d$

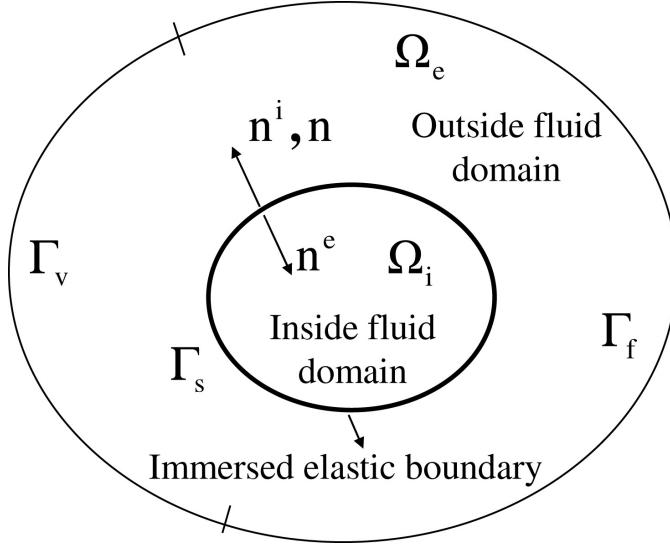


Figure 4: Immersed boundary illustration.

$$\int_{\Omega} [w_i \rho(\dot{v}_i - g_i) + w_{i,j} \sigma_{ij}] d\Omega + \int_{\Gamma_s} w_i^s (f_i^s + m\ddot{u}_i^s) d\Gamma - \int_{\Gamma_f} w_i f_i^{\Gamma_f} d\Gamma = 0. \quad (2.15)$$

Remark 2.2 In Eq. (2.15), the term involving the given surface traction \mathbf{f}^{Γ_f} will remain the same as if the variational form is carried out for the entire fluid domain instead of the interior and exterior parts. Thus the focus will be on the submerged interface Γ_s .

Remark 2.3 In Eq. (2.15), the external work comes from the external body force \mathbf{f}^e or $\rho\mathbf{g}$, the surface traction \mathbf{f}^{Γ_f} at the Neumann boundary Γ_f , and the elastic and inertial forces around the submerged interface Γ_s . Moreover, in Eq. (2.15), we do not stipulate the material derivative $\frac{d\mathbf{v}}{dt}$ and the stress $\boldsymbol{\sigma}$. Hence the turbulent and the non-Newtonian fluid models can eventually be incorporated. Finally, the kinematic matching at the submerged interface Γ_s also implies that the submerged interface will move at the same velocity as that of the fluid particles in the immediate vicinity.

In the immersed boundary method, we introduce the following two key equations:

$$f_i^{fsi} = - \int_{\Gamma_s} (f_i^s + m\ddot{u}_s) \delta(\mathbf{x} - \mathbf{x}^s) d\Gamma, \quad (2.16)$$

$$v_i^s = \int_{\Omega} v_i \delta(\mathbf{x} - \mathbf{x}^s) d\Omega, \quad (2.17)$$

where \mathbf{f}^{fsi} is the so-called equivalent body force.

Remark 2.4 In both Eqs. (2.16) and (2.17), the Dirac delta function is positioned at the current interface position \mathbf{x}^s . Before the discretization of the Dirac delta function, Eq. (2.17) can be simply

interpreted as the evaluation of the fluid velocity at the submerged interface. In the discretized form, the Dirac delta function in Eq. (2.17) is equivalent to the shape function or kernel of the meshfree method.

Note that Γ_s represents the current configuration of the submerged interface, and the nonlinear mechanics is employed to relate the elastic force \mathbf{f}^s with the interfacial position \mathbf{x}^s or the displacement \mathbf{u}^s . It is also clear that as long as we use the same delta function for both Eqs. (2.16) and (2.17), the virtual power input from the submerged elastic boundary (or the immersed boundary) to the fluid domain can be expressed as

$$\int_{\Omega} w_i f_i^{fsi} d\Omega = - \int_{\Gamma_s} \int_{\Omega} w_i \delta(\mathbf{x} - \mathbf{x}^s) (f_i^s + m\ddot{u}_i^s) d\Omega d\Gamma = - \int_{\Gamma_s} w_i^s (f_i^s + m\ddot{u}_i^s) d\Gamma. \quad (2.18)$$

Because Eq. (2.18) holds for all $\mathbf{w} \in [H_{0,\Gamma_v}^1(\Omega)]^d$, the effect of the submerged elastic boundary can be simply replaced with the equivalent body force \mathbf{f}^{fsi} . Hence the governing equations (2.10) to (2.13) can be rewritten as

$$\rho \dot{v}_i = \sigma_{ij,j} + \rho g_i + f_i^{fsi}, \text{ in } \Omega; \quad (2.19)$$

and the variational equations (2.14) and (2.15) are modified as

$$\int_{\Omega} [w_i (\rho \dot{v}_i - \rho g_i - f_i^{fsi}) + w_{i,j} \sigma_{ij}] d\Omega - \int_{\Gamma_f} w_i f_i^{\Gamma_f} d\Gamma = 0. \quad (2.20)$$

Remark 2.5 Eq. (2.20) provides us with the foundation of the key advantage of the immersed boundary method, namely, the independent solid mesh moves on top of a background fluid mesh. Moreover, we must also point out that the background fluid mesh could be a fixed Eulerian mesh or an arbitrary Lagrangian-Eulerian (ALE) mesh with a prescribed mesh motion. In practice, such a mesh motion could follow the moving structures or solids as well as conform to the boundary deformation.

3 Extension to Immersed Incompressible Continuum

Instead of representing the immersed structures/solids with fibers as in Fig. 3, the goal of the extended immersed boundary method or the immersed finite element method [41] [49] is to represent the immersed incompressible solid with finite mass and volume with nonlinear finite element formulations. The initial attempt of connecting a traditional linear elasticity model with the immersed boundary method can be traced back to Sulsky and Brackbill, 1991 [27], in which a stress function is transferred to the fluid grid. In recent finite element extensions of the immersed boundary method, a more direct connection between the fluid and solid domains is accomplished by employing the internal nodal forces calculated in the context of finite element methods. In these new attempts, submerged solids can experience large displacements and deformations. We must point out that contrary to the traditional method of tracking the fluid-structure interfaces both the interior and the fluid-structure interface of the submerged structure are modelled as submerged material points.

Note that although the *entire* submerged solid domain is decomposed into a collection of *submerged* material points, due to the distributed nodal forces, with a sufficiently dense solid mesh, the surrounding fluid will not penetrate into the structure interior and the fluid-structure interface will be *automatically* defined by the submerged material points enclosing the solid domain. The physical interpretation of finite element methods can be simplified as replacing a continuous medium with a collection of nodes or material points linked through a stiffness matrix. Therefore, if the structure is approximated with an equivalent fiber network, the stiffness matrix can be derived from truss elements or a finite difference scheme as used in the immersed boundary method. In fact, it was shown that the nodal force calculation in the immersed boundary is equivalent to a traditional nonlinear finite element formulation for truss elements using the second Piola-Kirchhoff stress and the Green-Lagrangian strain [37].

If the surrounding fluid is viscous and incompressible, the immersed solid *must be incompressible* in immersed methods. In addition, there are two views of immersed solids. The first one which matches with the original understanding of the immersed boundary method, namely, the immersed solid is wet and permeated with the same fluid as the surrounding. Therefore, the elasticity forces will be the additional force due to the solid portion of the immersed solid. This understanding is very realistic for biological system modeling, since the tissues are mostly fluid-solid systems and the elastic parts are contributed by elastomer, collagen, or other solid constituents. On the other hand, the second understanding of the immersed solid is more in tune with the traditional fluid-structure interaction systems in which the immersed solid is dry and impermeable. In this context, if we were to use the immersed methods to handle this type of fluid-structure system, the immersed solid must be incompressible to match with the surrounding incompressible fluid.

In the initial versions of the finite element formulations of immersed methods [3] [4] [41] [49], the immersed solids are assumed to be incompressible. Furthermore, if the immersed solids are impermeable, the additional elastic forces in comparison with the viscous counterparts calculated with the material properties of the surrounding fluid are very large, both views for immersed solids yield the same forces.

In the extended immersed boundary method, the fluid solver is still the same spectrum solver with periodic boundary conditions based on Chorin's algorithm and fast Fourier transform [5]. Since the background fluid mesh is uniform, the discretized delta function derived in the immersed boundary method is employed. In the immersed finite element method, the fluid solver is replaced with the stabilized finite element formulation and the discretized delta function is replaced with the kernel functions for meshless methods [29].

Consider a general three-dimensional incompressible hyperelastic material model with the following Mooney-Rivlin material description

$$W = C_1(J_1 - 3) + C_2(J_2 - 3), \quad (3.21)$$

with

$$J_1 = I_1, \quad J_2 = I_2, \quad I_1 = C_{kk}, \quad I_2 = (I_1^2 - C_{ij}C_{ij})/2, \quad (3.22)$$

where W is the elastic energy potential and \mathbf{C} is the Cauchy-Green deformation tensor defined as $\mathbf{C} = \mathbf{D}^T \mathbf{D}$ with the solid deformation gradient $D_{ij} = \partial x_i^s(t)/\partial x_j^s(0)$.

Note that since the solid displacements are mapped from the background fluid, if the surrounding fluid is incompressible, the solid must also be incompressible, which corresponds to $J_3 = I_3^{1/2} = (\det(\mathbf{C}))^{1/2} = 1$. For structures with large displacements and deformations, the 2nd Piola-Kirchhoff stress \mathbf{S} and the Green-Lagrangian strain ϵ are used along with a total Lagrangian formulation. Hence, employing Eq. (3.21), we derive

$$S_{kl} = \frac{\partial W}{\partial \epsilon_{kl}} \text{ and } \epsilon_{ij} = \frac{1}{2}(C_{ij} - \delta_{ij}). \quad (3.23)$$

Furthermore, the Cauchy stress can be calculated as

$$\tau_{ij} = \frac{1}{\det(\mathbf{D})} D_{im} S_{mn} D_{jn}. \quad (3.24)$$

Thus, the equivalent internal force for the material points of the flexible structure can be derived as

$$\mathbf{F}_k = \frac{\partial}{\partial \mathbf{x}_k^s} \left(\int_{\Omega_s(t)} W d\Omega \right) = \int_{\Omega_s(0)} S_{ml} \frac{\partial \epsilon_{ml}}{\partial \mathbf{x}_k^s} d\Omega, \quad (3.25)$$

where \mathbf{x}_k^s and \mathbf{F}_k stand for the current position vector of the k^{th} submerged node and the corresponding internal nodal force vector, respectively, and $\Omega_s(t)$ and $\Omega_s(0)$ represent the current and the original volume of the submerged solid.

Note that if the nonlinear structural material has a density ρ_s different from the fluid density ρ_f , we should include the inertial force. Similar to the discussion in Section 2, the resultant node force vector \mathbf{R} can be expressed for each material point:

$$\mathbf{R} = \mathbf{R}^f + \mathbf{R}^e = \begin{cases} \mathbf{F}, & \rho_s = \rho_f, \\ \mathbf{F} + \mathbf{M}\ddot{\mathbf{U}}, & \rho_s \neq \rho_f, \end{cases} \quad (3.26)$$

where \mathbf{U} represents the nodal displacement vector, and the consistent mass matrix \mathbf{M} is defined as

$$\mathbf{M} = \int_{\Omega_s(0)} (\rho_s - \rho_f) \mathbf{H}^T \mathbf{H} \det(\mathbf{D}) d\Omega, \quad (3.27)$$

with \mathbf{H} as the interpolation matrix.

