INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2008; **56**:2151–2177
Published online 23 August 2007 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld.1578

A hybrid immersed boundary and material point method for simulating 3D fluid–structure interaction problems

Anvar Gilmanov[1] and Sumanta Acharya[2,*,†]

[1]*Center for Computations and Technology, Louisiana State University, Baton Rouge, LA 70803, U.S.A.*
[2]*Mechanical Engineering, Louisiana State University, Baton Rouge, LA 70803, U.S.A.*

SUMMARY

A numerical method is developed for solving the 3D, unsteady, incompressible Navier–Stokes equations in curvilinear coordinates containing immersed boundaries (IBs) of arbitrary geometrical complexity moving and deforming under forces acting on the body. Since simulations of flow in complex geometries with deformable surfaces require special treatment, the present approach combines a hybrid immersed boundary method (HIBM) for handling complex moving boundaries and a material point method (MPM) for resolving structural stresses and movement. This combined HIBM & MPM approach is presented as an effective approach for solving fluid–structure interaction (FSI) problems. In the HIBM, a curvilinear grid is defined and the variable values at grid points adjacent to a boundary are forced or interpolated to satisfy the boundary conditions. The MPM is used for solving the equations of solid structure and communicates with the fluid through appropriate interface-boundary conditions.

The governing flow equations are discretized on a non-staggered grid layout using second-order accurate finite-difference formulas. The discrete equations are integrated in time *via* a second-order accurate dual time stepping, artificial compressibility scheme. Unstructured, triangular meshes are employed to discretize the complex surface of the IBs. The nodes of the surface mesh constitute a set of Lagrangian control points used for tracking the motion of the flexible body. The equations of the solid body are integrated in time *via* the MPM. At every instant in time, the influence of the body on the flow is accounted for by applying boundary conditions at stationary curvilinear grid nodes located in the exterior but in the immediate vicinity of the body by reconstructing the solution along the local normal to the body surface. The influence of the fluid on the body is defined through pressure and shear stresses acting on the surface of the body.

The HIBM & MPM approach is validated for FSI problems by solving for a falling rigid and flexible sphere in a fluid-filled channel. The behavior of a capsule in a shear flow was also examined. Agreement with the published results is excellent. Copyright © 2007 John Wiley & Sons, Ltd.

Received 13 March 2007; Revised 19 June 2007; Accepted 21 June 2007

KEY WORDS: artificial compressibility; dual time stepping; immersed boundaries; hybrid immersed boundary method; material point method; fluid–structure interaction

*Correspondence to: Sumanta Acharya, Mechanical Engineering, Louisiana State University, Baton Rouge, LA 70803, U.S.A.
†E-mail: acharya@me.lsu.edu

Contract/grant sponsor: State of Louisiana

1. INTRODUCTION

Flows involving *fluid–structure interaction* (FSI) represent a wide class of problems where it is necessary to simultaneously consider the coupled motions of the fluid and the solid–structure. FSI problems pose a great challenge since they involve complex moving boundaries and require the solution of the equations for nonlinear elasticity, nonlinear fluid mechanics and their coupling. In many applications, FSI involves complex geometries. Examples include inflating parachutes, flow through blood vessels, dynamics of heart valves, swimming of aquatic animals and many others.

The solution of FSI problem requires four key issues to be addressed: (1) solution of the governing flow equations; (2) solution of the equations of the deformable body; (3) tracking of the solid–fluid interface; and (4) interface-boundary conditions representing the mutual influence of the two environments. Since the solid–fluid interface can deform in space and time, the equations for both the fluid and the solid have to be solved in complex domains that deform in space and time. Therefore, computationally effective strategies for both the fluid and the solid have to be devised, and represent the primary motivation for the present paper.

1.1. Strategies for solving flow equations with complex and moving boundaries

In the published literature two approaches are typically used to represent the interface: (1) *a sharp interface* approach where the surface is tracked [1], and (2) *a diffused interface* method that utilizes surface capturing methods [2]. The notation of *diffused* and *sharp* interface is based on how well resolved is the interface: if the interface has a non-zero thickness, it is diffusive [1], and if the interface has negligible thickness, it is sharp.

Popular body-fitted methods for solving FSI problems are classified as *Lagrangian* [3] and *arbitrary Lagrangian–Eulerian* (ALE) methods [4, 5]. Body-fitted methods have the tremendous advantage that the boundary condition is addressed exactly at the boundary surfaces and resolved very accurately, because the moving boundary coincides at all times with one of the coordinate surfaces. In pure Lagrangian methods [3], both the fluid and the deformable body are described in a Lagrangian frame. For small and moderate displacements of the fluid such a description allows accurate solutions, since errors caused by the convective fluxes through surfaces of cells that move with the fluid are absent. A purely Lagrangian method was employed by Belytschko and Kennedy [6] and Donea *et al.* [7] to study hydro-structural problems. However, for significant movement of the fluid that is characteristic for most flow problems, without interpolation of the solution on a new grid, rotation or shearing of the fluid can lead to immediate entangling of the grid cells. The required interpolation increases the computational effort significantly.

ALE methods are less restrictive due to the moving grid that follows the deformable boundaries. A characteristic property inherent to body-fitted grid methods like ALE consists of the ability to adjust to a surface of a body and hence these methods [4, 5, 8] are better suited for carrying out high Reynolds number simulations with high order of accuracy. However, due to the need for the mesh to conform to the body at all times, they are inherently limited to problems with moderate body deformations to avoid excessive cell skewing.

Moving grid methods referred to as *deforming spatial domain/stabilized space–time* method have been studied by Tezduyar *et al.* [9] based on stabilized finite element (FE) method [10] for the fluid and deformable body. The moving mesh for fluid was updated automatically till it became too distorted, after which partial or full remeshing is carried out. The coupled, nonlinear equations are generated from the FE discretization of the governing equations for the fluid flow, and the

DOI: 10.1002/fld

structural deformation and the motion of the flow mesh [9]. The method was used to simulate the unsteady interaction of flow with a deformable parachute canopy.

The restrictions inherent to moving grid methods are absent in fixed-grid methods. However, these approaches also have certain disadvantages such as interface smearing, loss of accuracy, etc.

One approach to connect a Lagrangian body with the Eulerian solution of the fluid flow is the *cut-cell method* (CCM) [11–17]. The boundary is considered as a sharp interface and the grid cells adjacent to the interface are modified based on their intersections with the Eulerian grid. CCM allows a clear distinction between the solid and the fluid by practically generating a boundary-fitted grid around the body and are applicable to problems involving arbitrary deformations of a body (or an ensemble of bodies) relative to the fixed Eulerian grid. However, such approaches lead to a complex algorithm because of the large number of possible intersections between the fixed grid and the surface of the deformable body.

The *volume of fluid* (VOF) *method* [1, 18, 19] consists of reconstructing the interface in each cell containing a fractional fluid volume through a special surface function that is used to distinguish one fluid from another. This function F defines the shape and motion of the boundary and it is moved with the flow. The reconstruction could be performed with first-order accuracy [18], [20] or with higher accuracy [21]. One of the main features of VOF methods consist of preserving mass in a natural way.

The *level set method* was introduced by Osher and Sethian [22]. This technique relies on an implicit formulation of the interface whose zero-level set always gives the location of the propagating interface. These methods [23, 24] are attractive because they enable a convenient description of topologically complex interfaces and are quite simple to implement.

The *fictitious domain method* (FDM) was introduced by Saul'ev [25] and has been primarily applied to the interaction of fluid with rigid body particles by Glowinski *et al.* [26, 27]. The main idea of FDM consists of coupling the moving rigid particles with the fluid by using a Lagrangian multiplier. A new method combining the fictitious domain [26] and the mortar element [28] methods for the computational analysis of FSI of Newtonian flows with slender bodies was developed by Baaijens [29]. This method was extended to describe the interaction of a large leaflet motion with the surrounding fluid domain [30].

The *immersed boundary method* (IBM) was introduced by Peskin [31] to study the flow in a heart valve. The idea was useful in solving FSI problems with free movement of structure through a fluid domain. The interaction between fluid and deformable body was realized through nodal forces incorporated in the momentum equations. These external terms were spread over the computational domain through smoothed approximation of the Dirac delta function and they satisfy boundary conditions on the surface. Disadvantages include thickening of the interface which is inherent in all diffusive interface methods. IBM [31] is only first-order accurate in space and the boundary spreads over 3–5 grid nodes.

The *immersed interface method* (IIM) was proposed by LeVeque and Li [32] to further develop the IBM of Peskin [33]. Instead of using a smooth approximation of the delta function, the IIM used approximations of delta function with discontinuity across the boundary (jump conditions). The IIM is therefore similar to the sharp-interface method. In [34, 35] it was shown that the IIM has second-order accuracy and free from the shortcomings of IBM.

The *immersed FE method* was developed by Zhang *et al.* [36]. The equations for the fluid flow and the solid body are modeled with the FE method. To avoid expensive grid regeneration, a fixed Eulerian grid for the fluid was used. The connection between the Lagrangian solid body and the

DOI: 10.1002/fld

fluid was implemented as in the IBM, but instead of the Dirac delta function, the higher-order reproducing-kernel particle method [37] was used.

An approach different from the classical IBM [31] approach that does not require the explicit addition of discrete forces to the governing equations was developed by Mohd-Yusof [38] and Fadlun *et al.* [39]. Similar to Cartesian methods and the IIM, this approach treats the solid boundary as a sharp interface. In contrast to the Cartesian CCM that modifies the Eulerian grid cells in the vicinity of the boundary and applies boundary conditions exactly on the boundary, this method applies boundary conditions at the grid nodes closest to the solid boundary. The specific values of the various flow variables at such near-boundary nodes are calculated by interpolating linearly along an appropriate grid line between the nearest interior node, where flow variables are available from the solution of the governing equations, and the point where the grid line intersects the solid boundary, where physical boundary conditions are known. This approach can be classified as a *hybrid Cartesian/immersed boundary* (HCIB) approach [39].

Gilmanov and Sotiropoulos [40] have developed a new HCIB formulation applicable to 3D flows with arbitrarily complex immersed boundaries (IBs) moving with the prescribed motion. This methodology maintains a sharp fluid–body interface by discretizing the body surface using an unstructured, triangular mesh. The nodes of this mesh constitute a set of Lagrangian control points, which are used to track the motion and reconstruct the instantaneous shape of the moving IB. The reconstruction of the solution near the boundary is carried out by interpolation along the normal to the surface of the body [41].

1.2. Strategies for solving structural deformations

To solve the equations of a solid deformable body the *FE method* is most commonly applied. In this paper, we use material point method (MPM) [42–44] which has certain advantages over standard FE approaches. One of the major advantages of MPM consists in being able to consider large deformations, including situations with material rupture. This is possible due to the fact that MPM is a meshless method and the equations are considered in an Eulerian frame. Other advantage of MPM is its efficiency that enable a speedup by a factor of 4–10 in comparison with standard FE methods [42].

