11-13-2009

An Unstructured Finite Volume Method For Incompressible Flows with Complex Immersed Boundaries

Lin Sun
Purdue University - Main Campus, sun33@purdue.edu

Sanjay Mathur
Purdue University - Main Campus, smathur@purdue.edu

Jayathi Y. Murthy
School of Mechanical Engineering, Purdue University, jmurthy@purdue.edu

Proceedings of the ASME 2009 International Mechanical Engineering Congress & Exposition
IMECE2009
November 13-19, Lake Buena Vista, Florida, USA

IMECE2009-12917

AN UNSTRUCTURED FINITE VOLUME METHOD FOR INCOMPRESSIBLE FLOWS WITH COMPLEX IMMERSED BOUNDARIES

Lin Sun[1], Sanjay R. Mathur[2] and Jayathi Y. Murthy[2]
NNSA PRISM: Center for Prediction of Reliability, Integrity and Survivabiity of Microsystems
and
[1]Rosen Center for Advanced Computing
[2]School of Mechanical Engineering
Purdue University
West Lafayette, IN 47907
USA

ABSTRACT

A numerical method is developed for solving the 3D, unsteady, incompressible flows with immersed moving solids of arbitrary geometrical complexity. A co-located (non-staggered) finite volume method is employed to solve the Navier-Stokes governing equations for flow region using arbitrary convex polyhedral meshes. The solid region is represented by a set of material points with known position and velocity. Faces in the flow region located in the immediate vicinity of the solid body are marked as immersed boundary (IB) faces. At every instant in time, the influence of the body on the flow is accounted for by reconstructing implicitly the velocity the IB faces from a stencil of fluid cells and solid material points. Specific numerical issues related to the non-staggered formulation are addressed, including the specification of face mass fluxes, and corrections to the continuity equation to ensure overall mass balance. Incorporation of this immersed boundary technique within the framework of the SIMPLE algorithm is described. Canonical test cases of laminar flow around stationary and moving spheres and cylinders are used to verify the implementation. Mesh convergence tests are carried out. The simulation results are shown to agree well with experiments for the case of micro-cantilevers vibrating in a viscous fluid.

INTRODUCTION

Over the last two decades, unstructured mesh methods have become the default for the solution of industrial fluid flow problems because of their versatility in handling complex geometries. Fluid-structure interaction problems an important area of application, and address areas as diverse as in-cylinder combustion, flexible-wing aerodynamics, biological flows, micro-electro-mechanical systems (MEMS), among others. In many of these applications, interactions between the fluid and structure may cause extreme deformation and displacement, which may, in turn, significantly change the fluid field. The challenge is to develop accurate and stable numerical methods to address this class of large-displacement problems.

Among the most widely-used techniques for moving-body and fluid-structure interaction problems is the class of Arbitrary Lagrangian-Eulerian (ALE) methods whereby the fluid is treated in an Eulerian framework, but the motion of the mesh (and the solid) is treated in a Lagrangian one; the mesh is deformed as the body moves and deforms [1,2]. The advantage of the technique is that the fluid-solid interface is crisply-defined and the implementation of interface boundary conditions is straightforward. However, extreme mesh deformation requires either partial or complete re-meshing, with attendant algorithmic complexity and errors due to mesh skewness and interpolation between meshes.

Fixed-grid methods provide an antidote to re-meshing. In these approaches, the fluid flow is computed on a background mesh, and the solid body is moved across it. A variety of techniques have been developed to capture fluid-solid interaction. In the volume of fluid (VOF) technique [3], for example, the solid (or a secondary fluid) is represented by a volume fraction or VOF which is unity in the solid and zero in the fluid. The VOF is tracked through an advection equation, and a single volume-averaged velocity field is computed; the discretization requires the specification of volume and phase area fractions, which are found through surface reconstruction [4]. The level set method [5] tracks the interface implicitly

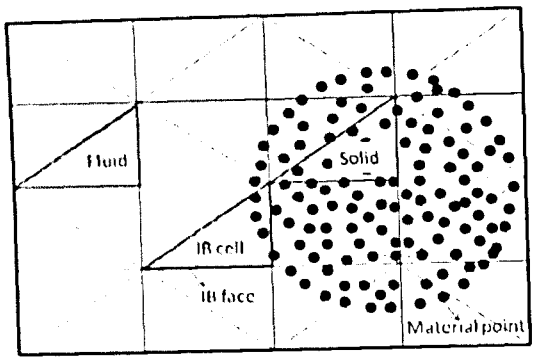

Fig 1: Definition of *fluid*, *solid* and *immersed boundary* (IB) cells and faces.

through the zero level of the level set function. Though these types of fixed grid methods are easy to implement and can handle complex interface evolution, they lose surface definition over time, and interface boundary conditions are difficult to implement. The cut-cell method [6] avoids interface smearing by considering the actual intersection between the interface and the cells through which it passes. Grid cells adjacent to the cells containing the interface are then modified to make the interface a face in the mesh. Though these practices avoid smearing, they engender significant geometric complexity.