Consequently, for the immersed solid, Eqs. (2.4) and (2.5) are extended to the following:

$$\mathbf{f} = - \int_{\Omega_s(t)} \mathbf{R}^f \delta(\mathbf{x} - \mathbf{x}^s) d\Omega, \quad (3.28)$$

$$\mathbf{R}^f = \int_{\Omega_s(0)} \mathbf{S} \frac{\partial \epsilon}{\partial \mathbf{x}^s} d\Omega + \int_{\Omega_s(0)} (\rho_s - \rho_f) \mathbf{H}^T \mathbf{H} \det(\mathbf{D}) d\Omega \ddot{\mathbf{U}} - \mathbf{R}^e, \quad (3.29)$$

$$\mathbf{v}^s = \int_{\Omega} \mathbf{v} \delta(\mathbf{x} - \mathbf{x}^s) d\Omega, \quad (3.30)$$

where Ω_s represents the original solid configuration employed in the total Lagrangian formulation.

Eqs. (2.2) and (2.3) are the governing equations for the fluid, Eq. (3.29) has to be calculated for each material point, and Eqs. (3.28) and (3.30) distribute the solid force vector to the fluid and the fluid velocity on the solid material using the same delta function. Notice that Eq. (3.28) is carried out in the current configuration of the submerged solid.

4 Immersed Continuum Method

In the case of the incompressible solid immersed in the incompressible fluid, if there is no excessively stiff boundary springs connecting with the tether points [15] and the additional elasticity moduli of the immersed solid are reasonable, the traditional explicit scheme can be employed. This is in fact addressed in the extended immersed boundary method or immersed finite element method [41] [49]. In this section, we mainly address the issue of implicit formulations for excessively stiff boundary springs and high elastic moduli. In particular, we use a general case of compressible solid immersed in compressible fluid to cover one seemingly very different scenario, namely, compressible solid immersed in incompressible fluid. In the immersed continuum method, we use a physical argument such that the bulk modulus of the solid can be much higher than the bulk modulus of the fluid. With the assumption that the acoustic wave speed within the fluid is assumed to be constant, a so-called pseudo compressible fluid model is often used in numerics to mimic incompressible fluid behaviors. Why do we have to bother to use this type of fluid model coupled with compressible solid? To use the concept of immersed methods, we need to replace the immersed solid with a fluid similar to the surrounding fluid. If the volume of the immersed solid changes ever so slightly, the volume of the corresponding fluid must also change, hence, the fluid model of the surrounding fluid must also be slightly compressible.

It is important to point out that here the immersed solid is viewed as impermeable thus such a volume of fluid does not exist physically. To account for the correct effect of the submerged solid exerting on the surrounding fluid, we must subtract the inertial force, the external body force, and the internal stress effects of such an imaginary fluid volume $\Omega_s(t)$. This idea is very similar to the fictitious domain method which was proposed to handle immersed rigid bodies [13]. The advantage of the fictitious domain method is the use of a fixed background fluid mesh whereas the immersed rigid bodies are modeled with the Lagrangian description. Recently, such an approach has been extended to immersed flexible bodies [33] [48]. It is the author's opinion that in full fledged implicit forms there must be some hidden mathematical equivalence of these two sets of approaches, namely, the immersed boundary/continuum methods and the fictitious domain method. Maybe some day, an inf-sup condition similar to the modeling of almost incompressible materials can be derived in the context of the treatment of compressible solids immersed in compressible fluids [2].

To clearly present this new form of immersed methods dealing with immersed continuum, we study the similar variational forms. Consider the same domain Ω , as depicted in Fig. 5, suppose there exists a submerged solid domain Ω_s enclosed by a sufficiently smooth boundary Γ_s (a line for two-dimensional cases and a surface for three-dimensional cases), the entire domain Ω is subdivided into two regions, namely, the solid region Ω_s and the fluid region Ω_f . Therefore, the boundaries of the solid and the fluid regions can be simply expressed as $\partial\Omega_s = \Gamma_s$ and $\partial\Omega_f = \Gamma_s \cup \Gamma_v \cup \Gamma_f$. Denote σ as the stress tensor, \mathbf{v} as the velocity vector, we establish the following set of governing equations (strong form):

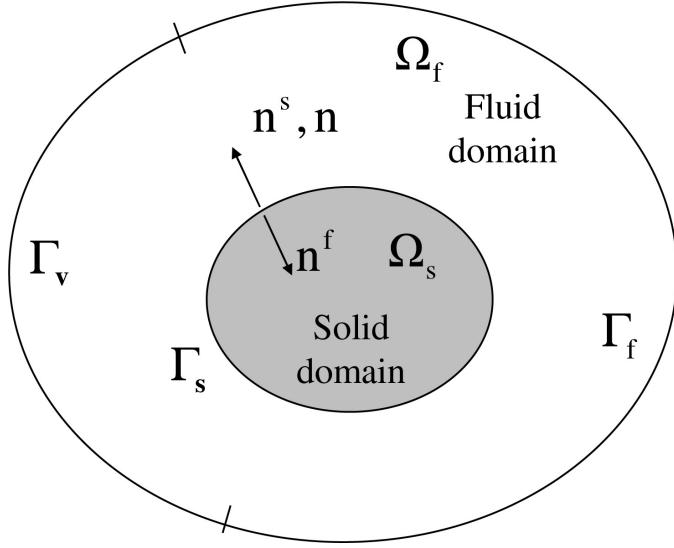


Figure 5: Immersed continuum illustration.

$$\rho_s \dot{v}_i^s = \sigma_{ij,j}^s + \rho_s g_i, \text{ in } \Omega_s, \quad (4.31)$$

$$\rho_f \dot{v}_i^f = \sigma_{ij,j}^f + \rho_f g_i, \text{ in } \Omega_f, \quad (4.32)$$

$$[v_i] = 0, \text{ on } \Gamma_s, \text{ kinematic matching}, \quad (4.33)$$

$$[\sigma_{ij} n_j] = 0, \text{ on } \Gamma_s, \text{ dynamic matching}, \quad (4.34)$$

where the surface normal vector \mathbf{n} is aligned with that of the solid domain \mathbf{n}^s and opposite to that of the fluid domain \mathbf{n}^f .

Note that we use the subscript (for scalars) or the superscript (for vectors) s and f represent the solid and fluid domains, respectively. Define the same Sobolev space $[H_{0,\Gamma_v}^1(\Omega)]^d$, we express Eqs. (4.31) to (4.34) in the variational form (weak form): $\forall \mathbf{w} \in [H_{0,\Gamma_v}^1(\Omega)]^d$

$$\int_{\Omega_s} w_i [\rho_s (\dot{v}_i^s - g_i) - \sigma_{ij,j}^s] d\Omega + \int_{\Omega_f} w_i [\rho_f (\dot{v}_i^f - g_i) - \sigma_{ij,j}^f] d\Omega = 0. \quad (4.35)$$

Remark 4.1 $\mathbf{w} \in [H_{0,\Gamma_v}^1(\Omega)]^d$ also implies that the kinematic matching of Eq. (4.33) is satisfied for all \mathbf{w} .

Again, using integration by parts and the divergence theorem, introducing dynamic matching at the interface Γ_s , and combining the solid and fluid domains with $\Omega_s \cup \Omega_f = \Omega$, Eq. (4.35) can be rewritten as: $\forall \mathbf{w} \in [H_{0,\Gamma_v}^1(\Omega)]^d$

$$\int_{\Omega} [w_i \rho_f (\dot{v}_i - g_i) + w_{i,j} \sigma_{ij}] d\Omega - \int_{\Gamma_f} w_i f_i^{\Gamma_f} d\Gamma - \int_{\Omega_s} w_i^s f_i^s d\Omega = 0, \quad (4.36)$$

with

$$\int_{\Omega_s} w_i^s f_i^s d\Omega = - \int_{\Omega_s} [w_i(\rho_s - \rho_f)(\dot{v}_i - g_i) + w_{i,j}(\sigma_{ij}^s - \sigma_{ij}^f)] d\Omega. \quad (4.37)$$

Remark 4.2 In Eq. (4.34), the unit surface normal vectors at the fluid-solid interface Γ_s are assigned as $\mathbf{n}^s = -\mathbf{n}^f = \mathbf{n}$; whereas in Eq. (4.37) the term involving the given surface traction \mathbf{f}^{Γ_f} will remain the same if the variational forms are carried out for the entire domain Ω instead of the solid and fluid parts. Thus the focus will be on the submerged solid Ω_s and its interface with the fluid Γ_s .

Remark 4.3 In Eq. (4.36), within the entire domain Ω , the external work comes from the external body force $\rho_f \mathbf{g}$, and the surface traction \mathbf{f}^{Γ_f} at the Neumann boundary Γ_f ; whereas the power input within the submerged solid domain Ω_s includes the contribution from the inertial force difference, the buoyancy force, and the internal energy difference. Again, the kinematic matching at the submerged interface Γ_s also implies that the submerged interface will move at the same velocity as that of the fluid particles in the immediate vicinity. In Eq. (4.36), we again do not stipulate the material derivative $\frac{d\mathbf{v}}{dt}$ and the stress $\boldsymbol{\sigma}$. Hence the turbulent and the non-Newtonian fluid models can eventually be incorporated.

In the explicit implementation, just as in the immersed boundary method, in the extended immersed boundary method and immersed finite element method [41] [49], we introduce the following two key equations to synchronize the fluid occupying the submerged solid domain Ω_s with the solid and distribute the solid force \mathbf{f}^s :

$$f_i^{fsi} = \int_{\Omega_s} f_i^s \delta(\mathbf{x} - \mathbf{x}^s) d\Omega, \quad (4.38)$$

$$v_i^s = \int_{\Omega} v_i \delta(\mathbf{x} - \mathbf{x}^s) d\Omega, \quad (4.39)$$

where \mathbf{f}^{fsi} represents the same equivalent body force as in the immersed boundary method.

It is very important to recognize that \mathbf{f}^s is the force density within the solid domain Ω_s ; whereas \mathbf{f}^{fsi} is the equivalent body force over the entire domain Ω . The physical significance of \mathbf{f}^s and \mathbf{f}^{fsi} is quite different. As a consequence, the entire fluid-solid interaction system is represented with the same set of governing equations (2.19) (strong form) and the corresponding variational form (2.20) (weak form).

Remark 4.4 Eqs. (4.38) and (4.39) are comparable to Eqs. (2.16) and (2.17) in the IB method. In Eq. (4.37), \mathbf{f}^s can be considered as the equivalent force density within the submerged solid domain Ω_s . In fact, this force density directly corresponds to the rigid link between the fluid occupying the submerged solid domain Ω_s and the solid. In other words, the force density \mathbf{f}^s stands for the Lagrange multiplier corresponding to the constraint in Eq. (4.39). Of course, the definition in Eq. (4.37) also matches the virtual power input from the submerged solid domain Ω_s .

Note that the inertial force difference $-\int_{\Omega_s} w_i(\rho_s - \rho_f) \dot{v}_i d\Omega$ of the submerged solid continuum corresponds to the inertial force $-\int_{\Gamma_s} w_i^s m \ddot{u}_i^s d\Gamma$ of the submerged elastic boundary; whereas the internal energy difference $-\int_{\Omega_s} w_{i,j} (\sigma_{ij}^s - \sigma_{ij}^f) d\Omega$ of the submerged solid continuum corresponds to the elastic force $-\int_{\Gamma_s} w_i^s f_i^s d\Gamma$ of the submerged elastic boundary. In comparison with the submerged elastic boundary, the contribution of the submerged solid includes an additional term to account for the external body force difference (the so-called buoyancy) $\int_{\Omega_s} w_i(\rho_s - \rho_f) g_i d\Omega$. This buoyancy force is the direct manifestation of the submerged solid which unlike the submerged elastic boundary occupies a finite volume. Likewise, the inertial effect of the submerged solid includes the difference between the solid and fluid densities assuming the fluid occupying the submerged solid domain is forced to have the same motions as those of the solid.