An application of MPM to solve FSI problem is given in [45], which used the same grid for the solid and fluid regions. Both regions are discretized by material points (MPs), and both the fluid and the solid are solved for by the MPM strategy. In this case, it is possible to connect the fluid and solid directly by the accumulation of the grid forces from the fluid and the membrane MPs at the nodes of the background grid. To our knowledge, this is the only application of the MPM to FSI problems, and utilizes the MPM for both the fluid and the solid regions. As discussed next, we have taken different approaches where the MPM is utilized only for the calculations of the structural deformations. This approach retains inherent advantages of Eulerian flow solvers and integrates them with the MPM approach for the solid. Thereafter, as discussed above MPM has specific advantages over the FE method.

1.3. Contribution of the present paper—a combined HIBM & MPM for FSI problems

In the present paper, the hybrid immersed boundary method (HIBM) for the fluid is combined with the MPM for the solid and is presented as an effective strategy for solving FSI problems. The HIBM is a cost-effective strategy for flow problems with complex moving interfaces. The MPM has the advantages of resolving strong structural deformations without grid stretching since

DOI: 10.1002/fld

it uses an arbitrary distribution of nodal points and falls in the category of meshless methods. The MPM nodes communicate with a background Cartesian mesh which is independent of the fluid grid and on which the structural equations are solved for simplicity and computational efficiency. The HIBM & MPM implementation, as presented in this paper, represents an unique approach for solving FSI problems with arbitrary geometrical complexity of the structural deformations.

2. GOVERNING EQUATIONS

Let us consider a 3D deformable structure Ω^s completely immersed in an incompressible fluid domain Ω^f (Figure 1). In the computational fluid domain, the fluid grid is represented by the time-invariant position vector $\mathbf{r}$. The MPs of the structure in the initial solid domain Ω_0^s and the current solid domain Ω^s are represented by $\mathbf{R}_s$ and $\mathbf{r}_s(\mathbf{R}_s, t)$, respectively. The subscript s is used for the solid variables to distinguish the fluid and solid domains. In the fluid calculations, the velocity $\mathbf{v}$ and the pressure p are the unknown fluid field variables. Calculation in the solid domain involves the determination of the nodal displacement $\mathbf{u}_s$, which is defined as the difference in the current and the initial Lagrangian coordinates: $\mathbf{u}_s = \mathbf{r}_s - \mathbf{R}_s$. The velocity $\mathbf{v}_s$ is the material derivative of the displacement $\mathbf{v}_s = \mathrm{d}\mathbf{u}_s/\mathrm{d}t$.

The unsteady, 3D, incompressible Navier–Stokes (NS) equations are solved using an efficient finite-difference method that is second-order accurate both in space and time. A hybrid approach that combines curvilinear grids, and the IBM was used to develop a powerful and very general methodology for efficiently and accurately resolving all geometrical features of the flow. Implementing the HIBM in curvilinear coordinates enables the accurate resolution of boundaries that are not deforming or moving. The deformations of the solid body are calculated as part of the solution procedure by implementing an FSI model. The mathematical model consists of the momentum equations for the flow and the solid body, the continuity equations for the fluid and solid structure, and the appropriate boundary conditions.

The unsteady, 3D, incompressible, NS equations in generalized curvilinear coordinates (repeated indices imply summation $l, r = 1$–3) are

$$\left(\frac{\Gamma}{J}\right)\frac{\partial Q}{\partial t} + \frac{\partial}{\partial \xi^l}(F^l - F_v^l) = 0 \tag{1}$$

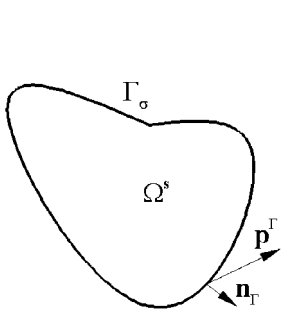

Figure 1. Type of boundary condition on the surface of the body. Solid line is the Neumann boundary with the established force from the fluid.

DOI: 10.1002/fld

where

$$\Gamma = \mathrm{diag}[\beta, 1, 1, 1]^{\mathrm{T}}$$

$$Q = [p, v_1, v_2, v_3]^{\mathrm{T}}$$

$$F^l = \frac{1}{J}[V^l, v_1 V^l + p\xi^l_{x1}, v_2 V^l + p\xi^l_{x2}, v_3 V^l + p\xi^l_{x3}]^{\mathrm{T}}$$

$$F^l_v = \frac{1}{J}\frac{1}{Re}\left[0, gr^{rl}\frac{\partial v_1}{\partial \xi^r}, gr^{rl}\frac{\partial v_2}{\partial \xi^r}, gr^{rl}\frac{\partial v_3}{\partial \xi^r}\right]^{\mathrm{T}}$$

Here β is a positive constant, which is the pseudo-compressibility parameter, p is the static pressure divided by the density, x_i are the Cartesian coordinates, v_i are the Cartesian velocity components, $V^r = v_l \xi^l_{x_r}$ are the contravariant velocity components and $\xi^l_{x_m}$ are the metrics of the geometric transformation, J is the Jacobian of the geometric transformation, g^{rl} is the contravariant metric tensor $g^{rl} = \xi^r_{x_s}\xi^l_{x_s}$, and Re is the Reynolds number.

As mentioned above the equations for the solid deformable body are given in Eulerian frame because the Eulerian background grid is used to solve the solid body equations:

$$\frac{\mathrm{d}\rho_s}{\mathrm{d}t} + \rho_s \nabla \cdot \mathbf{v}_s = 0 \tag{2}$$

$$\rho_s \frac{\mathrm{d}\mathbf{v}_s}{\mathrm{d}t} = \nabla \cdot \boldsymbol{\sigma}_s + \rho_s \mathbf{b}_s \tag{3}$$

$\mathbf{v}_s = \mathrm{d}\mathbf{u}_s/\mathrm{d}t$ is the velocity, ρ_s is the density, $\boldsymbol{\sigma}_s$ is the Cauchy stress tensor, and $\mathbf{b}_s$ is the specific body force. In order to obtain the relationship between the stress and the strain, a Lagrangian formulation is used. For analyzing strain in a material with large displacements and deformation two coordinate systems and the relationship between them is considered. The first system is a material (actual) coordinate system and the second system is a fixed spatial coordinate system. Deformation is quantified by expressing the spatial coordinates of a material particle in the deformed state ($\mathbf{r}_s$) as a function of the coordinates of the same particle in the undeformed state ($\mathbf{R}_s$). Length changes of material element can be determined from the known deformation fields and strain tensors may be calculated. In Figure 2 let Ω^s_0 be a fixed reference configuration of the undeformed body and has components $\mathrm{d}\mathbf{R}_s = (\mathrm{d}X_s, \mathrm{d}Y_s, \mathrm{d}Z_s)$ and let Ω^s be the deformed configuration and has components $\mathrm{d}\mathbf{r}_s = (\mathrm{d}x_s, \mathrm{d}y_s, \mathrm{d}z_s)$ with respect to the Cartesian coordinates (X, Y, Z). An infinitesimal vector $\mathrm{d}\mathbf{R}_s$ at the point $\mathbf{R}_s$ in the reference configuration is mapped to the infinitesimal vector $\mathrm{d}\mathbf{r}_s$ at the point $\mathbf{r}_s$ in the deformed configuration. The displacement of the MP is defined as $\mathbf{u}_s = \mathbf{r}_s - \mathbf{R}_s$ (Figure 2), and the deformation gradient is given by [46]

$$\mathbf{F}^t_0 = \frac{\partial \mathbf{r}_s}{\partial \mathbf{R}_s} \tag{4}$$

The deformation gradient $\mathbf{F}^t_0$ relates the infinitesimal vector $\mathrm{d}\mathbf{R}_s$ to the infinitesimal vector $\mathrm{d}\mathbf{r}_s$ and describes the translational, rotational and tensional motions that the material elements have undergone from an initial time $t = 0$ till the current time t. A measure of strain can be defined by

$$\mathbf{E} = \tfrac{1}{2}[(\mathbf{F}^t_0)^{\mathrm{T}}\mathbf{F}^t_0 - \mathbf{I}] = \tfrac{1}{2}(\mathbf{C} - \mathbf{I}) \tag{5}$$

DOI: 10.1002/fld

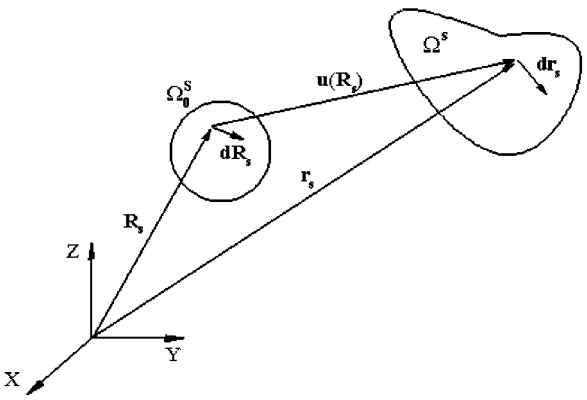

Figure 2. Diagram of deformation of the body.

or, in the tensorial form,

$$\varepsilon_{ij} = \tfrac{1}{2}(C_{ij} - \delta_{ij}) \tag{6}$$

where $\mathbf{E}$ is the Green–Lagrange strain tensor, $\mathbf{C} = (\mathbf{F}_0^t)^{\mathrm{T}}\mathbf{F}_0^t = \|C_{ij}\|$ is the right Cauchy–Green deformation tensor, and $\mathbf{I} = \|\delta_{ij}\|$ is the unit tensor. Note that an important property of the deformation gradient is that it can be decomposed into a product of matrices $\mathbf{F}_t^{t+\tau}$ corresponding to a deformation from t till $t + \tau$ and matrices $\mathbf{F}_0^t$ corresponding to a deformation from initial time $t = 0$ till t

$$\mathbf{F}_0^{t+\tau} = \mathbf{F}_t^{t+\tau}\mathbf{F}_0^t$$

Mechanical properties of a hyperelastic material can be characterized by a strain energy function (the Helmholtz potential) W. For hyperelastic isotropic materials W is a function of the three invariants of the deformation tensor $\mathbf{E}$ or $\mathbf{C}$ and can be written as

$$W = W(I_1, I_2, I_3) \tag{7}$$

where

$$I_1 = \operatorname{tr}\mathbf{C} = C_1^1 + C_2^2 + C_3^3, \quad I_2 = \tfrac{1}{2}[\operatorname{tr}(\mathbf{C}^2) - (\operatorname{tr}\mathbf{C})^2], \quad I_3 = \det\mathbf{C}$$