The immersed boundary method (IBM) has found increasing use in recent years; a recent review of the field has been given in [7]. IBM was originally developed by Peskin [8] to simulate flow through heart valves. In the original formulation, the deformable valve wall was represented as a set of nodal forces which were incorporated in the fluid momentum equations as line forces. A smeared delta function was used to represent them, resulting in a fuzzy interface spread over several cells. A number of variants of the method have since been developed [7]. In recent work, Mohd-Yusof [9] and Fadlun et al [10] dispensed with the idea of incorporating equivalent forces in the fluid momentum equation to represent the action of the interface. Instead, fluid nodes closest to the interface are identified, and a velocity interpolated from the interface and an appropriate interior fluid neighbor is imposed on these near-interface nodes. More recently, Gilmanov and Sotiropolous [11] represented the interface using triangular unstructured meshes to facilitate the identification of a sharp interface. This idea was combined with the material point method to solve for the solid stress and deformation field in [12], coupled to an IBM treatment of fluid-structure interaction.

In nearly all the work cited above, the underlying fluid mesh is Cartesian; the presumption is that the only reason for geometric complexity is the solid, whether stationary, moving or deforming. In general purpose applications, however, there is interest in supporting a fully-unstructured fluid flow computation for a variety of reasons. The fluid flow itself may require local mesh adaption due to shocks or boundary layers, and this is best facilitated by unstructured meshes. Furthermore, the industrial user frequently requires versatility in the underlying solver. He or she may use boundary-conforming unstructured solution-adaptive meshes where convenient and suitable, and IBM-like formulations when fluid flow in the presence of large-scale solid motion and deformation is involved. It is therefore desirable to have these features integrated in a single numerical formulation capable of handling all mesh types. Moreover, local mesh adaption to improve the accuracy of recent immersed boundary formulations requires the underlying flow solver to support unstructured data structures [13]. For these reasons, there is interest in generalizing IBM to arbitrary unstructured meshes. Furthermore, with the proliferation of non-staggered pressure-based methods for fluid flow, there is interest in integrating immersed boundary techniques with widely-used algorithms such as SIMPLE [14], particularly in the versatile unstructured finite volume form.

In this paper, we develop an immersed boundary formulation consistent with a cell-centered unstructured finite volume scheme used widely in flow computations [15]. The solid deformation and motion are to be computed using a material point method [16]. Consequently the baseline algorithm is developed by representing the solid as a collection of material points moving across a background unstructured mesh. Details of the development of immersed boundary treatment in a non-staggered framework are presented. The method is applied to a variety of stationary and moving body problems and show to perform satisfactorily, either by comparison to analytical results or to conformal-mesh computations.

GOVERNING EQUATIONS

The equations governing the incompressible flow of a Newtonian fluid are given by:

$$\nabla\cdot(\rho V)=0$$

$$\frac{\partial}{\partial t}(\rho V)+\nabla\cdot(\rho VV)=-\nabla p+\nabla\cdot\tau+f_b \qquad (1)$$

with boundary conditions

$$V=V_b(r,t) \qquad \text{on } \Gamma_b$$

Here τ is the Newtonian stress tensor. The overall domain containing the fluid and solid is denoted by Ω, while that containing the fluid is denoted by Ω_f and that containing the solid by Ω_s. The boundary demarcating the two is denoted by Γ_b. The motion of this boundary is assumed to be prescribed *a priori* in this paper through a spatial and time-dependent boundary velocity $V_b(r,t)$.

The momentum equation in (1) is recast in the form of a convection-diffusion equation as:

$$\rho\frac{\partial V}{\partial t}+\nabla\cdot(\rho VV)=-\nabla p+\nabla\cdot\mu\nabla V+f_b+S \qquad (2)$$

Here, S is the vector of source terms $\left(\nabla\cdot\tau-\nabla\cdot\mu\nabla V\right)$.

FORMULATION

Overview of Finite Volume Scheme

The basic scheme used in the formulation is that described in [15]. The computational domain is discretized into arbitrary unstructured convex polyhedra, referred to henceforth as control volumes or cells. All unknowns, including pressure, velocity and scalars, are stored at cell centroids. The momentum, continuity and scalar transport equations are integrated over the control volumes to yield cell balances of convective, diffusive, unsteady and source terms. Second-order discretization operators are then applied to convert these cell balances into nominally-linear algebraic equations which are solved using an algebraic multigrid scheme. A collocated formulation is employed for pressure and velocity, similar to that of Rhie and Chow[17]. The SIMPLE algorithm [14] is employed for pressure-velocity coupling. For unsteady situations, a second-order implicit formulation is employed.