5 Mixed Finite Element Formulations

We present here the velocity/pressure formulation for the compressible viscous fluid and the displacement/pressure formulation for the compressible solid with a hyperelastic material model. For simplicity, in this section, we omit the superscript or subscript f for fluid variables.

For the fluid domain, we adopt an Eulerian kinematic description, therefore, the material derivative of the fluid velocity is expressed as

$$\dot{v}_i = v_{i,t} + v_j v_{i,j}. \quad (5.40)$$

Although we refer to the fixed background fluid mesh, in the implementation, we also employ the arbitrary Lagrangian-Eulerian (ALE) kinematic description [10] [36] and replace the convective velocity in Eq. (5.40) with $\mathbf{v} - \mathbf{v}^m$, where \mathbf{v}^m stands for the given mesh velocity. For the solid domain, we employ a Lagrangian kinematic description, thus the fluid-solid interface will be tracked automatically by the position of solid particles. Moreover, there is no need for convective terms in the solid domain and the material derivative is the same as the time derivative. Hence, the solid velocity vector \mathbf{v}^s and the acceleration vector $\dot{\mathbf{v}}^s$ can be expressed as

$$\mathbf{v}^s = \dot{\mathbf{u}}^s \text{ and } \dot{\mathbf{v}}^s = \ddot{\mathbf{u}}^s, \quad (5.41)$$

with the displacement vector $\mathbf{u}^s(t) = \mathbf{x}^s(t) - \mathbf{x}^s(0)$, where $\mathbf{x}^s(t)$ and $\mathbf{x}^s(0)$ stand for the current and the original material point positions within the solid domain Ω_s .

In order to deal with the compressible viscous fluid, we subtract the pressure p from the stress components σ_{ij} and obtain the deviatoric stress components τ_{ij} , which is illustrated in a Newtonian fluid model,

$$\sigma_{ij} = -p \delta_{ij} + \tau_{ij}, \quad (5.42)$$

with $\tau_{ij} = \mu(v_{j,i} + v_{i,j})$.

Furthermore, to couple with the pressure unknown, the continuity equation of the compressible viscous fluid is expressed as

$$v_{i,i} + \frac{\dot{p}}{\kappa} = 0, \quad (5.43)$$

where κ is the bulk modulus of the fluid; and the material derivative \dot{p} can be simply expressed as $p_{,t} + v_i p_{,i}$ or $p_{,t} + (v_i - v_i^m) p_{,i}$ for the Eulerian or Arbitrary Lagrangian-Eulerian descriptions, respectively.

In the analysis of slightly compressible fluids, we assume the compressibility measured by bulk modulus κ is constant. Therefore, we have the following relationship between the density and the pressure of the fluid domain,

$$\frac{dp}{d\rho} = c^2 = \frac{\kappa}{\rho}. \quad (5.44)$$

From Eq. (5.44), it is straightforward to derive the following

$$p(t) - p(0) = \kappa \ln \frac{\rho(t)}{\rho(0)}. \quad (5.45)$$

Like the fluid stress tensor, we also decompose the solid stress tensor as a hydrostatic pressure p^s , and a deviatoric stress tensor τ_{ij}^s ,

$$\sigma_{ij}^s = -p^s \delta_{ij} + \tau_{ij}^s. \quad (5.46)$$

Unlike the fluid domain, since we use the Lagrangian description for the submerged solid, the treatments of the continuity equation and the Cauchy stress in nonlinear solid mechanics are not as straightforward. As a special case, if the submerged solid is a flexible structure with a linear elastic material law, we will only have the geometrical nonlinearity to deal with. In this case, suppose the Young's modulus and the Poisson ratio are E and ν , respectively, and the bulk modulus for the solid can be simply expressed as $\kappa^s = E/3(1 - 2\nu)$. However, Cauchy stress must still be depicted on the current configuration which itself is unknown.

In this work, we discuss a nonlinear solid mechanics model with both the geometrical and material nonlinearities [28] [41] [49]. First of all, we must introduce the solid deformation gradient $D_{ij} = \partial x_i^s(t)/\partial x_j^s(0)$, from which we can derive the Green-Lagrangian strain ϵ_{ij} . To obtain the energy conjugate stress S_{ij} , the second Piola-Kirchhoff stress, we must first introduce the elastic energy \bar{W} , which is often related to the three invariants of the Cauchy-Green deformation tensor \mathbf{C} defined as $\mathbf{D}^T \mathbf{D}$. Moreover, as discussed in Refs. [16] and [42], an elastic energy term $-[p^s + \kappa^s(J_3 - 1)]^2/2\kappa^s$ is added to \bar{W} , along with the solid unknown pressure p^s introduced as

$$J_3 - 1 + \frac{p^s}{\kappa^s} = 0, \quad (5.47)$$

where κ^s is the solid bulk modulus and J_3 stands for the determinant of the deformation gradient.

Of course, to match with the expression in Eq. (5.46), the solid Cauchy stress is converted from the second Piola-Kirchhoff stress,

$$\sigma_{ij}^s = \frac{1}{\det(\mathbf{D})} D_{im} S_{mn} D_{jn}. \quad (5.48)$$

Finally, since the solid displacement is dependent on the fluid velocity, the primary unknowns for the coupled fluid-solid system are the fluid velocity \mathbf{v} , the fluid pressure p , and the solid pressure p^s .

Define the Sobolev spaces, so the weak form of governing equations can be modified as: $\forall q \in L^2(\Omega)$, $q^s \in L^2(\Omega_s)$, $\mathbf{w} \in [H_{0,\Gamma_v}^1(\Omega)]^d$, which includes $\forall \mathbf{w}^s \in [H^1(\Omega_s)]^d$, and find \mathbf{v} and p in Ω , p^s in Ω_s , such that

$$\begin{aligned} & \int_{\Omega} w_i \rho (\dot{v}_i - g_i) d\Omega + \int_{\Omega} (w_{i,j} \tau_{ij} - p w_{i,i}) d\Omega - \int_{\Gamma_f} w_i f_i^{\Gamma_f} d\Gamma \\ & + \int_{\Omega_s} [w_i^s (\rho_s - \rho) (\dot{v}_i - g_i) + w_{i,j}^s (\tau_{ij}^s - \tau_{ij}^f) - (p^s - p) w_{i,i}^s] d\Omega + \\ & + \int_{\Omega} q (v_{j,j} + \frac{p_{,t}}{\kappa}) d\Omega + \int_{\Omega_s} q^s (J_3 - 1 + \frac{p^s}{\kappa^s}) d\Omega = 0. \end{aligned} \quad (5.49)$$

We recognize that there are two sets of discretizations, namely, one for the Lagrangian solid mesh and the other one for the Eulerian fluid mesh. The stabilization finite element scheme in this paper is similar to what has been published by Wang [36]. Note that the convective terms are hidden in \dot{v}_i^h and the detailed expressions of other stabilized Galerkin formulation for the Navier-Stokes equations can also be found in Refs. [29] [30] [32] [49].

In general the interpolation functions for the velocity vector and the unknown pressures are different. Therefore, we retain the superscripts v and p to denote such differences. For the fluid domain Ω , the following interpolations are used:

$$\mathbf{v}^h = h_I^v \mathbf{v}_I, \quad \mathbf{w}^h = h_I^v \mathbf{w}_I, \quad p^h = h_I^p p_I, \quad q^h = h_I^p q_I, \quad (5.50)$$

where h_I^v and h_I^p stand for the interpolation functions at node I for the velocity vector and the pressure; and \mathbf{v}_I , \mathbf{w}_I , p_I , and q_I are the nodal values of the discretized velocity vector, admissible velocity variation, pressure, and pressure variation, respectively.

For the solid domain Ω_s , the discretization is based on the following:

$$\mathbf{u}^{s,h} = h_J^u \mathbf{u}_J^s, \quad \mathbf{w}^{s,h} = h_J^u \mathbf{w}_J^s, \quad p^{s,h} = h_J^{p^s} p_J^s, \quad q^{s,h} = h_J^{p^s} q_J^s, \quad (5.51)$$

where h_J^u and $h_J^{p^s}$ stand for the interpolation functions at node J for the displacement vector and the unknown pressures; and $\mathbf{u}_J^{s,h}$, $\mathbf{w}_J^{s,h}$, $p_J^{s,h}$, and $q_J^{s,h}$ are the nodal values of the discretized displacement vector, admissible velocity variation, pressure, and pressure variation, respectively.

Substituting both discretizations (5.50) and (5.51) into Eq. (5.49), we obtain the following discretization of the weak form: $\forall q^h \in L^2(\Omega^h)$, $q^{s,h} \in L^2(\Omega_s^h)$, $\mathbf{w}^h \in [H_{0,\Gamma_v^h}^{1,h}(\Omega^h)]^d$, which includes $\forall \mathbf{w}^{s,h} \in [H^{1,h}(\Omega_s^h)]^d$,

$$\begin{aligned}
& \int_{\Omega^h} w_{iI} h_I^v \rho \dot{v}_i^h d\Omega - \int_{\Gamma_f^h} w_{iI} h_I^v f_i^{\Gamma_f^h} d\Gamma + \int_{\Omega^h} (w_{iI} h_{I,j}^v \tau_{ij} - p^h w_{iI} h_{I,i}^v) d\Omega \\
& + \int_{\Omega_s^h} [w_{iJ}^s h_J^u (\rho_s - \rho) (\dot{v}_i^h - g_i) + w_{iJ}^s h_{J,j}^u (\sigma_{ij}^s - \sigma_{ij}^f)] d\Omega - \int_{\Omega^h} w_{iI} h_I^v \rho g_i d\Omega \\
& + \int_{\Omega^h} q_I h_I^p (v_{j,j}^h + \frac{p_{,t}^h}{\kappa}) d\Omega + \int_{\Omega_s^h} q_J^s h_J^{p^s} (J_3 - 1 + \frac{p^{s,h}}{\kappa^s}) d\Omega = 0.
\end{aligned} \tag{5.52}$$

For clarity, we introduce a displacement nodal unknown vector \mathbf{U} , although it is only evaluated in the solid domain Ω_s in which a Lagrangian description is prescribed. In fact, within the solid domain, \mathbf{U} is denoted as \mathbf{U}^s and evolves according to $\dot{\mathbf{U}}^s$ and $\ddot{\mathbf{U}}^s$ which are mapped from the velocity nodal unknown vector \mathbf{V} and acceleration nodal unknown vectors $\dot{\mathbf{V}}$ for the fluid domain. Mathematically, we could say that \mathbf{v}^s is \mathbf{v} directly evaluated at the material point \mathbf{x}^s . Likewise, the pressure nodal unknown vectors \mathbf{P} and \mathbf{P}^s are introduced for the fluid and solid domains, respectively. Moreover, in the discussion of numerical procedures, we denote the time derivative of a variable a as \dot{a} . In this paper, for simplicity, we employ the Newton-Raphson iteration, and apply the Newmark time integration scheme as discussed in Refs. [16] [18] [31] [34] [35]. For a typical state variable a , in the incremental analysis, we have

$$\begin{aligned}
a(t + \Delta t) &= a(t) + \dot{a}(t)\Delta t + [(0.5 - \alpha)\ddot{a}(t) + \alpha\ddot{a}(t + \Delta t)]\Delta t^2, \\
\dot{a}(t + \Delta t) &= \dot{a}(t) + [(1 - \beta)\ddot{a}(t) + \beta\ddot{a}(t + \Delta t)]\Delta t,
\end{aligned} \tag{5.53}$$

where the unknown a stands for the scalar components of the nodal or discretized variables for \mathbf{V} , \mathbf{P} , and \mathbf{P}^s , and α and β are selected integration constants.