For nonlinear deformations of a solid incompressible, isotropic body, Mooney's law [46] is used:

$$W = c_1(J_1 - 3) + c_2(J_2 - 3) + \tfrac{1}{2}k(J_3 - 1)^2 \tag{8}$$

where c_1 and c_2 are constants that are the characteristic properties of the material, k is the bulk modulus that is several thousand times as large as their shear modulus, and the reduced invariants are

$$J_1 = I_1(I_3)^{-1/3}, \quad J_2 = I_2(I_3)^{-2/3}, \quad J_3 = (I_3)^{1/2}$$

A simpler model that is used for elastic isotropic material is the neo-Hookean constitutive model [47]

$$W = c_1(J_1 - 3) + \tfrac{1}{2}k(J_3 - 1)^2 \tag{9}$$

DOI: 10.1002/fld

The Cauchy stress in Equation (3) is given by

$$\sigma^{ij} = \frac{1}{2\sqrt{I_3}} \left(\frac{\partial W}{\partial \varepsilon_{ij}} + \frac{\partial W}{\partial \varepsilon_{ij}} \right) \tag{10}$$

In order to obtain a numerical solution for a given FSI problem, certain boundary conditions (indicated by superscript Γ) must be specified for the above-mentioned systems of partial differential equations (Figure 1). The boundary conditions for the fluid velocity are implemented on all the immersed surfaces

$$\mathbf{v}_f = \mathbf{v}^\Gamma \quad \text{on } \Gamma_\sigma \tag{11}$$

where $\mathbf{v}^\Gamma$ is velocity of the immersed surface; the boundary condition for the surface traction on the solid surface is

$$\boldsymbol{\sigma}_f \cdot \mathbf{n}_\Gamma = \mathbf{p}^\Gamma \quad \text{on } \Gamma_\sigma \tag{12}$$

where $\boldsymbol{\sigma}_f$ is stress tensor of the fluid, $\mathbf{p}^\Gamma$ is the traction vector, $\mathbf{n}_\Gamma$ is the normal to the surface Γ.

3. METHODS FOR SOLUTION OF GOVERNING EQUATIONS

3.1. Numerical solution for the Navier–Stokes equations

To solve the system of governing equations (1), a pressure-based, Residual Smoothing Multistage Pseudocompressibility Algorithm developed by Sotiropoulos and Constantinescu [48] was used. This approach combines the pressure-based method [49] and artificial compressibility [50] method to obtain an efficient diagonal pressure-based operator which was implemented in a four-stage Runga–Kutta algorithm. Sotiropoulos and Constantinescu [48] have shown that the proposed algorithm substantially enhances the damping of high-frequency errors on large aspect ratio meshes. In papers [40, 41] the validation of the IBM using a Cartesian grid was demonstrated. A second-order accuracy of the HCIB method was shown for the solution of problems with fluid flow past arbitrary geometry or moving bodies. The IB is treated as a sharp interface and the solution in its vicinity is reconstructed using interpolation along the local normal to the body. To facilitate the calculation of the normal, the body is discretized with an unstructured, triangular grid. The use of unstructured grid greatly enhances the generality of the method as it allows the modeling of arbitrarily complex, 3D IBs and eliminates ambiguities encountered when the reconstruction is carried out by interpolating along grid lines. A grid convergence study was carried out, which showed that the method is second-order accurate. Validation studies were also reported for flow induced by a sphere rotating steadily in a fluid that is at rest sufficiently far from the sphere as well as for the flow induced by a 3D flapping wing in a confined domain. For both cases, good agreement with benchmark solutions and laboratory measurements were obtained. Since the flow solver has been published [48] and validated [40, 41] additional details are not given here.

3.2. Numerical solution of the equations for the solid deformable body

The MPM method is implemented here in a meshless framework. The solid domain is covered with an arbitrary number of MPs. MPs are classified as internal and external (relative to the surface of body). Set of external MPs is triangulated on the surface in such a manner that it is possible to

DOI: 10.1002/fld

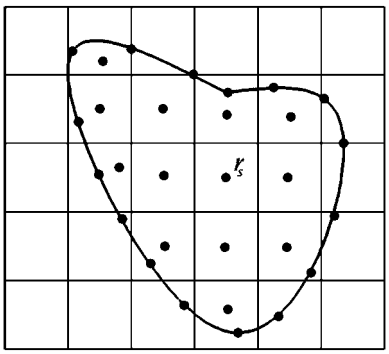

Figure 3. Computational domain and background grid with discrete material points.

define normals to each point on the surface. Normals are necessary for the implementation of the HIBM method, since these normals are used for the interpolation of the fluid variables at the IB nodes.

Let $\mathbf{r}_s^n,\ s=1,\ldots,N_s$ denote the current position of the MPs at time t^n (Figure 3). Each MP has an associated mass, m_s, density, ρ_s^n, velocity, $\mathbf{v}_s^n$, Cauchy stress tensor, $\boldsymbol{\sigma}_s^n$, and strain, ε_s^n at time t^n. MP mass should remain constant with time: $m_s^n = m_s = \text{const}$, insuring that the continuity equation is satisfied.

Mass of the each MP is determined on the basis of density and geometry of the object. For example, for problems where the body is initially a sphere, the total weight of each MP is calculated as

$$m_s = \tfrac{4}{3}\pi R_0^3 \rho_{s0}/N_s$$

where R_0 is the initial radius of the sphere, ρ_{s0} is the initial density. For problems with thin shells or membranes the mass of MP is defined, as

$$m_s = \left(\sum_{l=1}^{N_L} \rho_{s0} h_0 \Delta S_l\right) \Big/ N_s$$

where h_0 is the initial shell thickness, ΔS_l is surface area of lth triangular element, N_L is number of triangular elements discretizing the surface. All variables are updated at each time step in order to satisfy Equation (3). At each time step, information from the MPs is interpolated to a background computational grid. This background grid is Cartesian and chosen only for the computational simplicity to solve the deformable body equations. In a more general case, one may design the background grid to be curvilinear.

To obtain a system of algebraic equations, the momentum equations (3) are converted to a weak form by means of standard FE method techniques:

$$\int_{\Omega_s} \rho_s \left[\frac{\mathrm{d}\mathbf{v}_s}{\mathrm{d}t} \cdot \mathbf{w} + (\boldsymbol{\sigma}_s \otimes \nabla \mathbf{w})/\rho_s\right] \mathrm{d}V = \int_{\Gamma_\sigma} \rho_s \mathbf{p}^\Gamma \cdot \mathbf{w}\, \mathrm{d}S + \int_{\Omega_s} \rho_s \mathbf{b}_s \cdot \mathbf{w}\, \mathrm{d}V \tag{13}$$

where $\mathbf{w}$ denotes the test function, $\mathbf{p}^\Gamma$ is a specific traction vector (i.e. traction divided by mass density), Ω_s is the current configuration, Γ_σ is the part of the boundary with a prescribed traction

DOI: 10.1002/fld

and **w** is assumed to be zero on the boundary with a prescribed displacement (for example, fixed boundaries).

Since the mass density of the particle can be written as

$$\rho_s(\mathbf{r}, t) = \sum_{s=1}^{N_s} m_s \, \delta(\mathbf{r} - \mathbf{r}_s) \tag{14}$$

Equation (13) converts to the following

$$\sum_{s=1}^{N_s} m_s \left[\mathbf{w}(\mathbf{r}_s, t) \cdot \frac{\mathrm{d}\mathbf{v}}{\mathrm{d}t}(\mathbf{r}_s, t) \right] \tag{15}$$

$$= \sum_{s=1}^{N_s} (m_s/\rho_s(\mathbf{r}_s, t))[-\boldsymbol{\sigma}_s(\mathbf{r}_s, t) \otimes \nabla \mathbf{w}(\mathbf{r}_s, t)] \tag{16}$$

$$+ \sum_{s=1}^{N_s} m_s[\mathbf{w}(\mathbf{r}_s, t) \cdot \mathbf{p}^{\Gamma}(\mathbf{r}_s, t) h^{-1}(\mathbf{r}_s, t) + \mathbf{w}(\mathbf{r}_s, t) \cdot \mathbf{b}_s(\mathbf{r}_s, t)]$$

where $h(\mathbf{r}_s, t)$ is the thickness of a shell or membrane (in this paper we consider only such objects). In Equation (14), $\delta(\mathbf{r} - \mathbf{r}_0)$ denotes a Dirac delta function with dimensions of the inverse of volume about a point $\mathbf{r}_0$.

For 3D problems, a background mesh is constructed from 8-node cells. These cells are then employed to define standard nodal basis (shape) functions $S_i(\mathbf{r}_s)$, associated with spatial nodes, $\mathbf{r}_i(t)$, $i = 1, \ldots, N_g$, where N_g is number of nodes of the background grid. Further, in all notations we will use subscript index s for MPs and subscript index i (j) for the background grid nodes. The nodal basis functions are assembled from standard FE shape functions, so that an 8-node cell can use the shape functions given by

$$S_1(\mathbf{r}_s) = (1 - \xi_s)(1 - \eta_s)(1 - \zeta_s)$$

$$S_2(\mathbf{r}_s) = (1 - \xi_s)\eta_s(1 - \zeta_s)$$

$$S_3(\mathbf{r}_s) = (1 - \xi_s)\eta_s\zeta_s$$

$$S_4(\mathbf{r}_s) = (1 - \xi_s)(1 - \eta_s)\zeta_s$$

$$S_5(\mathbf{r}_s) = \xi_s(1 - \eta_s)(1 - \zeta_s)$$

$$S_6(\mathbf{r}_s) = \xi_s\eta_s(1 - \zeta_s)$$

$$S_7(\mathbf{r}_s) = \xi_s\eta_s\zeta_s$$

$$S_8(\mathbf{r}_s) = \xi_s(1 - \eta_s)\zeta_s$$

where $0 \leqslant \xi_s \leqslant 1$, $0 \leqslant \eta_s \leqslant 1$, and $0 \leqslant \zeta_s \leqslant 1$ are the natural coordinates of the MP s (Figure 4).

DOI: 10.1002/fld

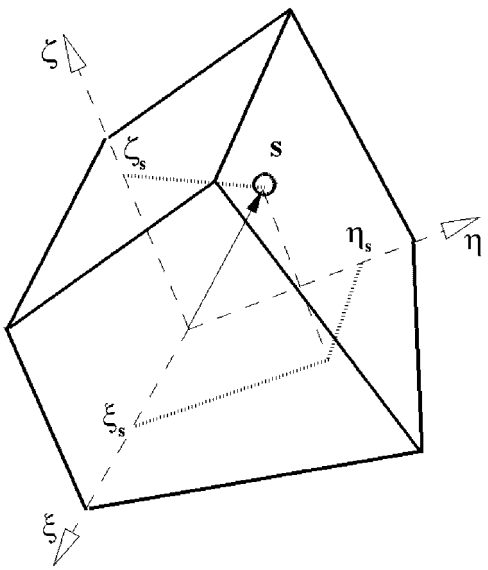

Figure 4. Illustration of the position of a material point **s** in the local curvilinear element (ξ, η, ζ).