Determination of Immersed Boundary Faces and Velocities

The fluid flow is solved on background unstructured polyhedral mesh. The solid body is represented by a collection of material points whose motion is computed using the Material Point method (MPM) [16]. The position of the material points $\mathbf{r}_s(t)$ and their velocities $\mathbf{V}_s(\mathbf{r}_s, t)$ are assumed known at each time instant. The material points (MP) are located on the background unstructured mesh using an octree search [18]. All cells which contain no material points are labeled *fluid* cells. All cells containing at least one material point, and at least one of whose face neighbors is a *fluid* cell are labeled *immersed boundary (IB)* cells. The rest are labeled *solid* cells; solid cells contain at least one material point, and are surrounded by face neighbors which also contain at least one material point. Furthermore, all faces shared by IB cells and fluid cells are marked as IB faces. These markings are illustrated in Fig. 1.

Once the cells and faces are marked, the next step is to interpolate from the set of fluid and material point velocities to obtain the interpolated velocity that will serve as the boundary condition on the fluid cells. In a co-located framework, there are two choices: we may interpolate either the cell-centered velocity of the IB cell, or the face-centered velocity of the IB face. In a finite volume scheme, boundary conditions are posed most easily on faces. Therefore, in this paper, we interpolate from a neighborhood of fluid-cell and material point velocities to obtain the velocity vector at IB faces.

In the current implementation, a sampling radius is prescribed, typically based on the length scale of the local IB cell. A neighborhood of fluid and IB cells around the centroid of the IB face falling within the sampling radius is chosen. A linear least-squares interpolation of the material point and fluid cell-centered velocities within this neighborhood is performed, and the IB face velocity $\mathbf{V}_b$ in Fig. 2 determined. From this IB velocity vector are computed the mass flow rate into the fluid

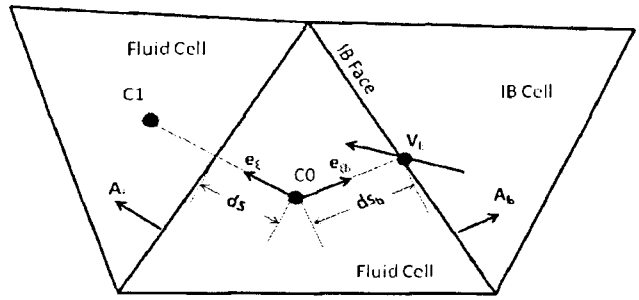

Fig. 2: Nomenclature associated with fluid cell C0 and C1 and IB face for momentum and mass balances.

cell neighboring the IB face, as well as the values of the velocity components convected by the mass flow rate.

Discretization of Momentum Equation

The treatment of interior fluid cells is that developed in [15] and is described briefly here for completeness. Then the modifications made to admit immersed boundaries are described. In keeping with a non-staggered mesh formulation, velocity and pressure are stored at the cell centroid. Pressure is also stored at all cell faces for convenience.

Consider the fluid cell shown in Fig.2, which has at least one IB face. The momentum equation (Eq. (2)) is integrated over the cell C0, and the divergence theorem applied. This yields a balance of the unsteady, convective, diffusive and source terms which may be written as:

$$\frac{(\rho V)_0^{n+1}-(\rho V)_0^{n}}{\Delta t}=$$

$$\left(\sum_{faces\ f}\dot{m}_f V_f+\sum_{faces\ f}\boldsymbol{D}_f-\sum_{faces\ f}p_f\boldsymbol{A}_f+\left(\boldsymbol{f}_b+\boldsymbol{S}\right)\Delta V_0\right)^{n+1} \tag{3}$$

Here, $\boldsymbol{V}_0$ is the velocity vector stored at the centroid of cell C0. All quantities superscripted n+1 are values at the current time level, while those superscripted n are those at the previous time. A fully-implicit scheme is employed whereby the left hand side of Eq. (3) is evaluated at the current time.

Treatment of Interior Faces. We consider first the treatment of interior faces, such as that between two fluid cells C0 and C1 in Fig. 2. In Eq. (3), $\dot{m}_f$ is the mass flow rate at the face f entering the cell C0, and $\boldsymbol{V}_f$ is the velocity vector at the face. At interior faces, the face velocity vector is determined using a second-order accurate upwinding scheme [15].