As a consequence, at every time step, the following nonlinear residual equations can be derived as

$$\mathbf{r}(\mathbf{V}, \mathbf{P}, \mathbf{P}^s) = \mathbf{0}. \tag{5.54}$$

6 Mapping between Fluid and Solid Domains

In the immersed boundary/continuum methods, the nonlinear mapping from the fluid mesh to the solid mesh is essential. At a typical solid node J , with a finite support domain Ω_J , the discretized form of the constraint of the velocities of the immersed solid and the corresponding fluid occupying the same solid domain can be expressed as

$$\mathbf{v}_J^s = \sum_I \mathbf{v}_I \phi_I(\mathbf{x}_I - \mathbf{x}_J^s), \forall \mathbf{x}_I \in \Omega_J, \tag{6.55}$$

where $\phi_I(\mathbf{x}_I - \mathbf{x}_J)$ is the kernel function centered at the solid node J , represented with \mathbf{x}_J^s .

It is very important to realize that the material points of the submerged solid domain will move in the entire domain, therefore even if we do not adjust the size of the support domain attached to these material points, Eq. (6.55) represents a nonlinear mapping which in this work for convenience is simply denoted as \mathcal{M} .

Note that in general within the solid domain, we have the stress difference $\sigma_{ij}^s - \sigma_{ij}^f$ or $-(p^s - p^f)\delta_{ij} + (\tau_{ij}^s - \tau_{ij}^f)$, in addition to the mapping of the velocity vector in Eq. (6.55). In order to use the definition of σ_{ij}^f , we must also map the unknown pressure from the fluid mesh denoted with node I to the solid mesh denoted with node J . In this case, it is beneficial to use the continuous pressure mixed finite element formulation for both fluid and solid domains [40] [2]. Consequently, like Eq. (6.55), we have

$$p_J^f = \sum_I p_I \phi_I(\mathbf{x}_I - \mathbf{x}_J), \forall \mathbf{x}_I \in \Omega_J. \quad (6.56)$$

Finally for the entire domain Ω , due to the arbitrariness of the variations w_{II} , q_I , and q_J^s , we have four equations at each fluid node I and one equation at each solid node J ,

$$r_{II}^v = 0, \quad r_I^p = 0, \quad r_J^{p^s} = 0, \quad (6.57)$$

where the residuals are defined as

$$\begin{aligned} r_{II}^v &= \int_{\Omega^h} h_I^v \rho v_i^h d\Omega + \int_{\Omega^h} [h_{I,j}^v \tau_{ij} - p^h h_{I,i}^v] d\Omega - \int_{\Gamma_f^h} h_I^{v,\Gamma_f^h} f_i^{\Gamma_f^h} d\Gamma \\ &\quad + \int_{\Omega_s^h} \mathcal{M}[h_J^u (\rho_s - \rho) (\dot{v}_i^h - g_i) + h_{J,j}^u (\sigma_{ij}^s - \sigma_{ij}^f)] d\Omega - \int_{\Omega^h} h_I^v \rho g_i d\Omega, \\ r_I^p &= \int_{\Omega^h} h_I^p (v_{j,j}^h + \frac{p_{,t}^h}{\kappa}) d\Omega, \\ r_J^{p^s} &= \int_{\Omega_s^h} h_J^{p^s} (J_3 - 1 + \frac{p^{s,h}}{\kappa^s}) d\Omega. \end{aligned} \quad (6.58)$$

It turns out that such discretized mapping using various kernel functions has been studied recently in the meshless finite element methods. For example, reproducing kernel particle method (RKPM) was proposed as an alternative or enhancement to various numerical procedures including finite element methods (Liu *et al.*, 1995 and 1996, [43] [45] [47] and Li and Liu, 1999 [20]). Unlike the discretized delta function originally proposed in the immersed boundary method [6], the kernel functions in the meshless methods can handle non-uniform meshing, which marks an important improvement for the increase of the local resolutions near the interfaces. Furthermore, the adjustable reproducing properties of the meshless kernels enable a better representation of the discretized delta function in the frequency domain, namely, as the polynomial order $n \rightarrow \infty$, the discretized delta function ϕ becomes flatter at $\omega = 0$ and approaches to an ideal filter in the frequency domain.

In the immersed boundary method, all discretized delta functions are continuous and have finite support points within a uniform background mesh. Moreover, the three-dimensional delta function is simply constructed with the multiplication of one-dimensional delta functions in three directions, i.e., $\delta(\mathbf{x}) = \delta(x_1)\delta(x_2)\delta(x_3)$. Consider first a one-dimensional case, in accordance with the translation invariance [5], for all r , where r is the parameter representing the position of the submerged boundary point and is scaled with respect to the grid size h , the discretized delta function satisfies $\sum_j \phi^2(r - j) = C$, where C is a numerical constant. In addition, to uniquely define the

discretized delta function for all r , we also have $\sum_j (r-j)^m \phi(r-j) = 0$, where the selection of the m^{th} moment depends on the number of support points. For instance, the discretized delta function with four support points is uniquely defined by the following:

1. ϕ is a continuous function, with $\phi(r) = 0$ for $|r| \geq 2$;
2. For all r , $\sum_{j \text{ even}} \phi(r-j) = \sum_{j \text{ odd}} \phi(r-j) = \frac{1}{2}$, $\sum_j (r-j) \phi(r-j) = 0$, and $\sum_j \phi^2(r-j) = C$.

In general, for $0 < r < 1$, the discretized delta function $\phi(r-j)$ covers four non-zero support points. However, for the degenerate case of the 4-point discretized delta function centered at $r = 0$, we have five support points, namely, $r - j = -2, -1, 0, 1, 2$. From the degenerate case, we can easily derive the constant C . Hence, we obtain the following four admissible branches of solutions for $0 < r < 1$,

$$\phi(r-2) = \frac{1}{2} - \phi(r), \quad \phi(r-1) = -\frac{1}{4} + \frac{r}{2} + \phi(r), \quad \phi(r+1) = \frac{3}{4} - \frac{r}{2} - \phi(r). \quad (6.59)$$

with $\phi(r) = \frac{-(2r-3) + \sqrt{4r-4r^2+1}}{8}$.

Therefore, in the three-dimensional case, one of the smoothed approximations to the delta function is given by

$$\delta_h(\mathbf{x}) = \frac{1}{h^3} \phi\left(\frac{x_1}{h}\right) \phi\left(\frac{x_2}{h}\right) \phi\left(\frac{x_3}{h}\right). \quad (6.60)$$

It is interesting to point out that the discretized delta function in Eq. (6.59) is very close to $(1 + \cos \pi r / 2) / 4$, with $r \in [-2, 2]$. Moreover, it is easy to confirm that the discretized delta function $\phi(r)$, with $r \in [-2, 2]$, defined in Eq. (6.59), has C^1 continuity [37].

For non-uniform meshing in the fluid domain, different kernel functions must be introduced. As illustrated in Ref. [44], both wavelet and smooth particle hydrodynamics (SPH) methods belong to a class of reproducing kernel methods where the “reproduced” function $u^R(x)$ is derived as

$$u^R(x) = \int_{-\infty}^{+\infty} u(y) \phi(x-y) dy, \quad (6.61)$$

with a projection operator or a window function $\phi(x)$. The reproducing condition requires that up to n^{th} order polynomial can be reproduced,

$$x^n = \int_{-\infty}^{+\infty} y^n \phi(x-y) dy. \quad (6.62)$$

From the convolution theorem, in the Fourier transform domain, Eq. (6.61) can be expressed as

$$\hat{u}^R(\omega) = \hat{u}(\omega) \hat{\phi}(\omega), \quad (6.63)$$

where ω is the natural frequency or wave number of the functions $u(x)$ and $\phi(x)$.

Suppose $\hat{\phi}(\omega)$ is a perfect rectangular window function, a so-called ideal low-pass filter function, which corresponds to the *sinc* function in the function domain, spatial or temporal, $\hat{u}(\omega)$ contains signals within $|\omega| \leq \omega_c$, where ω_c is the cut-off or band-limited frequency or wave number defined by the rectangular window function. Nevertheless, it has been recognized that the ideal low-pass filter does not have a compact support in the function domain and hence is not useful as interpolation functions in computational mechanics. Therefore, as presented in Ref. [44], a correction function $C(x; x-y)$ is introduced in the finite computation domain Ω , i.e., the domain of influence or support,

$$C(x; x-y) = \sum_{k=0}^n \beta_k(x)(x-y)^k, \quad (6.64)$$

and the consequent modified window function $\bar{\phi}$ is constructed as

$$\bar{\phi}(x; x-y) = \sum_{k=0}^n \beta_k(x)(x-y)^k \phi(x-y). \quad (6.65)$$

Define $m_k(x)$ as the k^{th} moment of the window function $\phi(x)$, with $i = \sqrt{-1}$, and we have

$$m_k(x) = \int_{\Omega} (x-y)^k \phi(x-y) dy = i^k \hat{\phi}^{(k)}(0), k = 0, 1, 2, \dots, n. \quad (6.66)$$

Similarly, define $\bar{m}_k(x)$ as the k^{th} moment of $\bar{\phi}(x)$, and we have

$$\bar{m}_k(x) = \int_{\Omega} (x-y)^k \bar{\phi}(x-y) dy = i^k \hat{\bar{\phi}}^{(k)}(0), k = 0, 1, 2, \dots, n. \quad (6.67)$$

In general, with a proper construction of the correction function with respect to the selected window functions such as the scaling function, wavelet, or spline family, we will be able to enforce

$$\hat{\bar{\phi}}(0) = 1, \hat{\bar{\phi}}^{(1)}(0) = 0, \dots, \hat{\bar{\phi}}^{(n)}(0) = 0. \quad (6.68)$$

In essence, the window function is required to be flatter at $\omega = 0$ in the Fourier domain as the order n of reproducing gets higher in Eq. (6.62). This implies that as $n \rightarrow \infty$, $\hat{\bar{\phi}}(\omega)$ becomes flatter at $\omega = 0$ and approaches to an ideal filter. Therefore, based on Eqs. (6.66), (6.67), and (6.68), we can set up the moment equations to solve for β_k ,

$$\begin{bmatrix} m_0(x) & m_1(x) & \cdots & m_n(x) \\ m_1(x) & m_2(x) & \cdots & m_{n+1}(x) \\ \vdots & \vdots & \ddots & \vdots \\ m_n(x) & m_{n+1}(x) & \cdots & m_{2n}(x) \end{bmatrix} \begin{Bmatrix} \beta_0(x) \\ \beta_1(x) \\ \vdots \\ \beta_n(x) \end{Bmatrix} = \begin{Bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{Bmatrix}. \quad (6.69)$$

Furthermore, we can also define a dilation parameter a and introduce

$$\bar{\phi}_a(x-y) = \frac{1}{a} \bar{\phi}\left(\frac{x-y}{a}\right). \quad (6.70)$$

Thus, by changing the value of the dilation parameter a , usually by a factor of 2, a sequence of low-pass filter is defined. Note that the dilation parameter a is directly linked to the mesh density.