Coordinates $\mathbf{r}_s$, displacements $\mathbf{u}_s$, velocities $\mathbf{v}_s$, and test function $\mathbf{w}_s$ can then be represented as follows:

$$\mathbf{r}_s = \sum_{i=1}^{N_g} \mathbf{r}_i S_i(\mathbf{r}_s), \quad \mathbf{u}_s = \sum_{i=1}^{N_g} \mathbf{u}_i S_i(\mathbf{r}_s)$$

$$\mathbf{v}_s = \sum_{i=1}^{N_g} \mathbf{v}_i S_i(\mathbf{r}_s), \quad \mathbf{w}_s = \sum_{i=1}^{N_g} \mathbf{w}_i S_i(\mathbf{r}_s)$$

Substituting the expressions for acceleration and test functions into Equation (15) yields

$$\sum_{i=1}^{N_g} \mathbf{w}_i \cdot \sum_{j=1}^{N_g} m_{ij} \frac{\mathrm{d}\mathbf{v}_j}{\mathrm{d}t} = -\sum_{i=1}^{N_g} \mathbf{w}_i \cdot \sum_{s=1}^{N_s} (m_s/\rho_s)\boldsymbol{\sigma}_s \cdot \nabla S_i|_{\mathbf{r}_s} + \sum_{i=1}^{N_g} \mathbf{w}_i \cdot \mathbf{p}_i^{\Gamma,t} + \sum_{i=1}^{N_g} \mathbf{w}_i \cdot \mathbf{b}_{s,i} \tag{17}$$

Since, in general $\mathbf{w}_i$ is non-zero, the discrete form of governing momentum equation (17) becomes

$$\sum_{j=1}^{N_g} m_{ij} \frac{\mathrm{d}\mathbf{v}_j}{\mathrm{d}t} = \mathbf{f}_i^{\mathrm{int}} + \mathbf{f}_i^{\mathrm{ext}}, \quad 1 \leqslant i \leqslant N_g \tag{18}$$

In Equation (17) mass matrix is given by

$$m_{ij} = \sum_{s=1}^{N_s} m_s S_i(\mathbf{r}_s) S_j(\mathbf{r}_s)$$

The mass matrix is simplified with the corresponding lumped nodal masses $m_i = \sum_{s=1}^{N_s} m_s S_i(\mathbf{r}_s)$ by neglecting non-diagonal terms and introducing a slight amount of numerical dissipation [51];

DOI: 10.1002/fld

therefore, the approximation of momentum equations is given by

$$m_i \frac{\mathrm{d}\mathbf{v}_i}{\mathrm{d}t} = \mathbf{f}_i^{\mathrm{int}} + \mathbf{f}_i^{\mathrm{ext}} \tag{19}$$

where the internal force vector is given by

$$\mathbf{f}_i^{\mathrm{int}} = -\sum_{s=1}^{N_s} (m_s/\rho_s)\boldsymbol{\sigma}_s \cdot G_i(\mathbf{r}_s) \tag{20}$$

with $G_i(\mathbf{r}_s) = \nabla S_i|_{\mathbf{r}_s} = [\partial S_i/\partial x, \partial S_i/\partial y, \partial S_i/\partial z]^{\mathrm{T}}_{\mathbf{r}_s}$. The external force vector is

$$\mathbf{f}_i^{\mathrm{ext}} = \mathbf{p}_i^{\Gamma,t} + \mathbf{b}_{s,i} \tag{21}$$

where discrete-specific traction vector is

$$\mathbf{p}_i^{\Gamma} = \sum_{s=1}^{N_s} m_s \mathbf{p}^{\Gamma}(\mathbf{r}_s, t) S_i(\mathbf{r}_s) h^{-1}(\mathbf{r}_s, t) \tag{22}$$

and the body force is discretized as

$$\mathbf{b}_{s,i} = \sum_{s=1}^{N_s} m_s \mathbf{b}_s(\mathbf{r}_s, t) S_i(\mathbf{r}_s) \tag{23}$$

3.3. MPM algorithm

The following steps are performed at each physical time step in the solid structure solver:

(1) For each MP perform the mapping operation from particles to the nodes of a background mesh:

$$m_i^n = \sum_{s=1}^{N_s} m_s S_i(\mathbf{r}_s^n)$$

where m_i^n is the mass at node i at time t^n, $S_i(x)$ is the nodal basis (shape) function associated with the node i, $\mathbf{r}_s^n$ and m_s are the locations of the particle at time t^n.

(2) Map the momentum from the MP to the nodes of the cell containing these particles:

$$m_i^n \mathbf{v}_i^n = \sum_{s=1}^{N_s} m_s \mathbf{v}_s^n S_i(\mathbf{r}_s^n)$$

(3) Find the internal force vector at each point on the background mesh:

$$\mathbf{f}_i^{\mathrm{int},n} = -\sum_{s=1}^{N_s} \frac{m_s}{\rho_s^n} \boldsymbol{\sigma}_s^n \cdot G_i(\mathbf{r}_s^n)$$

where $G_i(\mathbf{r}_s^n) = \nabla S_i(\mathbf{r}_s^n)$ is a gradient of the basis functions with the ∇S_i components given by

$$\frac{\partial S_i}{\partial x_j} = \frac{\partial S_i}{\partial \xi_s}\left(\frac{\partial \xi_s}{\partial x_j}\right) + \frac{\partial S_i}{\partial \eta_s}\left(\frac{\partial \eta_s}{\partial x_j}\right) + \frac{\partial S_i}{\partial \zeta_s}\left(\frac{\partial \zeta_s}{\partial x_j}\right) \quad (x_j = x, y, z)$$

DOI: 10.1002/fld

(ξ_s, η_s, ζ_s) are the coordinates of MP in the local coordinate system (background mesh) for the structure (Figure 4). For Cartesian coordinates, the components of the internal force are computed as

$$f_{x,i}^{\text{int},n} = -\sum_{s=1}^{N_s} \frac{m_s}{\rho_s^n} \left[\frac{\partial S_i}{\partial x}(\sigma_{xx})_s^n + \frac{\partial S_i}{\partial y}(\sigma_{yx})_s^n + \frac{\partial S_i}{\partial z}(\sigma_{zx})_s^n \right]$$

$$f_{y,i}^{\text{int},n} = -\sum_{s=1}^{N_s} \frac{m_s}{\rho_s^n} \left[\frac{\partial S_i}{\partial x}(\sigma_{xy})_s^n + \frac{\partial S_i}{\partial y}(\sigma_{yy})_s^n + \frac{\partial S_i}{\partial z}(\sigma_{zy})_s^n \right]$$

$$f_{z,i}^{\text{int},n} = -\sum_{s=1}^{N_s} \frac{m_s}{\rho_s^n} \left[\frac{\partial S_i}{\partial x}(\sigma_{xz})_s^n + \frac{\partial S_i}{\partial y}(\sigma_{yz})_s^n + \frac{\partial S_i}{\partial z}(\sigma_{zz})_s^n \right]$$

(4) Find the external force vector at the background mesh using the dynamic boundary conditions (12):

$$\mathbf{f}_i^{\text{ext},n} = \mathbf{p}_i^{\Gamma,n} + \mathbf{b}_{s,i}^n$$

(5) Update the nodal velocity:

$$\mathbf{v}_i^L = \mathbf{v}_i^n + \frac{\Delta t}{m_i^n}(\mathbf{f}_i^{\text{int},n} + \mathbf{f}_i^{\text{ext},n})$$

(6) Map the current velocities back to the MP:

$$\mathbf{v}_s^{n+1} = \mathbf{v}_s^n + \sum_{i=1}^{N_g} (\mathbf{v}_i^L - \mathbf{v}_i^n) S_i(\mathbf{r}_s^n)$$

(7) Compute the displacement and current position of MPs:

$$\Delta \mathbf{u}_s^n = \Delta t \sum_{i=1}^{N_g} \mathbf{v}_i^L S_i(\mathbf{r}_s^n)$$

$$\mathbf{r}_s^{n+1} = \mathbf{r}_s^n + \Delta \mathbf{u}_s^n$$

(8) Map the displacement of MPs to the background grid:

$$\Delta \mathbf{u}_i^n = \sum_{s=1}^{N_s} \Delta \mathbf{u}_s^n S_i(\mathbf{r}_s^n)$$

(9) Compute the deformation gradient matrix during the time step Δt:

$$\mathbf{F}_{s,n}^{n+1} = \sum_{i=1}^{N_g} [G_i(\mathbf{r}_s) \cdot \Delta \mathbf{u}_i^n + I]$$

(10) Compute the deformation gradient matrix from $t = 0$ to $t = t^{n+1}$:

$$\mathbf{F}_{s,0}^{n+1} = \mathbf{F}_{s,n}^{n+1} \mathbf{F}_{s,0}^n$$

DOI: 10.1002/fld

(11) Compute the Green–Lagrangian strain tensor:

$$\boldsymbol{\varepsilon}_s^{n+1} = \tfrac{1}{2}[(\mathbf{F}_{s,0}^{n+1})^{\mathrm{T}}\mathbf{F}_{s,0}^{n+1} - I]$$

(12) Calculate the stresses by constitutive equation:

$$\boldsymbol{\sigma}_s^{n+1} = \boldsymbol{\sigma}(\boldsymbol{\varepsilon}_s^{n+1})$$

At the end of these steps, the complete characteristics of the deformable body will be found. Steps (6) and (7) define the new velocity and position of the solid, respectively. These values will be used for the solution of the fluid equations.

Note that this MPM algorithm can be used for volumetric bodies. To solve structures that are thin membranes and shells we have changed step (3) of this algorithm to ensure the absence of normal forces [52] from the stresses on the membrane.

(3a) Consider MP s and adjacent triangular elements (Figure 5). For each line originating from s we define extension ratios $\lambda_{s,k}^n$ of linear element k:

$$\lambda_{s,k}^n = \mathrm{d}s_k^n / \mathrm{d}s_{0,k}$$

where $\mathrm{d}s_k^n$ and $\mathrm{d}s_{0,k}$ are the stretched and the original undeformed lengths of the element k.

(3b) Thereafter, by using constitutive equation [53] we define stresses $T_{s,k}$ as

$$T_{s,k}^n = 4h_0\left(1 - \frac{1}{(\lambda_{s,k}^n)^6}\right)(c_1 + (\lambda_{s,k}^n)^2 c_2)$$

(3c) The full internal force at MP is defined by summation over all linear elements k originating from s:

$$\mathbf{f}_s^{\mathrm{int}} = \sum_k T_{s,k}\mathbf{e}_k$$

where $\mathbf{e}_k$ is tangential vector along the kth linear element.