The diffusion term $\boldsymbol{D}_f$ at an interior face is decomposed into its primary and secondary components, as described in [15], and written as:

$$D_f = \frac{A_f \cdot A_f}{A_f \cdot e_\xi} \frac{(V_1 - V_0)}{ds} + S_f \quad (4)$$

Here, A_f is the outward-pointing area vector at face f, and e_ξ is the unit vector aligned with the line connecting the centroids of the two cells on either side of the face, as shown in Fig. 2. The quantity S_f is the secondary gradient term, described in [15] and is zero for an orthogonal mesh.

The pressure gradient term at interior faces requires the determination of the face pressure p_f. It is found by linear interpolation from the cell-centered values, in keeping with the non-staggered formulation described in [15]. The last two terms in Eq. (3) are evaluated at the cell centroid using velocity gradient evaluations obtained through linear least-squares reconstruction [19] of the velocity field.

To prevent checkerboarding, the face mass flow rate $\dot{m}_f$ at interior faces is found using an added dissipation scheme similar to that of Rhie and Chow [17] and described in [15]. For the face f between cells C0 and C1 in Fig. 2, the face mass flow rate is written in terms of a face normal velocity $\tilde{V}_{f,n}$ pointing outward from cell C0, and defined as:

$$\tilde{V}_{f,n} = V_{linear,n} + d_f \left(\bar{\nabla} p \cdot e_\xi - \frac{(p_1 - p_0)}{ds} \right) \quad (5)$$

Here, $V_{linear,n}$ is the linearly interpolated face-normal velocity computed from the cell-centroid velocity vectors of cells C0 and C1. The second term is the added dissipation term; the coefficient d_f is found through momentum interpolation [17] and is described in [15]. The term $\bar{\nabla} p$ is the average pressure gradient at the face, computed from the cell-centered pressure gradients in cells C0 and C1. The face mass flow rate $\dot{m}_f$ into the fluid cell is computed as:

$$\dot{m}_f = -\rho_f \tilde{V}_{f,n} A_f \quad (6)$$

Treatment of IB Faces. We now consider the treatment of the IB face shown in Fig. 2. The convective momentum flux in Eq. (3) is modified to use the linearly-interpolated IB velocity V_b in Fig. 2 rather than a second-order upwinded value V_f. Similarly, the face mass flux on the IB face is computed as:

$$\dot{m}_b = -\rho_b V_b \cdot A_b \quad (7)$$

Here, $\dot{m}_b$ and ρ_b are the mass flow rate and fluid density at the IB face centroid and A_b is the area vector pointing from the fluid to the IB cell in Fig.2.

The diffusive term in Eq. (4) is modified as:

$$\frac{A_b \cdot A_b}{A_b \cdot e_{\xi b}} \frac{(V_b - V_0)}{ds_b} + S_b \quad (8)$$

For the evaluation of pressure gradient term, the face pressure on the IB face is found by linear reconstruction, using the cell pressure p_0 and the pressure gradient ∇p_0 in the C0 cell as:

$$p_b = p_0 + \nabla p_0 \cdot ds_b e_{\xi b} \quad (9)$$

Treatment of IB and Solid Cells. Fluid momentum equations are not solved in IB and solid cells, and the fluid velocity vector and pressure in these cells is never used.

Discretization of Continuity Equation

The treatment of fluid cells follows the formulation described in [15]. For cell C0 in Fig. 2, the continuity equation is integrated over the control volume, yielding a balance of mass flow rates. In a co-located formulation, the discrete continuity equation is written in terms of the face normal velocity defined in Eqns. (5) and (6) as:

$$\sum_{faces\ f} \dot{m}_f + S_m \Delta V_0 = -\sum_{faces\ f} \rho_f \tilde{V}_{f,n} A_f + S_m \Delta V_0 = 0 \quad (10)$$

Eq. (10) is written only for fluid cells in the domain; no continuity equation is solved in IB and solid cells. For fluid cells with IB faces, Eq. (7) replaces Eq. (6) in writing the face mass flow rate expression in Eq. (10).

The S_m term in the above equation is the volumetric mass source in the cell, and is normally zero. In the immersed boundary formulation for an incompressible flow, it is required that:

$$\sum_{\substack{IB\ \& \\ \text{boundary } faces}} \dot{m}_f = R_{m,total} = 0 \quad (11)$$

However, numerical interpolation errors in finding the IB face velocity may lead to Eq. (11) not being satisfied. A number of remedies for this problem have been proposed in the literature [20,21]. In the present work, the net mass imbalance $R_{m,total}$ is distributed in a volume-weighted fashion in the fluid cells in the domain. Thus, the corrective mass source term is computed as

$$S_m \Delta V_0 = R_{m,total} \frac{\Delta V_0}{\sum_{fluid\ cells} \Delta V} \quad (12)$$

In all computations performed in this paper, the normalized mass imbalance over the domain is found to be less than 0.01% at convergence.