The difference between two such low-pass filters defines a high-pass filter. Therefore, we can perform a multi-resolution analysis with the projection operator P_a expressed as

$$P_a u(x) = \int_{\Omega} u(y) \bar{\phi}_a(x - y) dy. \quad (6.71)$$

Based on the resolution of the projection operator $P_{a/2^n}$, a hierarchical representation of a function $u(x)$ is defined as

$$u(x) = \lim_{n \rightarrow \infty} P_{a/2^n} u(x), \dots, P_a u(x), \dots, \lim_{n \rightarrow \infty} P_{2^n a} u(x) = \emptyset.$$

Furthermore, we also define a complementary projection operator Q_{2a} , such that the higher scale projected solution $P_a u$ can be represented by the following sum of the low and high scale projections

$$P_a u(x) = P_{2a} u(x) + Q_{2a} u(x), \quad (6.72)$$

where the Q_{2a} projection, which can be viewed as a *peeled off* scale or the *rate of variation* of $P_a u(x)$, is simply given as a wavelet projection,

$$Q_{2a} u(x) = \int_{\Omega} u(y) \bar{\psi}_{2a}(x - y) dy, \quad (6.73)$$

with $\bar{\psi}_{2a}(x - y) = \bar{\phi}_a(x - y) - \bar{\phi}_{2a}(x - y)$.

In accordance with the reproducing property, we have

$$\int_{\Omega} y^k \bar{\psi}_{2a}(x - y) dy = 0, \text{ with } k = 0, 1, \dots, n, \quad (6.74)$$

where n can be considered as the order of the polynomial based wavelet. The discretized reconstruction of the delta function for non-uniform spacing can be written as

$$\bar{\phi}_{aJ}(x) = C(x; x - x_J) a^{-1} \phi\left(\frac{x - x_J}{a}\right) \Delta x_J. \quad (6.75)$$

For uniform spacing, if the basis functions are chosen to be $1, x, x^2$, and the cubic spline as the window function, the modified window function satisfying the moment reproducing conditions is given as,

$$\begin{aligned} \delta_h(r) &= \bar{\phi}_h(x) = \frac{1}{h} \bar{\phi}(r) \\ &= \frac{1}{h} \left(\frac{27}{17} - \frac{30}{17} r^2 \right) \begin{cases} \frac{1}{6} (r + 2)^3, & -2 \leq r < -1, \\ \frac{2}{3} - r^2 \left(1 + \frac{r}{2}\right), & -1 \leq r < 0, \\ \frac{2}{3} - r^2 \left(1 - \frac{r}{2}\right), & 0 \leq r < 1, \\ -\frac{1}{6} (r - 2)^3, & 1 \leq r < 2, \\ 0, & \text{otherwise.} \end{cases} \end{aligned} \quad (6.76)$$

Likewise, to reproduce $1, x, x^2, x^3, x^4$ terms, the following correction function can be introduced,

$$\frac{170010}{80347} - \frac{429450}{80347}r^2 + \frac{178290}{80347}r^4. \quad (6.77)$$

In Figs. 7 and 8, we provide a comparison in the function and Fourier domains of the modified window functions with the original discretized delta function. It is clear, as we increase the reproducing order n , with the same finite support region, the kernel functions approach more to the ideal low pass filter. In the immersed continuum method, we employ the RKPM kernel functions. It is interesting to note that it is also possible to interpolate the solid point velocity within a single finite element [3]. The mathematical implication with respect to the convergence behaviors and accuracy is still to be discovered. In addition, as shown in Fig. 9 the translation invariance is no longer satisfied for RKPM kernel functions. Numerical evidence also suggests that such a restriction based on the translation invariance could be relaxed.

7 Matrix-Free Newton-Krylov

In the k^{th} Newton-Raphson iteration at time step $m+1$ of the nonlinear residual equation (5.54), from \mathcal{R}^N to \mathcal{R}^N , with N as the number of the total unknowns, we start with a first guess of the incremental unknowns $\Delta\Theta^{k,0}$, namely, $\Delta\mathbf{V}^0$, $\Delta\mathbf{P}^0$, and $\Delta\mathbf{P}^{s,0}$, which often are zero vectors. Then the residual of the linearized systems of equations at the k^{th} Newton-Raphson iteration is evaluated as

$$\mathbf{p} = -\mathbf{r}_{,v}^{m+1,k-1} - \mathbf{r}_{,p}^{m+1,k-1}\Delta\mathbf{V}^0 - \mathbf{r}_{,p}^{m+1,k-1}\Delta\mathbf{P}^0 - \mathbf{r}_{,p^s}^{m+1,k-1}\Delta\mathbf{P}^{s,0}. \quad (7.78)$$

This error vector is used to construct the n -dimensional Krylov subspace $\mathcal{K}^n = \text{span}\{\mathbf{p}, \mathbf{J}\mathbf{p}, \mathbf{J}^2\mathbf{p}, \dots, \mathbf{J}^{n-1}\mathbf{p}\}$, where \mathbf{J} is the $N \times N$ Jacobian matrix evaluated at time step $m+1$ and the k^{th} Newton-Raphson iteration of the nonlinear residual equation (5.54) and can be rewritten as

$$\mathbf{J} = (\mathbf{r}_{,v}^{m+1,k-1}, \mathbf{r}_{,p}^{m+1,k-1}, \mathbf{r}_{,p^s}^{m+1,k-1}). \quad (7.79)$$

The approximate solution $\Delta\Theta$ is written as the combination of the initial guess $\Delta\Theta^{k,0}$ and \mathbf{z}^n , with $\mathbf{z}^n \in \mathcal{K}^n$. Note that the dimension of the subspace \mathcal{K}^n is n which is much smaller than the dimension N of the unknown vector $\Delta\Theta$. The N -dimensional unknown vector $\Delta\Theta$ or rather \mathbf{z}^n is represented with $\mathbf{V}_n \mathbf{y}$, where \mathbf{y} is a much smaller n -dimensional unknown vector. In the Generalized Minimum Residual (GMRES) method, the modified Gram-Schmidt orthogonalization procedures are used to derive a set of orthonormal vectors v_i , with $1 \leq i \leq n$ in the Krylov space \mathcal{K}^n and an $(n+1) \times n$ upper-Hessenberg matrix $\bar{\mathbf{H}}_n$. Define $\mathbf{V}_n = (\mathbf{v}_1 \mathbf{v}_2 \dots \mathbf{v}_n)$ and $\mathbf{V}_{n+1} = (\mathbf{v}_1 \mathbf{v}_2 \dots \mathbf{v}_{n+1})$, we have the following

$$\mathbf{J}\mathbf{V}_n = \mathbf{V}_{n+1}\bar{\mathbf{H}}_n. \quad (7.80)$$

The remaining process in the GMRES method is to solve the least square problem

$$\min_{\mathbf{z} \in \mathcal{K}^n} \|\mathbf{p} - \mathbf{J}\mathbf{z}\| \text{ or } \min_{\mathbf{y} \in \mathcal{R}^n} \|\mathbf{p} - \mathbf{J}\mathbf{V}_n \mathbf{y}\|. \quad (7.81)$$

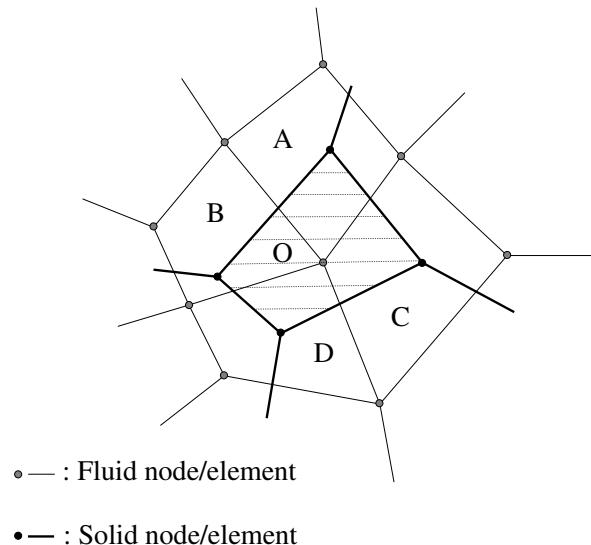
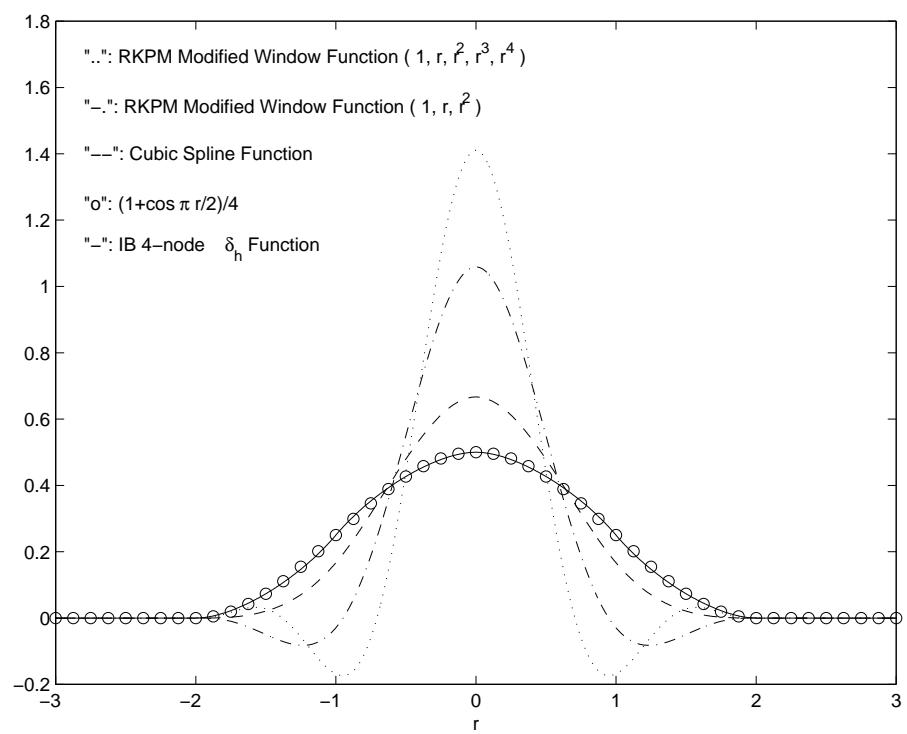


Figure 6: Communication between Lagrangian solid nodes and Eulerian fluid nodes.



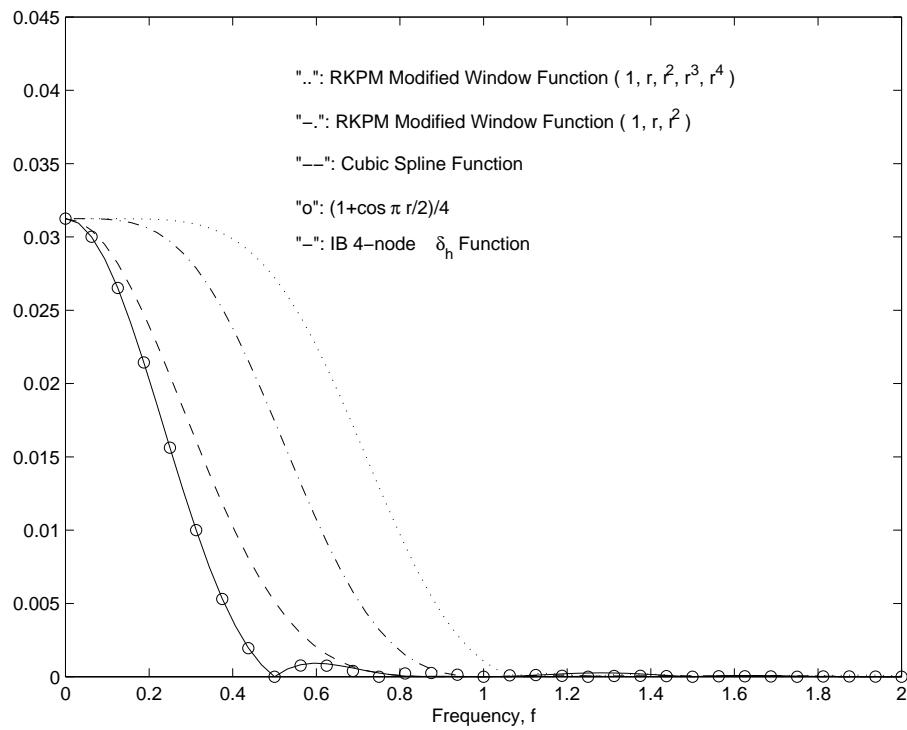


Figure 8: A comparison of the spectra of various discretized delta functions.