(3d) The internal force vector at each point on the background grid is found as

$$\mathbf{f}_i^{\mathrm{int}} = \sum_{s=1}^{N_s} \mathbf{f}_s^{\mathrm{int}} S_i(\mathbf{r}_s^n)$$

With respect to this procedure there will be no normal components to the surface of the membrane. This model for the membrane is similar to a model for a polymeric net [54], where the energy functional of a free polymer is considered as the energy of a chain of connected springs.

3.4. Treatment of flexible immersed boundaries

The MPM will define a new position and velocity for the solid body at each time step. The flow equations have to be then solved around the newly defined surface of the body. In this section, we discuss the numerical treatment in domains with complex, flexible, IBs, which move under forces from the fluid. Our approach consists of the local reconstruction of the solution near the IB such

DOI: 10.1002/fld

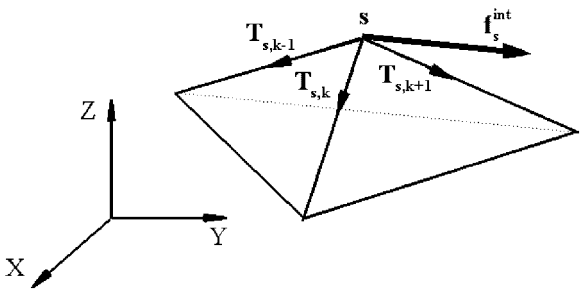

Figure 5. Illustration of how the internal force at material point **s** for a membrane is calculated.

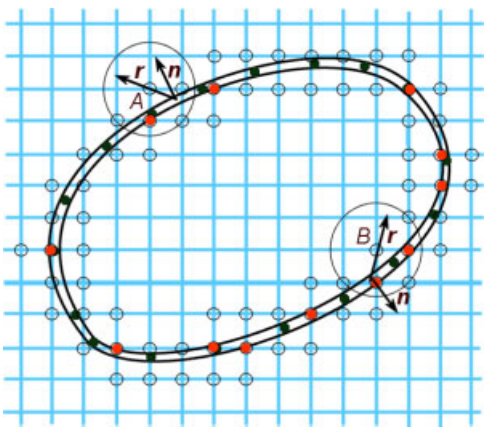

Figure 6. Schematic illustrating the search algorithm for identifying the IB nodes for a complex immersed boundary. Circles (open) are the near-boundary nodes that are identified in the first step of the search procedure. Open circles mark the IB near-boundary nodes. Black nodes are material points. Red nodes are situated on the surface of the body.

that Equations (11) and (12) are satisfied exactly on the surface of the body at every instant in time.

In the HCIB formulation proposed in [41, 40] the IB is treated as a sharp interface. Boundary conditions are applied at nodes in the immediate vicinity of the IB (see Figure 6) by reconstructing the solution along the well-defined normal to the body direction using information from the fluid nodes and the known solution on the surface of the body. To facilitate the reconstruction of the solution in the vicinity of arbitrarily complex IBs, the IB is discretized using an unstructured, triangular mesh with M triangular elements of size similar to the grid spacing in the vicinity of the body.

First, we locate all grid nodes that are in the immediate vicinity of the body. At this stage we do not distinguish between internal and external nodes to the body but rather seek to identify all *near-boundary* nodes located within a small search radius ds_0 from some region of the body. These nodes are marked with open circles in Figure 6. A grid node will be designated as a *near-boundary*

DOI: 10.1002/fld

node if its position vector $\mathbf{r}_{\mathrm{nb}}$ satisfies the following condition:

$$\min_{m=1,M} |\mathbf{r}_{\mathrm{nb}} - \mathbf{r}_{m+1/2}| < \mathrm{d}s_0 \tag{24}$$

where $\mathbf{r}_{m+1/2}$ is the position vector of the centroid of the mth triangular element on the interface, $\mathrm{d}s_0$ is a prescribed search radius (see Figure 6), which is set approximately equal to the minimum mesh spacing in the vicinity of the body. After the above search has been completed a total number of NB grid nodes (i.e. $\mathrm{nb} = 1, \mathrm{NB}$) have been identified.

In the second step of the algorithm we separate the NB near-boundary nodes into nodes internal and external (IB nodes) to the body as follows. For every near-boundary grid node nb, we first identify all triangular elements (and their position vector $\mathbf{r}_{m+1/2}$) located within a sphere of radius $\mathrm{d}s_0$ centered at node nb and for these triangular elements we examine the sign of the scalar product

$$\mathbf{n}_{m+1/2} \cdot (\mathbf{r}_{\mathrm{nb}} - \mathbf{r}_{m+1/2})$$

If $\mathbf{n}_{m+1/2} \cdot (\mathbf{r}_{\mathrm{nb}} - \mathbf{r}_{m+1/2}) > 0$ for at least one boundary node $\mathbf{r}_{m+1/2}$ within the local search neighborhood, then the grid node $\mathbf{r}_{\mathrm{nb}}$ is external to the body (see node A in Figure 6). If $\mathbf{n}_{m+1/2} \cdot (\mathbf{r}_{\mathrm{nb}} - \mathbf{r}_{m+1/2}) < 0$ for all triangular elements $\mathbf{r}_{m+1/2}$ within the local search neighborhood, then the grid node $\mathbf{r}_{\mathrm{ns}}$ is internal to the body (see node B in Figure 6).

After the IB nodes have been determined, boundary conditions need to be specified for the velocity field and pressure at all IB nodes at the new physical time level $n + 1$. These boundary conditions consist of Dirichlet conditions for the velocities (no-slip conditions, fluid velocity on the surface equal to the velocity of the rigid or deformable body) and Neumann conditions for the pressure (derived from the projection of momentum equations on the normal to the surface). These boundary conditions, along with the flow solution at the interior nodes adjacent to the IB points, are used to define the values at the IB points. Additional details are described in [41].

It is important to note that at some IB nodes located in regions where the curvature of the body is changing rapidly in space, the projection onto the surface of the body may not be uniquely defined or even exist. At such nodes, boundary conditions are reconstructed by interpolating along the line defined by the IB node and the nearest node on the surface of the body.

4. AN FSI ALGORITHM

The FSI algorithm consists of successive solutions of fluid and solid equations with communication between them at each time step. If all characteristics of the fluid ($\mathbf{v}_f^n$, p_f^n) and the solid ($\mathbf{r}_s^n$, $\mathbf{v}_s^n$, σ_s^n) are known at some time t^n, the sequence of FSI algorithm is as follows:

(1) *Solve the flow equations*. Solve Equations (1) for the new fluid velocity $\mathbf{v}_f^{n+1}$ and pressure p_f^{n+1}.
(2) *Boundary conditions for the solid body coupling the fluid and the solid*. Interpolate the full stresses from the fluid $p_{ij}^{n+1} = -p_f^{n+1}\delta_{ij} + \tau_{ij}^{n+1}$ onto the surface of the body and evaluate the external forces $\mathbf{p}_s^{\Gamma}$ for all MPs. Here, τ_{ij}^{n+1} are the viscous stresses in the fluid.
(3) *Solve structure equations*. Interpolate masses m_s, momentum $m_s\mathbf{v}_s$, of MPs onto the background grid. Find internal $\mathbf{f}_i^{\mathrm{int},n+1}$ and external $\mathbf{f}_i^{\mathrm{ext},n+1}$ forces. Solve the equations of solid body Equations (19), updating the new position $\mathbf{r}_s^{n+1}$ and the velocities $\mathbf{v}_s^{n+1}$ of the MPs.

DOI: 10.1002/fld

(4) *Boundary conditions for the fluid.* Identify the IB nodes and evaluate the velocities and pressure at IB nodes.
(5) *Return* to the solution of Equations (1) for the next time step.

This algorithm is implemented explicitly, which implies that the sequence of fluid and solid solutions are implemented one after another.

5. RESULTS AND VALIDATION

The goal of the present paper is to develop and validate the combined HIBM & MPM as a strategy for solving FSI problems with strong structural deflections. To the best of our knowledge this is the first time that a sharp-interface method for the fluid (e.g. HIBM) has been integrated with the MPM for the structure and successfully applied to simulate flows past 3D moving bodies with deforming or moving surfaces.

5.1. Inflating a thin sphere

We have solved the problem of a spherical incompressible thin shell inflating under the action of internal pressure. The objective of this initial simulation is to test the implementation of the MPM alone with a problem that has an analytical solution. Although this is a 1D problem, we have used the whole 3D algorithm to solve equations of the solid body. For Mooney's material, an analytical solution that provides the relation between the radial deformations and the pressure is known [53]

$$p = \frac{8h_0}{r_0\lambda_1}(1 - \lambda_1^{-6})(c_1 + c_2\lambda_1^2) \tag{25}$$

Depending on the ratio c_2/c_1 there are possible existence of anomalous regime [53] when radius increases with a decrease in pressure inside the sphere. We have used $c_2/c_1 = 0.5$ where there is monotonic strain–pressure relationship, and for the validation test we have used $c_1 = 20$, $c_2 = 10$. The sphere surface is discretized with 350, 1026, and 3782 triangular elements. In order to obtain the needed stationary solution we have used a quasi-stationary approach. The pressure inside the sphere was increased incrementally and at each value of the internal pressure the solution was calculated. As seen in Figure 7 the agreement of our numerical solution with the analytical solution is rather good and it converges to the exact solution with decreasing of the space discretization of the surface.

5.2. Rigid sphere falling in a box

To validate the FSI algorithm we solved a problem of a sphere falling in a box (solid side and bottom walls and open top) under the action of gravitational forces. We compared the data from the experiments of ten Cate *et al.* [55] with our numerical simulation obtained by the solution of the coupled system of fluid and solid equations. The complete HIBM & MPM algorithm was therefore implemented and tested.

ten Cate *et al.* [55] performed experimental and computational studies of a nylon sphere with a diameter $d = 0.015$ m and density $\rho_s = 1120\ \mathrm{kg/m^3}$ falling in the box with the size of $0.1 \times 0.1 \times 0.16\ \mathrm{m^3}$ under the action of gravitational force. Initial position of the particle was at a height $h = 0.12$ m from the bottom of the box (Figure 8). As the particle is rigid we have

DOI: 10.1002/fld

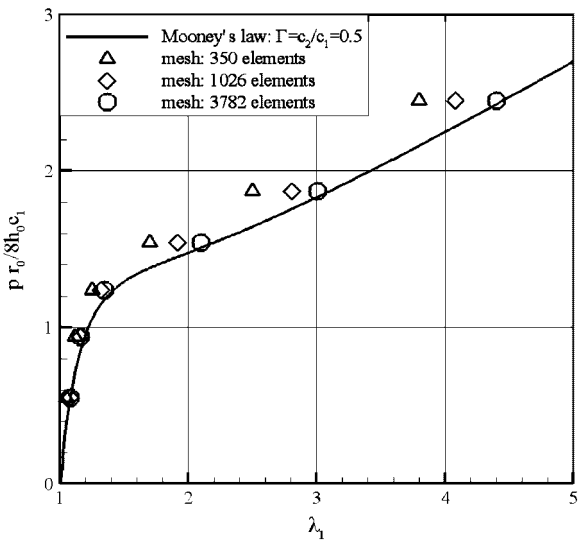

Figure 7. Comparison of the calculated variation of sphere radius with pressure with the analytical solution [53]. Solid line is an exact solution, and triangles, squares, and circles are the computed data for 350, 1026 and 3782 triangular elements, respectively.