PRESSURE CORRECTION EQUATION

The SIMPLE algorithm is used to the momentum and continuity equations in the fluid domain. In keeping with the procedure described in [15], the face normal velocity field after the solution of the momentum equations, $\tilde{V}^*_{f,n}$, does not satisfy discrete mass balance. Therefore, a normal velocity correction corresponding to Eq. (5) is applied, and is defined as:

$$\tilde{V}_{f,n} = \tilde{V}^{*}_{f,n} + \tilde{V}'_{f,n}$$
$$\tilde{V}'_{f,n} = d_f \frac{(p'_1 - p'_0)}{ds} \quad (13)$$

The IB face velocities are not expanded in term of a pressure correction, but are included as nominally-known mass flow rates in the continuity balance. Substituting Eqns. (6), (7) and (13) in Eq. (10) yields the pressure-correction for each fluid cell *i* in the domain:

$$a_{ii} p'_i = \sum_{j=1}^{nb} a_{ij} p'_j + b_i \quad (14)$$

Here, *j* is the index of the cell sharing a face with cell *i*; *nb* is the total number of such cells. The term b_i contains the mass imbalance in cell *i* [15]. The mass flow rates corresponding to the IB face velocities, as well as the corrective source term (Eq. 12) therefore appear in b_i.

In keeping with the SIMPLE procedure, once the pressure correction equation, Eq. (14), is solved, the cell pressure, and face mass flow rate are corrected as:

$$p_0 = p^{*}_0 + \alpha_p \frac{d_f}{ds} p'_0$$
$$\dot{m}_f = \dot{m}^{*}_f - \rho_f A_f d_f \frac{(p'_1 - p'_0)}{ds} \quad (15)$$

Here, starred values represent quantities based on the solution of the momentum equations, and α_p is the pressure under-relaxation coefficient [14]. In addition to the corrections applied in Eq. (15), the cell-centered velocity vector is also corrected in the manner described in [15] to promote convergence. IB face velocities are not corrected after the solution of the pressure correction equations.

OVERALL SOLUTION PROCEDURE

The overall solution procedure for an unsteady computation consists of the following steps.

1. Establish the state at time level *n*. Initialize solution variables, including cell-centered velocities, face mass flow rates and cell and face pressures. Initialize the positions and velocities of material points.
2. Establish the state of material points at time level *n+1* by moving material points over the time step Δt based on the material point velocity vector at time *n*. Update material point velocities at time level *n+1* based on the prescribed motion of the solid.
3. Locate the positions of the material points on the background unstructured mesh. Mark cells as fluid, solid or IB, and cell faces as interior, IB or boundary. Compute geometric interpolation factors at *n+1* using linear least-squares relating IB face velocities to the material point and fluid cell-centered velocities.
4. If it is the first time step, copy the fluid velocities, face mass flow rates and cell pressures from time level *n* to time level *n+1* to establish the initial guess at *n+1*.
5. Compute IB face velocities from prevailing solid and fluid cell-centered velocities.
6. Discretize and solve the fluid momentum equations in the fluid cells.
7. Discretize and solve the pressure correction equations in the fluid cells.
8. Correct face mass flow rates, cell pressures and cell-centered velocities.
9. Check for convergence within time step n+1. If unconverged, go to step 5. If converged, go to step 10.
10. Check if the desired number of time steps have been completed. If incomplete, copy the state at level *n+1* to level *n* and go to step 2. If complete, stop.

RESULTS

In this section, we demonstrate the immersed boundary scheme on a variety of problems to verify the performance of the algorithm and software and to validate its performance against experiments. Our interest is in computing steady and unsteady flows in microscale applications, and consequently, a number of low Re problems are considered here.

Verification Test 1: Flow Over Cylinder in Driven Cavity

The lid-driven cavity problem has long been used a test case for new codes and new solution methods. The standard problem consists of a fluid contained in a square domain of side L with three stationary sides; the top wall is imparted a tangential velocity U, as shown in Fig.3. A variant of this problem is computed here. A stationary cylinder of radius R is inserted at the center of the cavity, and is represented by material points. The interaction between fluid and cylinder is treated by the immersed boundary scheme described here and the fluid flow is computed using the finite volume scheme. A steady 2D isothermal incompressible flow is assumed, with Re=1.0 and *R/L*=5.0

The problem is first solved using the commercial CFD tool FLUENT with a body-fitted triangular mesh of 1988cells. Testing with finer meshes established that the velocity profile was mesh invariant to 0.5 percent. Then the same problem is solved by the immersed boundary method with four different meshes: a 50×50 and a 100×100 uniform Cartesian mesh, and two triangular unstructured meshes of 1988 and 7952 cells respectively.