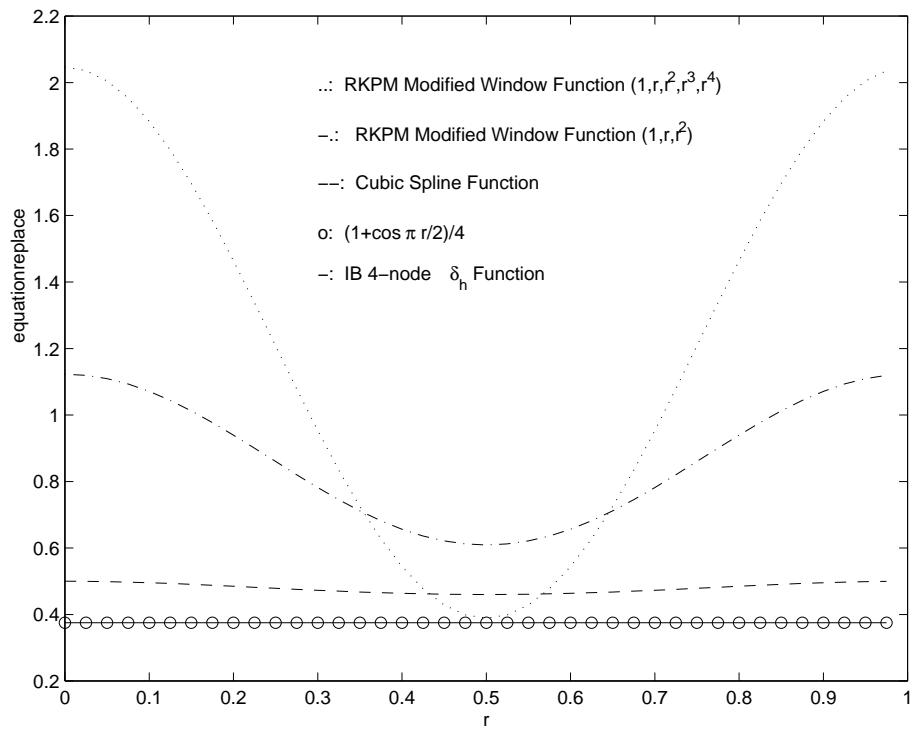


Figure 9: A comparison of various discretized delta functions with respect to the translation invariance.

Assume γ is the length of the initial residual vector \mathbf{p} and \mathbf{e}_1 is the unit vector representing the first column of $(n+1) \times (n+1)$ identity matrix, substituting Eq. (7.80), we can show that Eq. (7.81) is equivalent to the following minimization within a much smaller space

$$\min_{\mathbf{y} \in \mathcal{R}^n} \|\gamma \mathbf{e}_1 - \bar{\mathbf{H}}_n \mathbf{y}\|, \quad (7.82)$$

In the matrix-free Newton-Krylov iteration, we do not form the Jacobian matrix. In general, this Jacobian matrix in the immersed boundary/continuum methods has an $O(n^2)$ storage requirement. For large systems with million degrees of freedoms, this Jacobian matrix requires a terabyte (10^{12}) memory which is beyond the limit of computational facilitates available for most scientific researches. It is based on this understanding, we would also like to design a preconditioning technique without the use of the Jacobian matrix [7] [14] [23].

First of all, the initial residual vector \mathbf{p} in the k^{th} Newton-Raphson iteration at time step $m + 1$ of the nonlinear residual equation (5.54) is normalized as \mathbf{v}_1 with the length $\gamma = \|\mathbf{p}\|_2$. Using Eq. (7.82), we have the corresponding n dimensional residual vector $\mathbf{b} = \gamma \mathbf{e}_1$. Introduce a preconditioning matrix Λ , for $i = 1$ to n , using the modified Gram-Schmidt orthogonalization process, we have $\mathbf{q}_i = \Lambda^{-1} \mathbf{v}_i$ and $\mathbf{w} = \mathbf{J} \mathbf{q}_i$, and for $j = 1$ to i , we have $h_{ji} = \mathbf{w}^T \mathbf{v}_j$ and \mathbf{w} is updated with $\mathbf{w} - h_{ji} \mathbf{v}_j$. As a consequence, we obtain $h_{(i+1)i} = \|\mathbf{w}\|_2$ and $\mathbf{v}_{i+1} = \mathbf{w}/h_{(i+1)i}$.

An important procedure in the matrix-free Newton-Krylov is to replace $\mathbf{w} = \mathbf{J} \mathbf{q}_i$ with a finite difference based calculation,

$$\mathbf{J} \mathbf{q}_i \simeq \frac{\mathbf{r}(\Theta^{m+1,k-1} + e \mathbf{q}_i) - \mathbf{r}(\Theta^{m+1,k-1})}{e}, \quad (7.83)$$

where e is often set to be around the square root of the machine error [7].

After we establish the elements of an upper $n \times n$ Hessenberg matrix \mathbf{H}_n as well as an upper $(n+1) \times n$ Hessenberg matrix $\bar{\mathbf{H}}_n$, for $j = 1$ to n , and $i = 1$ to $j-1$, a factorization of \mathbf{H}_n is carried out through the following rotation matrix operations,

$$\begin{aligned} h_{ij} &= c_i h_{ij} + s_i h_{(i+1)j}, \\ h_{(i+1)j} &= -s_i h_{ij} + c_i h_{(i+1)j}, \end{aligned} \quad (7.84)$$

where the entities of the rotation processes are calculated as

$$r = \sqrt{h_{jj}^2 + h_{(j+1)j}^2}, \quad c_j = h_{jj}/r, \text{ and } s_j = h_{(j+1)j}/r. \quad (7.85)$$

Through this rotation process, the upper Hessenberg matrix is converted to a diagonal matrix with the coefficients defined as: for $j = 1$ to n

$$h_{jj} = r, \quad p_j = c_j b_j, \text{ and } p_{j+1} = -s_j b_j. \quad (7.86)$$

Finally, the termination criteria of the GMRES iteration will rest at the absolute value of b_{n+1} in comparison with a given error ϵ . If $|b_{n+1}| < \epsilon$, the solution vector $\Delta\Theta$, or rather $\Delta\mathbf{V}$, $\Delta\mathbf{P}$, and $\Delta\mathbf{P}^s$ is expressed as

$$\Delta\Theta^{k,n} = \Delta\Theta^{k,0} + \sum_{i=1}^n y_i \mathbf{q}^i, \text{ or } \begin{bmatrix} \Delta\mathbf{V} \\ \Delta\mathbf{P} \\ \Delta\mathbf{P}^s \end{bmatrix} = \Delta\Theta^{k,0} + \sum_{i=1}^n y_i \mathbf{q}^i. \quad (7.87)$$

As a final remark, if the initial guess $\Delta\Theta^{k,0}$ does not produce a good estimate within a sufficiently small Krylov subspace \mathcal{K}^n . $\Delta\Theta^{k,n}$ will be introduced as an updated initial guess and the GMRES iteration procedure will continue until a solution with the desired accuracy is obtained.

8 Numerical Examples

Recently, some breakthroughs have been made in the development of finite element formulations for the immersed boundary/continuum methods [41] [49]. In addition, a preliminary formulation for beams with both bending and torsional moments has also been developed [12] [21]. Many results have been documented in Refs. [22] [38] [39] [46]. In this paper, we focus on a series of driven cavity problems with immersed solids. For the case with one immersed square solid, it is still possible to solve this fluid-solid system with traditional modeling techniques such as the arbitrary Lagrangian-Eulerian (ALE) formulation. We will compare the solution derived from the immersed continuum method with those from the traditional approaches.

The driven cavity problem considered here has the dimension of 1×1 m. A typical immersed solid has the dimension of 0.13×0.13 m. In the immersed continuum method, the background fluid mesh for the entire cavity which includes the space occupied by the immersed solids consists of 20×20 $9/4c$ mixed finite elements. A typical immersed solid is represented by 4×4 $9/4c$ mixed finite elements. In the plotting stage, we split each 9-node element into four 4-node element. As a result, the velocity mesh will be two times denser than the pressure mesh. As shown in Fig. 10, the construction of this mesh demonstrates the philosophical difference between the immersed boundary/continuum methods and the traditional ALE type of modeling approaches. In the immersed continuum method, the solid mesh is right on top of the background fluid mesh, whereas in the ALE formulation, the solid mesh is surrounded by the fluid mesh with a different mesh density.

To compare the dynamical behaviors, the top shear velocity of the cavity is set as $0.1 \sin(2\pi/40t)$ m/s. The deformable solid is situated initially in the center of the cavity with zero velocity. The submerged solid is made of a compressible rubber material with the material constants $C_1 = 2000$, $C_2 = 1000$, and $\kappa_s = 1 \times 10^8$ Pa and the density $\rho_s = 1000$ kg/m³. In addition, the viscous fluid is also represented with compressible model with a constant wave speed. In this case, we have the dynamic viscosity $\mu = 1$ Pa · s, the bulk modulus $\kappa_f = 2.1 \times 10^7$ Pa, and the density $\rho_f = 1000$ kg/m³.

Even with the coarse mesh used, the developed formulation with high-order elements provides reasonable results comparable to a reference solution. In addition, no spatial oscillation and checkerboard pressure bands are observed, as demonstrated in Fig. 11. Of course, the proposed description can be used for refined meshes. However, as one of the main messages, we must point out that in engineering practice, before a large number of finite elements are used, it is always beneficial to employ coarse meshes with high-order elements to obtain a reasonable estimation of complicated problems.

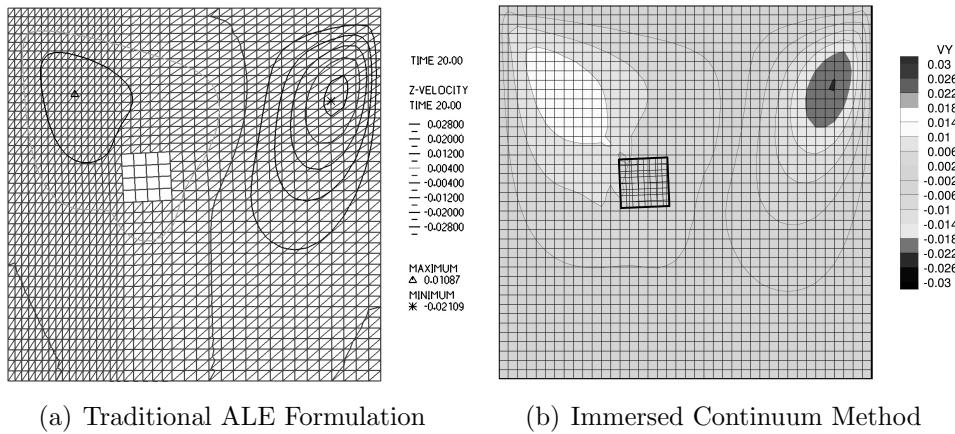


Figure 10: Horizontal velocity comparison between the traditional ALE formulation and the immersed continuum method.