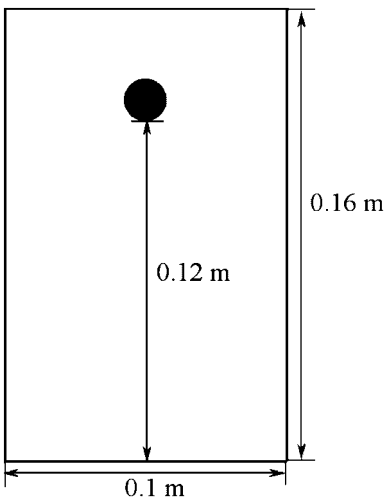

Figure 8. Picture of a box used in the falling sphere experiment of [55].

used the full FSI algorithm with very high mechanical properties of the material of the sphere: $c_1 = 10^5$, $c_2 = 10^5$, where c_1 and c_2 are constants describing the properties of Mooney's material [53].

Calculations were carried out for Reynolds number $Re = 11.6$ and 31.9 as per the property and flow date given by ten Cate *et al.* [55], and is given in Table I. The Reynolds number was defined

DOI: 10.1002/fld

Table I. Fluid properties in the experiments of ten Cate *et al.* [55].

# Experiment	ρ_f (kg/m^3)	μ_f (10^{-3} Pa s)	u_∞ (m/s)	Re
1	962	113	0.091	11.6
2	960	58	0.128	31.9

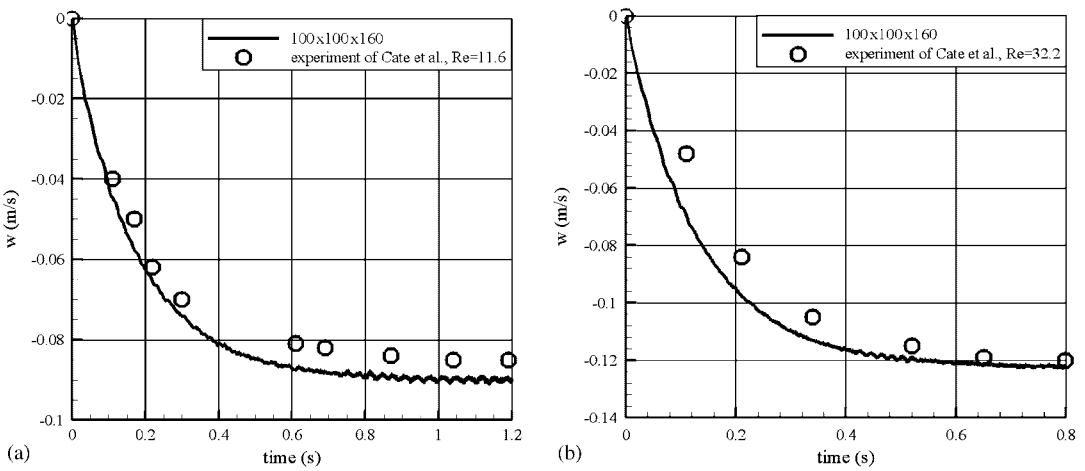

Figure 9. Simulation of a falling sphere in a tube: (a) $Re = 11.6$ and (b) $Re = 31.9$. Fluid grids $100 \times 100 \times 160$ were used. Experimental data [55] are shown in open circle.

on the basis of the terminal velocity of the particle. To discretize the computational region we used a grid with $100 \times 100 \times 160$ grid nodes. A physical time step of $\Delta t = 10^{-2}$ was used. The sphere surface was discretized with 3782/1893 triangular elements/nodes. These grid sizes were found to give grid-independent results. In accordance with experiment, no-slip boundary conditions were established along all the walls of the box and non-reflecting boundary conditions [56] were implemented on the top boundary.

Figure 9 shows the time history of velocities of the particle falling in the box for $Re = 11.6$ (a) and $Re = 31.9$(b). As is seen from Figure 9, the particle experiences an initial acceleration and eventually achieves a steady terminal velocity. Comparisons of our solutions with the experimental data [55] show a very good agreement thus validating the FSI algorithm for a rigid material.

5.3. Rigid and elastic sphere falling in a channel

Figure 10 shows the streamlines and pressure contours for the rigid sphere falling in an unbounded channel at three time instances. The sphere is seen to retain its shape during its descent, symmetry across the channel centerline is reasonably well preserved, and the separation zones expectedly originate at the 90° position on either side.

As an example of FSI problem where the solid object deforms under action of fluid stresses, we have extended the above problem and solved it for a falling soft sphere in an unbounded channel. The sphere material properties were arbitrarily selected and for Mooney's law $c_1 = 20, c_2 = 10$

DOI: 10.1002/fld

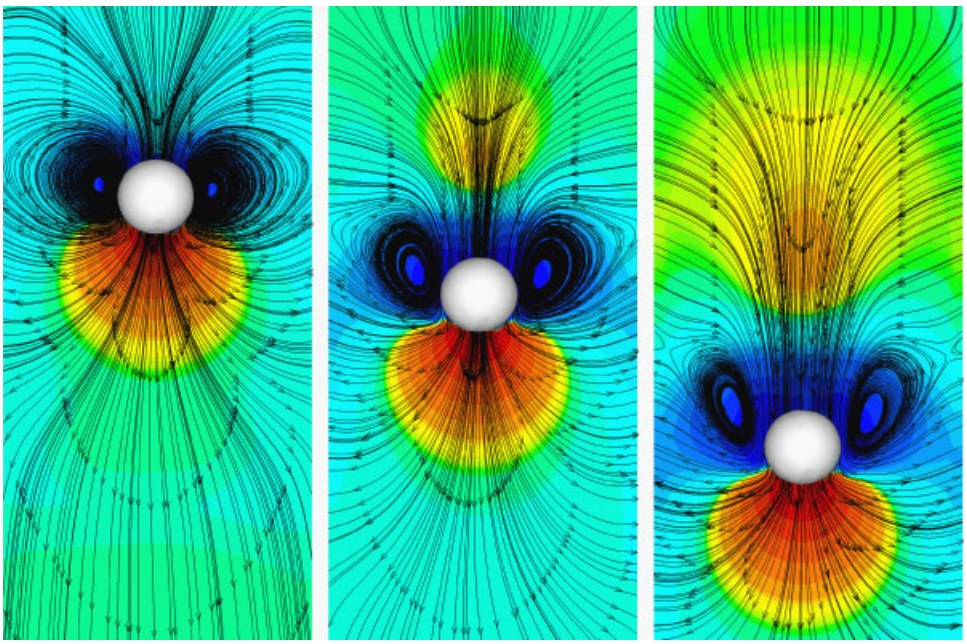

Figure 10. Pressure contours and streamlines for a falling rigid sphere at three time instances.

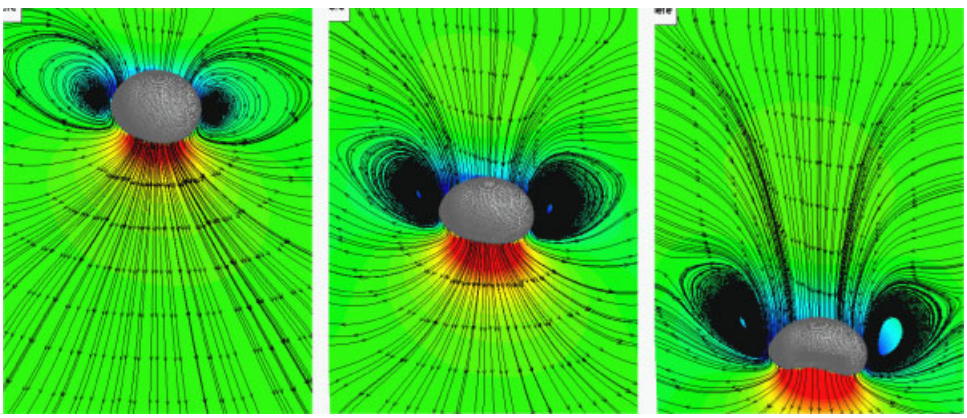

Figure 11. Pressure contours and streamlines for a falling soft sphere at three time instances.

were used. Calculations were carried out on a $71 \times 71 \times 141$ grid with 7770 triangular elements on the surface of the sphere. Here, non-reflecting boundary conditions were used on all sides of the computational region. Figure 11 shows the pressure contours and streamlines at three time levels. In order to show more clearly the deformation of the sphere we have used a specific viewing angle and the observed asymmetry is due to the viewing orientation. It can be seen in Figure 11 that considerable deformation of the sphere occurs under the action of the stresses that act on the

DOI: 10.1002/fld

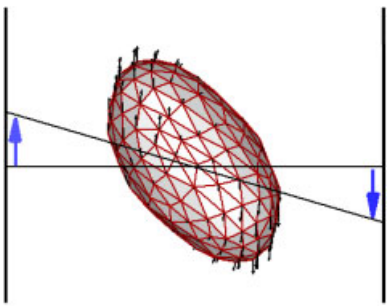

Figure 12. Capsule in a shear flow.

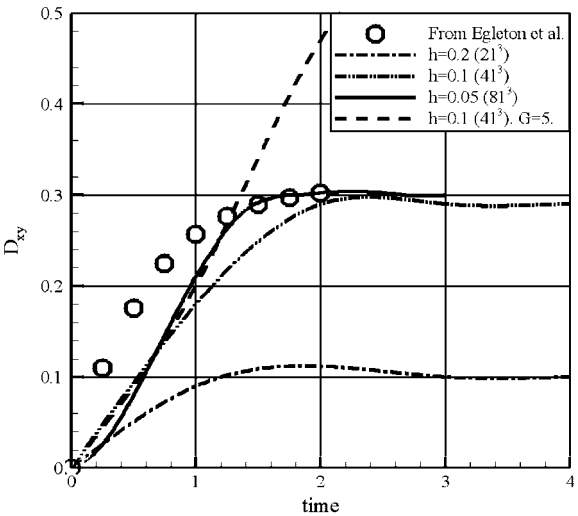

Figure 13. Comparison of computed Taylor parameter with data from Egleton and Popel [47]. Grid refinement was performed for three grids: 21^2, 41^3, and 81^3 for $G = 0.05$. Dashed line represents the data for $G = 5$.

surface of the sphere. The originally spherical solid body assumes a kidney-beam shape with a pronounced concave arch along the bottom. The pressure contours can be seen to be considerably altered relative to the rigid sphere leading to a higher difference of the pressure between the front and back side of the sphere.