Velocity vectors for Re= UL/v=1.0 are shown in Fig. 3 for the body-fitted mesh case, and indicate the formation of a primary recirculation driven by the top wall, as well as a secondary recirculation due to the presence of the cylinder. These features were also captured by the IBM solution.

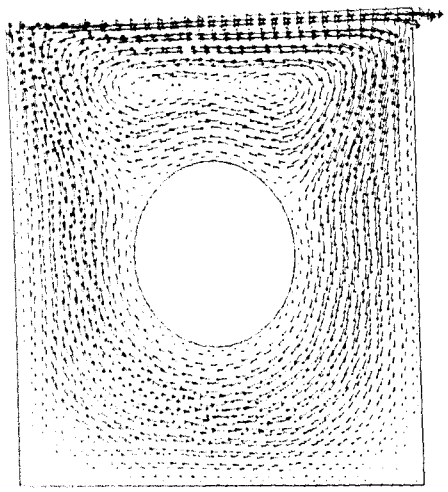

Fig. 3: Velocity vectors at Re=1.0 computed using a body-fitted triangular mesh.

Fig. 4 and Fig. 5 show the non-dimensional velocity profile at the horizontal and vertical mid-planes. The length is scaled by the diamerter of the cylinder D and the velocity is scaled by the top wall velocity U.

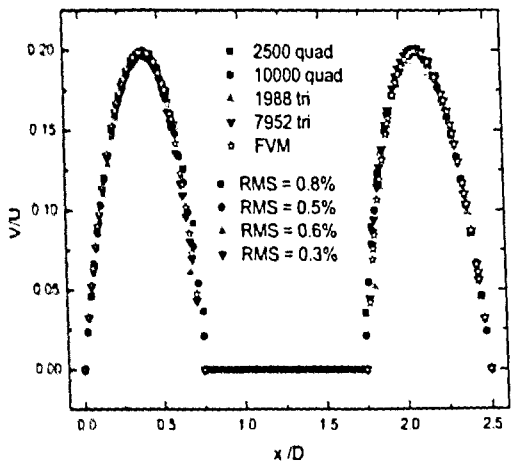

Fig. 4: Velocity profile at the horizontal mid-plane at Re = 1.0

To estimate the error between FLUENT and IBM results, we interpolate the IBM data points to the FLUENT mesh locations and calculated the root mean square(RMS) error as

$$RMS = \sqrt{\frac{\sum_{i=1}^{N} \left(\frac{V_{IBM,i} - V_{Fluent,i}}{V_{ref}}\right)^2}{N}} \qquad (16)$$

where V_{IBM} and V_{Fluent} are the velocities calculated from IBM and FLUENT respectively. V_{ref} is a reference value to normalize the error. In this case, the reference value takes the velocity of the top wall (U). The calculated RMS error for each mesh are inserted in Fig. 4 and 5. It is seen that the IBM results match FLUENT results well and generally the error is about or less than 1.0%. Larger deviationa are seen for coarse meshes because the cylinder surface is not well-resolved by the coarse Cartesian mesh and this difference is signicantly reduced by refining the mesh. This cavity cylinder problem has also been run at Re=0.1 and Re=100 and the results (not shown here) demonstrate similar accuracy.

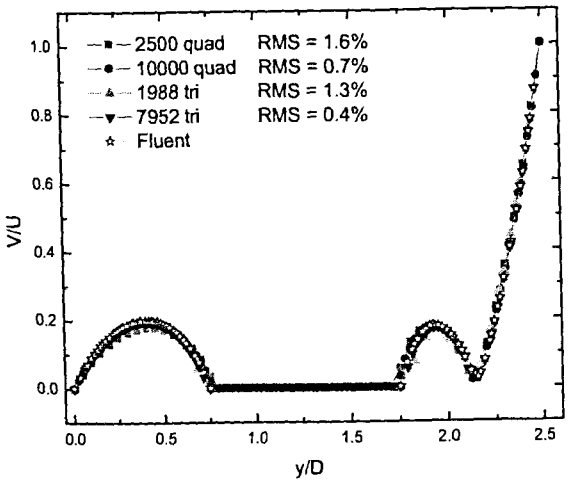

Fig. 5: Velocity profile on the vertical mid-plane at Re = 1.0

Verification Test 2: Flow Over Stationary and Rotating Cylinders

Another fundamental fluid mechanics problem computed is laminar flow past a cylinder, as shown in Fig. 6. The incoming velocity profile is uniform, with a velocity U; the cylinder diameter is D. At low Re (=UD/v), the flow is symmetric about the horixontal md-plane. As the Reynolds number increases, flow begins to separate behind the cylinder causing vortex shedding. In this work, we compute low Reynolds flow (Re<10) where no vortex shedding occurs.