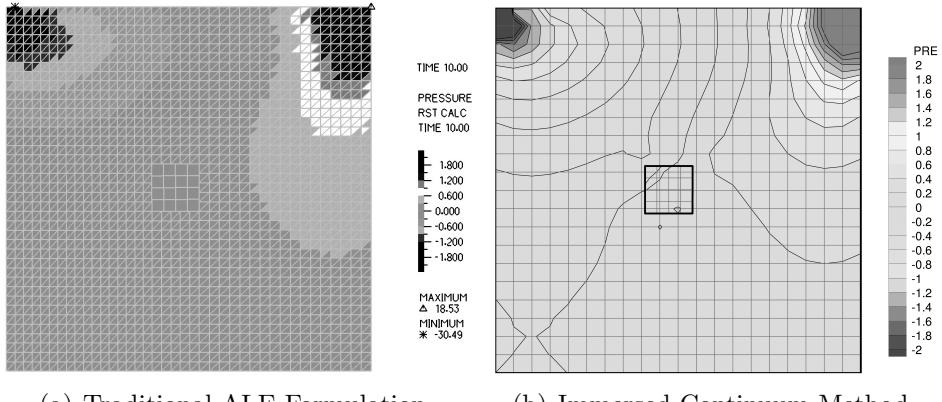


Figure 11: Pressure comparison between the traditional ALE formulation and the immersed continuum method

The benefit of the immersed methods is clearly depicted in Figs. 14 and 13 for the case with five immersed solids. For this case, it is no longer feasible to use the ALE formulation, whereas it is a simple task to add a few more deformable solids in the immersed continuum method. The pressure and the vertical velocity distributions at different time stations calculated using the immersed continuum method are depicted in Figs. 13 and 14. Notice that in the immersed continuum method the contact between two immersed solids ($T = 35$ and 38.5 sec) is accomplished for free just as in the material point method and the immersed boundary method.

The immersed continuum method was implemented in a regular laptop, the documented running time for the implicit matrix-free Newton-Krylov iterative procedure is as follows:

Although this implicit matrix-free Newton-Krylov solution scheme is not fast, the procedure is not extremely time consuming or prohibitive to carry out for a typical laptop. By choosing the proper preconditioner, the inner GMRES iteration can be minimized to two to three steps while the

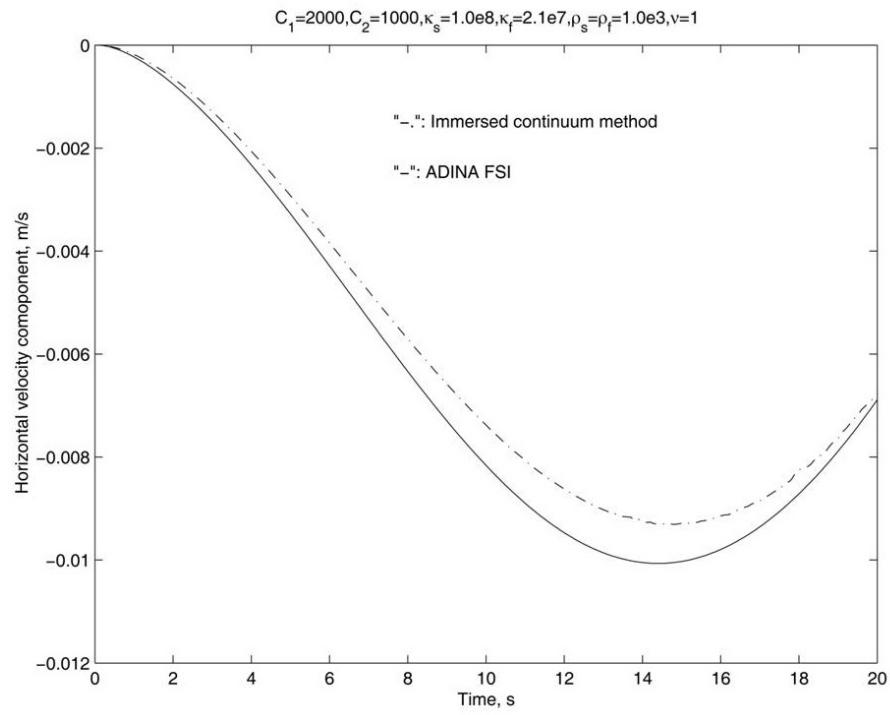


Figure 12: A comparison of horizontal velocity derived from the traditional ALE formulation and the immersed continuum method.

Step Size (s)	Number of Steps	Running Time (s)
0.5	100	26163
1	50	13452
2	25	7105

Table 1: Comparison of running times on a typical laptop with the implicit procedures.

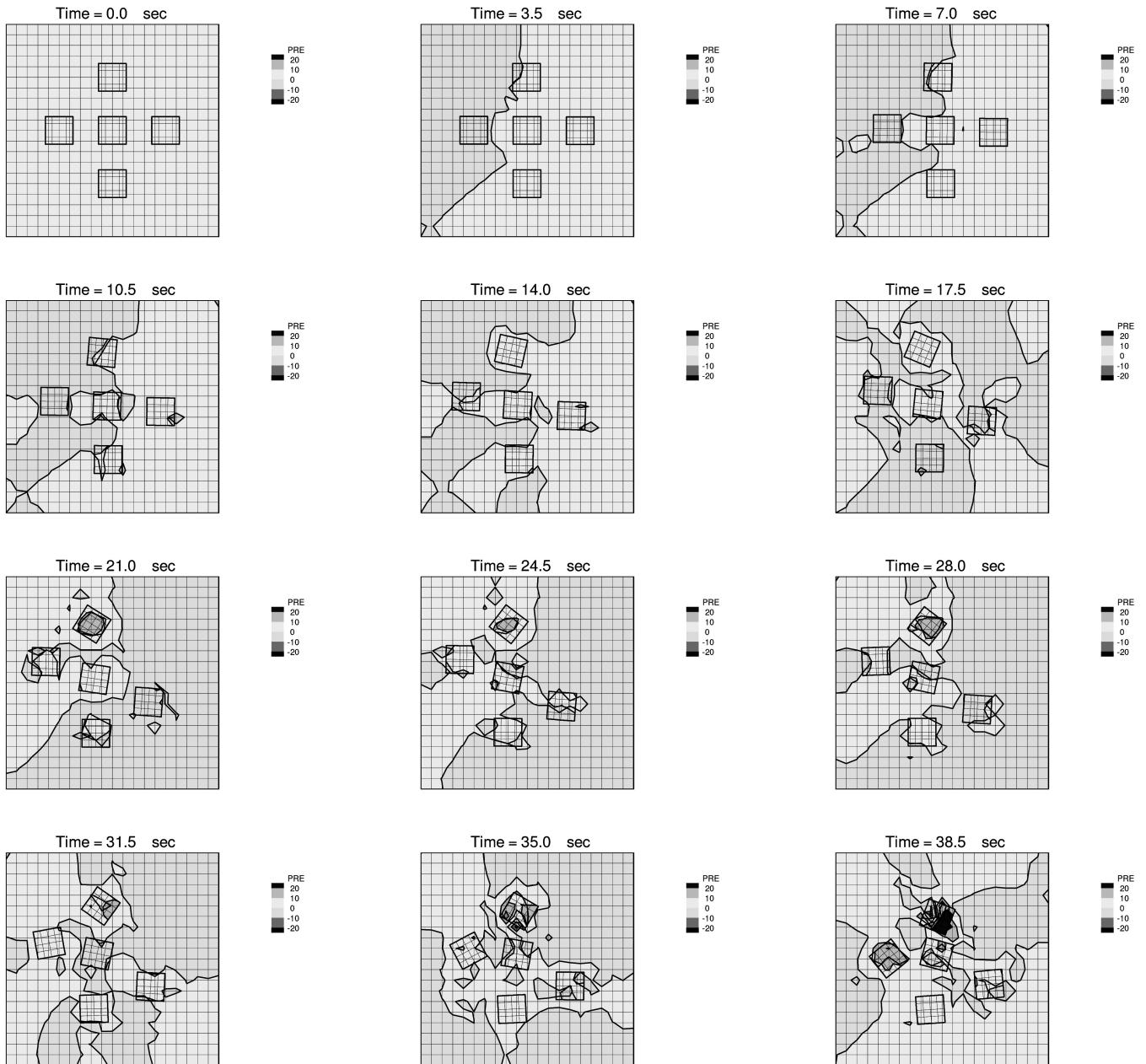


Figure 13: A comparison of horizontal velocity derived from the traditional ALE formulation and the immersed continuum method.

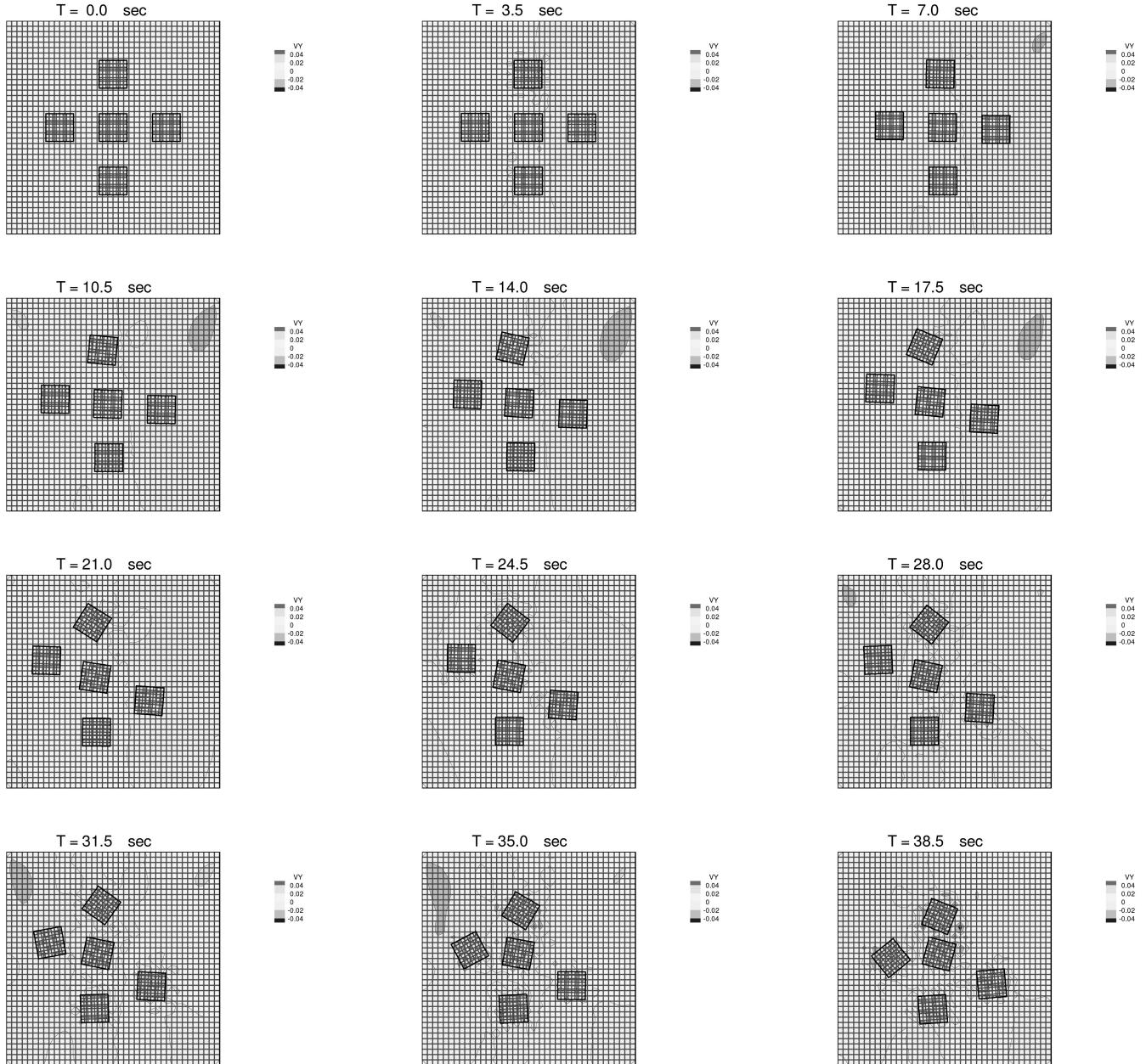


Figure 14: A comparison of horizontal velocity derived from the traditional ALE formulation and the immersed continuum method.

outer Newton iteration still depends on how close to the solution the initial guess is.