5.4. Capsule in a shear flow

As an extension of the above problem, we consider the behavior of an initially spherical capsule in a shear flow. This problem represents a model of a biological deformable red blood cell (RBC) which is deformed under the action of shear stresses and pressure acting on its boundary surface. We use solution of this problem as a validation test using the results in [47] for conformation.

DOI: 10.1002/fld

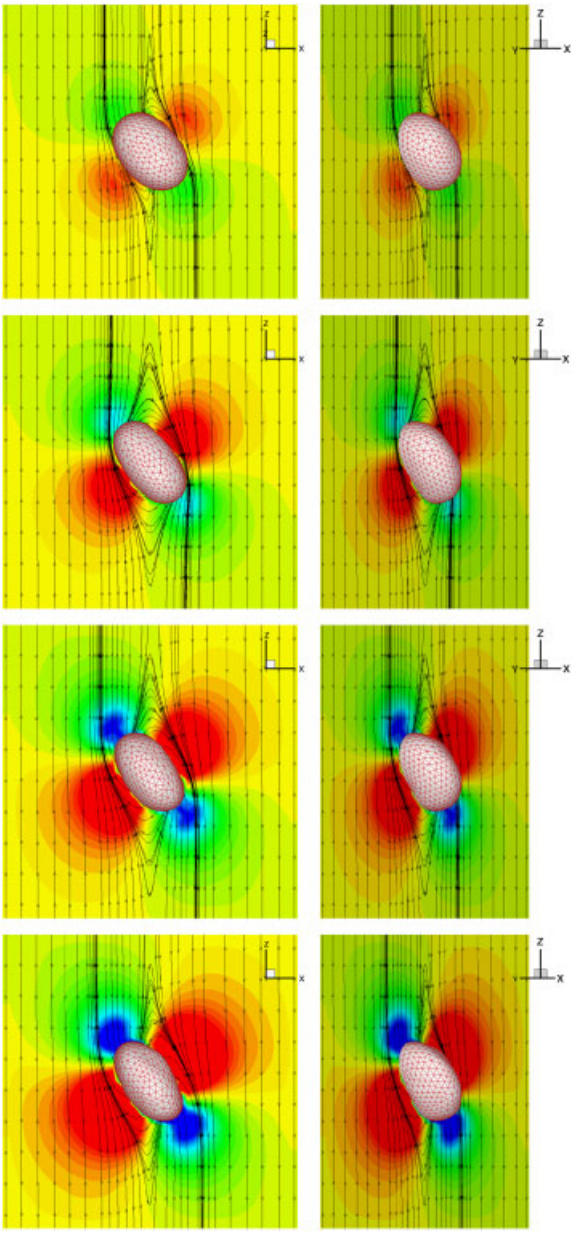

Figure 14. Deformation of a capsule in a shear flow. Pressure contours and streamlines. Instantaneous snapshots of the shape of the capsule at the $y = 0$ diametral plane depicting the stages of the flow evolution toward steady state for $G = 0.05$: (a) plane $y = 0$ and (b) the same pictures with viewpoint $\varphi = 45°$.

We consider this RBC as a capsule freely suspended in a viscous shear flow. The capsule consists of an elastic membrane containing another viscous fluid. We used the same fluid properties both inside and outside of the capsule. The shear flow is generated by initial conditions and by imposed

DOI: 10.1002/fld

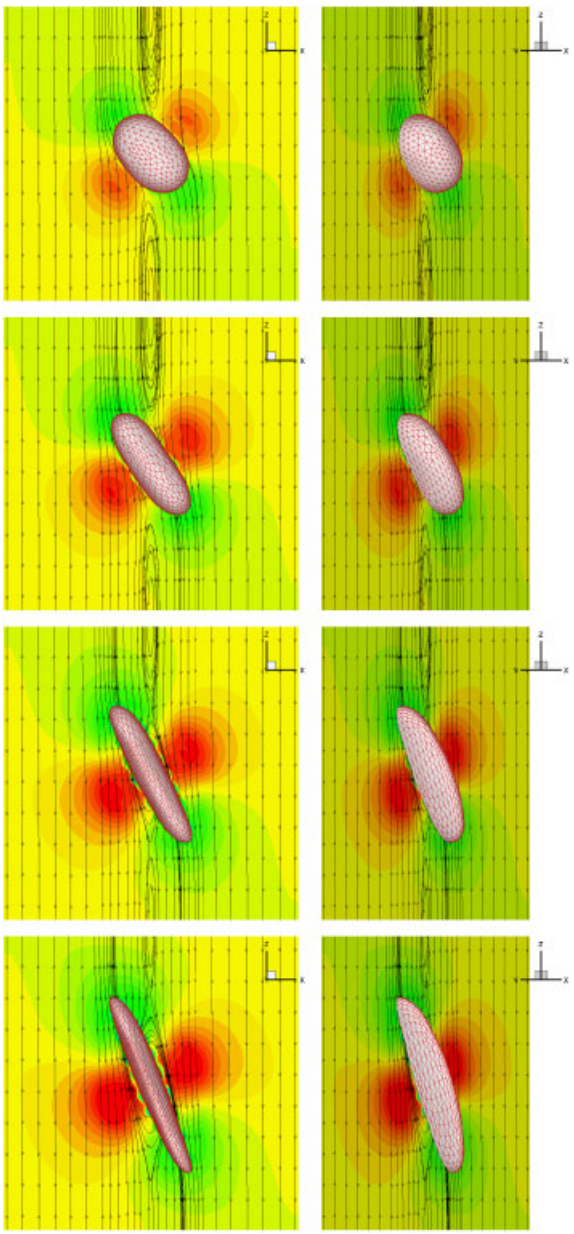

Figure 15. Deformation of a capsule in a shear flow. Pressure contours and streamlines. Instantaneous snapshots of the shape of the capsule at the $y = 0$ diametral plane depicting the stages of the flow evolution toward steady state for $G = 5.0$: (a) plane $y = 0$ and (b) the same pictures with viewpoint $\varphi = 45°$.

shear velocity boundary conditions on left and right sides of the channel (Figure 12). In the flow direction periodic boundary conditions are used. The deformations of capsules have been considered by several authors (see, for example, [47, 54, 57]).

DOI: 10.1002/fld

In order to compare our computational data with [47] we used a neo-Hookean membrane strain energy given by

$$W = \frac{Eh}{6}(\lambda_1^2 + \lambda_2^2 + \lambda_3^2 - 3)$$

where E is Young's modulus, h is the membrane thickness, λ_1, λ_2, and $\lambda_3 = 1/(\lambda_1\lambda_2)$ are principal strains. One of the characteristic parameter of the behavior of the capsule in shear flow is the dimensionless shear rate number [57] given by

$$G = \mu\dot{\gamma}a/\mu_{\mathrm{RBC}}$$

where μ is the fluid viscosity, $\dot{\gamma}$ is the shear rate, a is the initial radius of the capsule, $\mu_{\mathrm{RBC}} = Eh$ is the extension modulus of the membrane. We carried out simulations for the following set of parameters: $\mu = 1.2 \times 10^{-3}$ Pa s, $a = 3.3 \times 10^{-6}$ m, $\mu_{\mathrm{RBC}} = 6 \times 10^{-6}$ Pa m, $\dot{\gamma} = 75, 7500\,\mathrm{s}^{-1}$, leading to a $G = 0.05, 5.0$.

Grid refinement study was made for three grids 21^3, 41^3, and 81^3 with mesh sizes $h = 0.2, 0.1, 0.05$, respectively. Corresponding to each of these fluid grids, different number of material nodes $N_s = 350, 515, 1893$ on the surface of the membrane were used.

Figure 13 shows a comparison of the computed Taylor parameter

$$D_{xy} = (L - B)/(L + B)$$

with the published data. Here L is the length of the cross section of the capsule and B is the breadth. The comparison was made for $G = 0.05$ and there is good agreement with [47] for the mean terminal value of D_{xy}. Difference in the time evolution of D_{xy} is due to the difference in the initial conditions of the problem. In [47] the capsule was held stationary until the shear flow was fully developed. In our case, the capsule was put in the developing shear flow (i.e. for $t<0$, the capsule and channel walls were stationary and at $t = 0$, the walls begin moving to generate shear) and deformations of the capsule were delayed.

Figures 14 and 15 show the results of solution of the FSI problem of deformation of an initially spherical soft capsule with time for $G = 0.05$ and 5, respectively. The deformation of the capsule from the initial shape is clearly evident particularly for $G = 5.0$, where the capsule is showed into a pancake shape. Two perspectives are shown in Figures 14 and 15. It is clear from Figure 15 that there is a significant stretching of the capsule shell.

6. CONCLUDING REMARKS

We have developed an effective numerical method for simulating fluid–structure interaction (FSI) problems with large deformations of solid body. The method developed uses the *hybrid immersed boundary method* (HIBM) for resolving complex boundaries for the fluid flow, and couples this with the *material point method* (MPM) for the structural stresses and deformation. The combined method is implemented in the framework of a finite-difference procedure on curvilinear grid.

The benefits of the proposed FSI approach is based on using effective methods for both the fluid and the structure solvers. HIBM of second-order accuracy allows the solution of complex arbitrary bodies immersed in the fluid more effectively compared with the body-fitted methods like *arbitrary Lagrangian–Eulerian* method with grid regeneration at every time step. MPM, as a

DOI: 10.1002/fld

structure solver on a Cartesian background grid, is more efficient in comparison with *finite element method* and, as mentioned earlier, MPM allows the solution for arbitrary large deformation without numerical limitations.

The methodology is ideally suited for flow problems with initially complex geometries where the surfaces undergo large structural deformation. The methodology is demonstrated on a number of simple test cases and show reasonable qualitative and quantitative agreement with the published results. Although all the results presented here were obtained on the Cartesian grid, we have developed this method for a general case with curvilinear grids. This strategy for simulation of iris deformation in a human eye has been presented in [58, 59].

ACKNOWLEDGEMENTS

This work was supported by a Health Excellence Grant to the Biological Computation and Visualization Center (BCVC) by the State of Louisiana. Their support is greatly appreciated.