Fig. 6 shows the non-uniform cartesian mesh employed in the immersed boundary method. A flow domain is created surrounding the cylinder. The upstream length is 5D and the downstream section is 20D. The width of the flow domain is 10D. The inlet condition is velocity inlet with uniform profile. The outlet is prescribed to be at a uniform static pressure. The top and bottom surfaces are stationary non-slip walls.

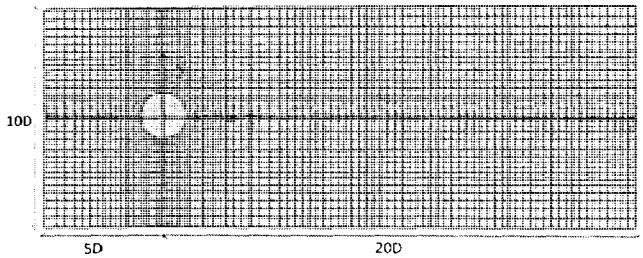

Fig. 6: Non-uniform cartesian mesh for flow past a cylinder using immersed boundary method.

First we simulate a steady flow past a stationary cylinder immersed in an bounded uniform flow at Re=1.0. The problem is first solved using FLUENT with a body-fitted mesh of 4356 triangular cells. It is then solved by the immersed boundary method using a Cartesian mesh of 5000 cells, as shown in Fig. 6.

Fig. 7 (left) shows the stream function for the flow over a stationary cylinder. Fig. 8 shows the velocity comparison between FLUENT and IBM results on different vertical lines in the domain. The velocity is scaled by inlet velocity U and the length is scale by cylinder diameter D. Using the inlet velocity U as reference, the RMS error between the FLUENT and the IBM results is around 1%. Fig. 9 shows the velocity comparison at the horizontal centerline of the cylinder. The RMS error is less than 1%.

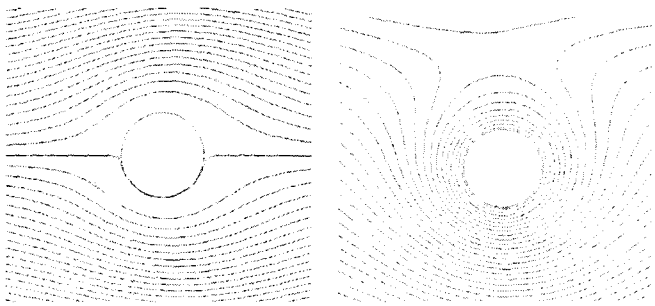

Fig. 7: Stream function for flow past a stationary (left) and rotating (right) cylinder.

Next, we simulate the flow past a circular cylinder rotating about its axis at a rotational speed ω, corresponding to a roational Reynolds number $Re_\omega=(\omega D^2/\nu)=10$; in addition to the rotational motion, there is an incoming flow with a Reynolds number Re=1.0, as before. The stream function for this case is shown in Fig. 7 (right). Similar to stationary case, the velocity comparison with respect to the body-fitted mesh case is illustrated in Figs. 10 and 11 on multiple vertical and horizontal locations. A reference velocity $V_{ref}=5U$ is used in the estimate of RMS error and the error is less than 1%.

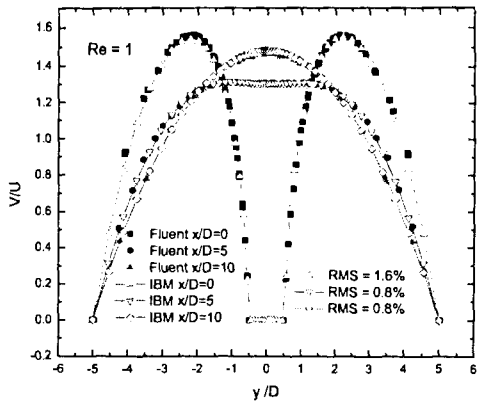

Fig. 8: Velocity profile on different vertical lines in the domain for flow over a stationary cylinder.

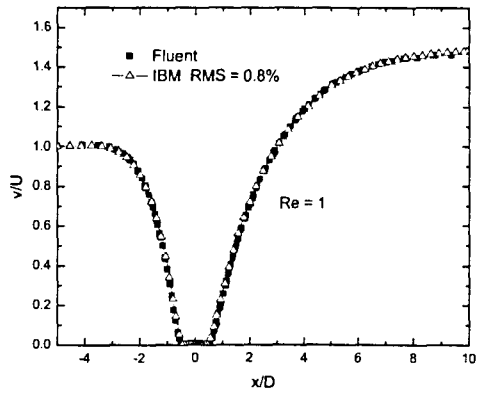

Fig. 9: Velocity profile on horizontal center line for flow over a stationary cylinder.