9 Conclusions

In the immersed boundary/continuum methods, in order to satisfy energy conservation, namely, that the energy input to the fluid domain from the elastic boundary/solid is the same as that from the equivalent body force, the same Dirac delta function must be used in both the distribution of the resultant nodal force and the interpolation of the velocity based on the surrounding fluid grid point velocities. The key treatment in the immersed boundary/continuum methods is to synchronize the fluid motion with the solid motion within the immersed solid domain Ω_s , namely,

$$\mathbf{v}^s = \mathbf{v}^f. \quad (9.88)$$

In fact, the constraint of Eq. (9.88) introduces the (distributed) Lagrange multiplier as the equivalent body force. It is conceivable that full-fledged augmented Lagrange multipliers can be introduced as primary unknowns. In this case, the equivalent body forces can be directly calculated along with independent fluid and solid velocity vectors. Nevertheless, such a procedure will introduce a set of new unknowns equal to the number of velocity unknowns for solids. We conclude that mixed finite element formulations namely, velocity/pressure for fluids and displacement/pressure for solids can be successfully adopted to the immersed continuum method. Of course, it is only the beginning of this type of modeling treatment for complex FSI systems. More mathematical understandings and studies are urgently needed.

10 Acknowledgment

The supports from NSF (CMMI-0734357, CMMI-0503652, and CBET-0503649) are gratefully acknowledged.

References

- [1] A.L. Fogelson and C.S. Peskin. A fast numerical method for solving the three-dimensional stokes equations in the presence of suspended particles. *Journal of Computational Physics*, 79:50–69, 1988.
- [2] W. Bao, X. Wang, and K.J. Bathe. On the Inf-Sup condition of mixed finite element formulations for acoustic fluids. *Mathematical Models & Methods in Applied Sciences*, 11:883–901, 2001.
- [3] D. Boffi and L. Gastaldi and L. Heltai. On the CFL condition for the finite element immersed boundary method. *Computers & Structures*, 85:775–783, 2007.
- [4] D. Boffi and L. Gastaldi. A finite element approach for the immersed boundary method. *Computers & Structures*, 81:491–501, 2003.

- [5] C.S. Peskin. Numerical analysis of blood flow in the heart. *Journal of Computational Physics*, 25:220–252, 1977.
- [6] C.S. Peskin. The immersed boundary method. *Acta Numerica*, 11:479–517, 2002.
- [7] D.A. Knoll and D.E. Keyes. Jacobian-free Newton-Krylov methods: a survey of approaches and applications. *Journal of Computational Physics*, 193:357–397, 2004.
- [8] R. Dillon, L. Fauci, A. Fogelson, and D. Gaver III. Modeling biofilm processes using the immersed boundary method. *Journal of Computational Physics*, 129:57–73, 1996.
- [9] D.M. McQueen and C.S. Peskin. Computer-assisted design of butterfly bileaflet valves for the mitral position. *Scand. J. Thor. Cardiovasc. Surg.*, 19:139–148, 1985.
- [10] J. Donea, S. Giuliani, and J.P. Halleux. An arbitrary Lagrangian-Eulerian finite element method for transient dynamic fluid-structure interactions. *Computer Methods in Applied Mechanics and Engineering*, 33:689–723, 1982.
- [11] L.J. Fauci and C.S. Peskin. A computational model of aquatic animal locomotion. *Journal of Computational Physics*, 77:85–108, 1988.
- [12] A. Ferent, C.S. Peskin, and X. Wang. Modeling a bent twisted filament immersed in fluid. *7th US National Congress on Computational Mechanics*, 2003.
- [13] R. Glowinski, T.W. Pan, T.I. Hesla, D.D. Joseph, and J. Périault. A fictitious domain approach to the direct numerical simulation of incompressible viscous flow past moving rigid bodies: Application to particulate flow. *Journal of Computational Physics*, 169:363–426, 2001.
- [14] D. Kay, D. Loghin, and A. Wathen. A preconditioner for the steady-state Navier-Stokes equations. *SIAM J. Sci. Comput.*, 24(1):237–256, 2002.
- [15] Y. Kim, L. Zhu, X. Wang, and C.S. Peskin. On various techniques for computer simulation of boundaries with mass. *2nd MIT Conference on Computational Fluid and Solid Mechanics*, 2003.
- [16] K.J. Bathe. *Finite Element Procedures*. Prentice Hall, Englewood Cliffs, N.J., 1996.
- [17] K.J. Bathe. Fluid-structure interactions - The fully coupled solution of fluid flows with structural interactions, a rapidly evolving discipline, represents the natural next step in simulating mechanical systems. *Journal of Mechanical Engineering*, pages 66–68, April 1998.
- [18] K.J. Bathe and V. Sonnad. On effective implicit time integration in analysis of fluid-structure problems. *International Journal for Numerical Methods in Engineering*, 15:943–948, 1980.
- [19] K.J. Bathe, H. Zhang, and M.H. Wang. Finite element analysis of incompressible and compressible fluid flows with free surfaces and structural interactions. *Computers & Structures*, 56(2/3):193–213, 1995.

- [20] S. Li and W.K. Liu. Reproducing kernel hierarchical partition of unity part I: Formulation and theory. *International Journal for Numerical Methods in Engineering*, 45:251–288, 1999.
- [21] S. Lim, A. Ferent, X. Wang, and C.S. Peskin. Dynamics of a closed rod with twist and bend in fluid. *SIAM Journal on Scientific Computing*, 2007.
- [22] Y. Liu, L. Zhang, X. Wang, and W.K. Liu. Coupling of Navier-Stokes equations with protein molecular dynamics and its application to hemodynamics. *International Journal for Numerical Methods in Fluids*, 46:1237–1252, 2004.
- [23] M.D. Tidriri. Preconditioning techniques for the Newton-Krylov solution of compressible flows. *Journal of Computational Physics*, 132:51–61, 1997.
- [24] H. Morand and R. Ohayon. *Fluid Structure Interaction - Applied Numerical Methods*. John Wiley & Sons, 1995. Translated by C.A. James.
- [25] Jr. R.P. Beyer. A computational model of the cochlea using the immersed boundary method. *Journal of Computational Physics*, 98:145–162, 1992.
- [26] J.M. Stockie and B.R. Wetton. Analysis of stiffness in the immersed boundary method and implications for time-stepping schemes. *Journal of Computational Physics*, 154:41–64, 1999.
- [27] D. Sulsky and J.U. Brackbill. A numerical method for suspension flow. *Journal of Computational Physics*, 96:339–368, 1991.
- [28] T. Sussman and K.J. Bathe. A finite element formulation for nonlinear incompressible elastic and inelastic analysis. *Computers & Structures*, 26(1/2):357–409, 1987.
- [29] T.E. Tezduyar. Stabilized finite element formulations for incompressible-flow computations. *Advances in Applied Mechanics*, 28:1–44, 1992.
- [30] T.E. Tezduyar. Finite element methods for flow problems with moving boundaries and interfaces. *Archives of Computational Methods in Engineering*, 8:83–130, 2001.
- [31] T.J.R. Hughes. *The finite element method: Linear static and dynamic analysis*. Prentice Hall, 1987.
- [32] T.J.R. Hughes, L.P. Franca, and M. Balestra. A new finite element formulation for computational fluid dynamics: V. Circumventing the Babuška-Brezzi condition: A stable Petrov-Galerkin formulation of the Stokes problem accommodating equal-order interpolations. *Computer Methods in Applied Mechanics and Engineering*, 59:85–99, 1986.
- [33] R. Van Loon, P.D. Anderson, F.N. Van de Vosse, and S.J. Sherwin. Comparison of various fluid-structure interaction methods for deformable bodies. *Computers & Structures*, 85:833–843, 2006.
- [34] X. Wang. Finite element analysis of air-sheet interactions and flutter suppression devices. *Computers & Structures*, 64(5/6):983–994, 1997.

- [35] X. Wang. Numerical analysis of moving orthotropic thin plates. *Computers & Structures*, 70:467–486, 1999.
- [36] X. Wang. Velocity/Pressure mixed finite element and finite volume formulation with ALE descriptions for nonlinear fluid-structure interaction problems. *Advances in Engineering Software*, 31:35–44, 2000.
- [37] X. Wang. On the discretized delta function and force calculation in the immersed boundary method. In K.J. Bathe, editor, *Computational Fluid and Solid Mechanics*, pages 2164–2169. Elsevier, 2003.
- [38] X. Wang. From immersed boundary method to immersed continuum method. *International Journal for Multiscale Computational Engineering*, 4:127–145, 2006.
- [39] X. Wang. An iterative matrix-free method in implicit immersed boundary/continuum methods. *Computers & Structures*, 85:739–748, 2007.
- [40] X. Wang and K.J. Bathe. Displacement/pressure based finite element formulations for acoustic fluid-structure interaction problems. *International Journal for Numerical Methods in Engineering*, 40:2001–2017, 1997.
- [41] X. Wang and W.K. Liu. Extended immersed boundary method using FEM and RKPM. *Computer Methods in Applied Mechanics and Engineering*, 193:1305–1321, 2004.
- [42] K. Washizu. *Variational Methods in Elasticity and Plasticity*. Pergamon, 1975.
- [43] W.K. Liu, Y. Chen, C.T. Chang, and T.B. Belytschko. Advances in multiple scale kernel particle methods. *Computational Mechanics*, 18:73–111, 1996.
- [44] W.K. Liu, Y. Chen, R.A. Uras, and C.T. Chang. Generalized multiple scale reproducing kernel particle methods. *Computer Methods in Applied Mechanics and Engineering*, 139:91–157, 1996.
- [45] W.K. Liu, S. Jun, and Y.F. Zhang. Reproducing kernel particle methods. *International Journal for Numerical Methods in Fluids*, 20:1081–1106, 1995.
- [46] W.K. Liu, Y. Liu, D. Farrell, L. Zhang, X. Wang, Y. Fukui, N. Patankar, Y. Zhang, C. Bajaj, X. Chen, and H. Hsu. Immersed finite element method and its applications to biological systems. *Computer Methods in Applied Mechanics and Engineering*, 195:1722–1749, 2006.
- [47] W.K. Liu and Y.J. Chen. Wavelet and multiple scale reproducing kernel methods. *International Journal for Numerical Methods in Fluids*, 21:901–932, 1995.
- [48] Z. Yu. A DLM/FD method for fluid/flexible-body interactions. *Journal of Computational Physics*, 207:1–27, 2005.
- [49] L. Zhang, A. Gerstenberger, X. Wang, and W.K. Liu. Immersed finite element method. *Computer Methods in Applied Mechanics and Engineering*, 193:2051–2067, 2004.