REFERENCES

1. Scardovelli R, Zaleski S. Direct numerical simulation of free-surface and interfacial flow. *Annual Review of Fluid Mechanics* 1999; **31**:567–603.
2. Popinet S, Zaleski S. A front-tracking algorithm for accurate representation of surface tension. *International Journal for Numerical Methods in Fluids* 1999; **30**:775–793.
3. Fritts MJ, Cowley W, Trease HE. *The Free Lagrange Method*. Lecture Notes in Physics. Springer: New York, 1985; 238.
4. Hirt CW, Amsden AA, Cook JL. An arbitrary Lagrangian–Eulerian computing method for all flow speeds. *Journal of Computational Physics* 1974; **14**(3):227–253.
5. Donea J, Giuliani S, Halleux JP. An arbitrary Lagrangian–Eulerian finite element method for transient dynamic fluid–structure interactions. *Computer Methods in Applied Mechanics and Engineering* 1982; **33**:689–723.
6. Belytschko T, Kennedy J. Finite element approach to pressure wave attenuation by reactor fuel subassemblies. *Journal of Pressure Wave Technology* 1975; 172–177.
7. Donea J, Fasoli-Stella P, Giuliani S. Finite element solution of transient fluid–structure problems in Lagrangian coordinates. *Proceedings of the International Meeting on Fast Reactor Safety and Related Physics*, Chicago, vol. 3, 1976; 1427–1435.
8. Liu H, Kawachi K. A numerical study of undulatory swimming. *Journal of Computational Physics* 1999; **155**(2):223–247.
9. Tezduyar TE, Sathe S, Stein K. Solution techniques for the fully discretized equations in computation of fluid–structure interactions with the space–time formulations. *Computer Methods in Applied Mechanics and Engineering* 2006; **195**:5743–5753.
10. Tezduar TE. Stabilized finite element formulation for incompressible flow computations. *Advanced in Applied Mechanics* 1991; **28**:1–44.
11. Noh WF. CEL: a time-dependent, two-space-dimensional, coupled Eulerian–Lagrange code. *Methods in Computational Physics*. Academic Press: New York, 1964; 117–179.
12. Udaykumar HS, Kan H-C, Shyy W, Tran-Son-Tay R. Multiphase dynamics in arbitrary geometries on fixed Cartesian grids. *Journal of Computational Physics* 1997; **137**:366–405.
13. Almgren AS, Bell JB, Colella P, Marthaler T. A Cartesian grid projection method for the incompressible Euler equations in complex geometries. *SIAM Journal on Scientific Computing* 1997; **18**(5).
14. Johansen H, Colella P. A cartesian grid embedded boundary method for Poisson's equation on irregular domains. *Journal of Computational Physics* 1998; **147**:60–85.
15. Udaykumar HS, Mittal R, Rampunggoon P, Khanna A. Sharp interface cartesian grid method for simulating flows with complex moving boundaries. *Journal of Computational Physics* 2001; **174**:345–380.
16. Popinet S. Gerris: a tree-based adaptive solver for the incompressible Euler equations in complex geometries. *Journal of Computational Physics* 2003; **190**(2):572–600.

DOI: 10.1002/fld

17. Kirkpatrick MP, Armfield SW, Kent JH. A representation of curved boundaries for the solution of the Navier–Stokes equations on a staggered three-dimensional Cartesian grid. *Journal of Computational Physics* 2003; **184**(1):1–36.
18. Hirt CW, Nichols BD. Volume of fluid (VOF) methods for the dynamics of free boundaries. *Journal of Computational Physics* 1981; **39**:201–225.
19. Hyman JM. Numerical methods for tracking interfaces. *Physica D* 1984; **12**:396–407.
20. Noh W, Woodward P. SLIC (simple line interface calculation). *Proceedings of the 5th International Conference on Fluid Dynamics*. Lecture Notes in Physics, vol. 59. Springer: Berlin, 1976; 330–340.
21. Ashgriz N, Poo JY. Flux linesegment model for advection and interface reconstruction. *Journal of Computational Physics* 1991; **92**:449–468.
22. Osher SJ, Sethian JA. Front propagating with curvature depend speed: algorithm based on Hamilton–Jacobi equations. *Journal of Computational Physics* 1988; **79**:12–49.
23. Sethian JA, Smereka P. Level set method for fluid interface. *Annual Review of Fluid Mechanics* 2003; **35**:341–372.
24. Enright D, Fedkiw R, Ferziger J, Mitchel I. A hybrid particle level set method for improved interface capturing. *Journal of Computational Physics* 2002; **183**:83–116.
25. Saul'ev VK. On the solution of some boundary value problems on high performance computers by fictitious domain method. *Siberian Mathematical Journal* 1963; **4**(4):912–925 (in Russian).
26. Glowinski R, Pan T-W, Hesla TI, Joseph DD, Periaux J. A fictitious domain method with distributed Lagrange multipliers for the numerical simulation of particulate flow. *Contemporary Mathematics* 1998; **218**:121–137.
27. Glowinski R, Pan TW, Hesla TI *et al*. 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* 2001; **169**:363–426.
28. Bernardi C, Maday Y, Patera AT. Domain decomposition by the mortar element method. In *Asymptotic and Numerical Methods for PDEs with Critical Parameters*, Kaper HG, Garbey M (eds). NATO ASI Series C: Mathematical and Physical Sciences, vol. 384. Kluwer: Dordrecht, 1993; 269–286.
29. Baaijens FPT. A fictious domain/mortar element method for fluid–structure interaction. *International Journal for Numerical Methods in Fluids* 2001; **35**:743–761.
30. De Hart J, Peters GWM, Schreurs PJG, Baaijens FPT. A three-dimensional computational analysis of fluid–structure interaction in the aortic valve. *Journal of Biomechanics* 2003; **36**:103–112.
31. Peskin CS. Flow patterns around heart valves: a numerical method. *Journal of Computational Physics* 1972; **10**:252–271.
32. LeVeque RJ, Li Z. The immersed interface method for elliptic equations with discontinuous coefficients and singular sources. *SIAM Journal on Numerical Analysis* 1994; **31**:1019.
33. Peskin CS. Numerical analysis of blood flow in the heart. *Journal of Computational Physics* 1977; **25**:220–252.
34. Lee L, LeVeque RJ. An immersed interface method for incompressible Navier–Stokes equations. *SIAM Journal on Scientific Computing* 2003; **25**(3):832–856.
35. Xu S, Wang ZJ. An immersed interface method for simulating the interaction of a fluid with moving boundaries. *Journal of Computational Physics* 2006; **216**:454–493.
36. Zhang L, Gerstenberg A, Wang X, Liu WK. Immersed finite element method. *Computer Methods in Applied Mechanics and Engineering* 2004; **193**:2051–2067.
37. Liu WK, Jun S, Zhang YF. Reproducing kernel particle methods. *International Journal for Numerical Methods in Fluids* 1995; **20**:1081–1106.
38. Mohd-Yusof J. Combined immersed boundaries/b-splines methods for simulations of flows in complex geometries. *Ctr Annual Research Briefs*, Stanford University, NASA Ames, 1997.
39. Fadlun EA, Verzicco R, Orlandi P, Mohd-Yusof J. Combined immersed-boundary finite-difference methods for three-dimensional complex flow simulations. *Journal of Computational Physics* 2000; **161**:35–60.
40. Gilmanov A, Sotiropoulos F. A hybrid cartesian/immersed boundary method for simulating flows with 3d geometrically complex moving bodies. *Journal of Computational Physics* 2005; **207**:457–492.
41. Gilmanov A, Sotiropoulos F, Balaras E. A general reconstruction algorithm for simulating flows with complex 3D immersed boundaries on Cartesian grids. *Journal of Computational Physics* 2003; **191**:660–669.
42. Schreyer HL, Sulsky D, Zhou S-J. Modeling delamination as a strong discontinuity with the material point method. *International Journal for Numerical Methods in Engineering* 2002; **191**:2483–2507.
43. Chen Z, Brannon R. An evaluation of the material point method. *Technical Report 2002-0482*, SAND REPORT, Sandia National Laboratory, 2002.
44. Guilkey JE, Weiss JA. Implicit time integration for material point method: quantitive and algorithmic comparison with the finite element method. *International Journal for Numerical Methods in Engineering* 2003; **57**:1323–1338.

DOI: 10.1002/fld

45. York II AR, Sulsky D, Schreyer HL. Fluid–membrane interaction based on the material point method. *International Journal for Numerical Methods in Engineering* 2000; **48**:901–924.
46. Bathe K-J. *Finite Element Procedure*. Prentice-Hall: Englewood Cliffs, NJ, 1996.
47. Egleton CD, Popel AS. Large deformation of red cell ghost in a simple shear flow. *Physics of Fluids* 1998; **10**(8):1834–1845.
48. Sotiropoulos F, Constantinescu G. Pressure-based residual smoothing operators for multistage pseudo-compressibility algorithms. *Journal of Computational Physics* 1997; **133**(1):129–145.
49. Sotiropoulos F, Abdallah S. A primitive variable method for the solution of external, 3-D incompressible, viscous flows. *Journal of Computational Physics* 1992; **103**(2):336–349.
50. Merkle CL, Athavale M. Time-accurate unsteady incompressible flow algorithm based on artificial compressibility. *AIAA Paper*, *87-1137*, 1987.
51. Burgers D, Sulsky D, Brackbill JU. Mass matrix formulation of the flip particle-in-cell method. *Journal of Computational Physics* 1992; **103**:1–15.
52. York II AR, Sulsky D, Schreyer HL. The material point method for simulation of thin membranes. *International Journal for Numerical Methods in Engineering* 1999; **44**:1429–1456.
53. Green AE, Adkins JE. *Large Elastic Deformation and Non-linear Continuum Mechanics*. Clarendon Press: Oxford, 1960.
54. Navot Y. Elastic membranes in viscous shear flow. *Physics of Fluids* 1998; **10**(8):1819–1833.
55. ten Cate A, Nieuwstad CH, Derksen JJ, Van den Akken HEA. Particle imaging velocimetry experiments and lattice-Boltzmann simulation on a single sphere setting under gravity. *Physics of Fluids* 2002; **14**:4012–4025.
56. Thompson KW. Time-dependent boundary conditions for hyperbolic systems. ii. *Journal of Computational Physics* 1990; **89**(2):439–461.
57. Pozrikidis C. Finite deformation of liquid capsules enclosed by elastic membranes in simple shear flow. *Journal of Fluid Mechanics* 1995; **297**:123–152.
58. Gilmanov A, Acharya S. A fluid/structure interaction method for compliant biological tissues: application to simulate iris accommodation. *Seventh World Congress on Computational Mechanics*, Los Angeles, 16–22 July 2006.
59. Gilmanov A, Acharya S. An immersed boundary and material point methodologies for moving/compliant surfaces with heat transfer. *Proceedings of ASME International Mechanical Congress and Exposition*, Chicago, IL, 5–10 November 2006.

DOI: 10.1002/fld