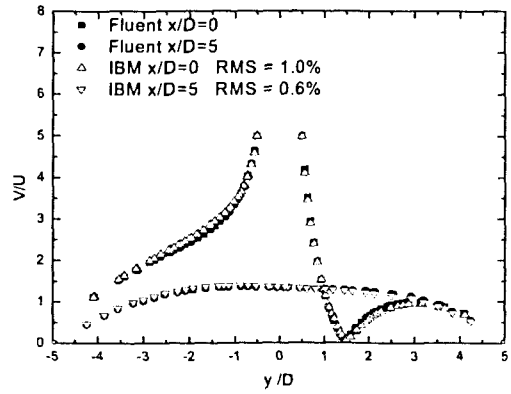

Fig. 10: Velocity profiles on different vertical lines in domain for flow over a rotating cylinder.

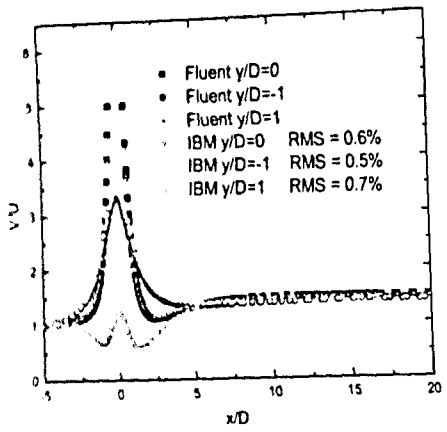

Fig. 11: Velocity profile on horizontal lines in domain for flow over a rotating cylinder.

Verification Test 3: Stokes Flow Over Sphere

The last verification test is the Stokes flow past a stationary sphere. The well-known analytical solution to this problem [22] is given in spherical coordinates

$$u_r = U\cos\theta\left[1-\frac{3}{4}\frac{D}{r}+\frac{1}{16}\frac{D^3}{r^3}\right]$$
$$u_\theta = -U\sin\theta\left[1-\frac{3}{8}\frac{D}{r}-\frac{1}{32}\frac{D^3}{r^3}\right] \quad (17)$$

The corresponding drag force is given by

$$F_D = 3\pi\mu DU \quad (18)$$

Here, D is the sphere diameter, and r is the distance from a point to the origin. θ is the azimuthal angle in the x-y plane from the x axis with $0 \le \theta < 2\pi$. U is the uniform free stream velocity. μ is fluid viscosity. In this study, computations are performed for Re (=UD/v) =0.02.

One of the difficulties encountered in solving this problem is the proper specification of outer boundary conditions. Because of the vanishingly-small Reynolds number, very large domains are necessary to capture the effect of an unconstrained free-stream flow. To avoid this difficulty, an alternative problem specification is chosen. A relatively small domain of size 5D about the sphere is chosen, and the exact analytical velocity solution is imposed at the boundaries. Mesh dependence studies are carried out using three uniform Cartesian meshes: $30\times30\times30$, $50\times50\times50$ and $100\times100\times100$. The sphere occupies 20% of the domain in any given coordinate direction; thus the approximate cube occupied by the sphere is $6\times6\times6$, $10\times10\times10$, $20\times20\times20$.

The velocity profile computed from IBM using the 100x100x100 mesh is shown in Fig. 12 and is compared to analytical solution (Eqn. 16). The RMS error is less than 1%. The drag forces calculated by IBM using the three meshes are shown in Table 1. The analytical drag force is calculated by Eqn. 17. The error of IBM result with respect to the analytical value is also shown in the table.

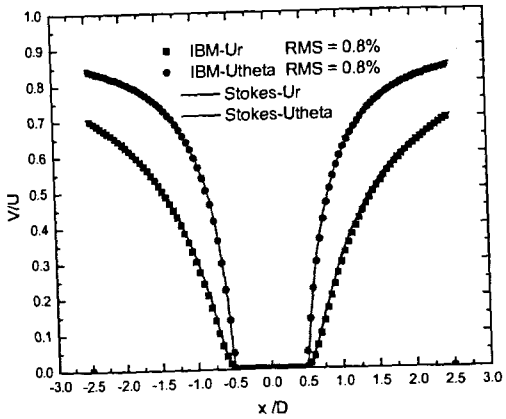

Fig. 12. Velocity profile for flow over a sphere.

Table 1. Mesh dependence of drag force

Mesh size	27,000	125,000	1,000,000
$F_D/(0.5\rho U^2D^2)$	835	870	905
Error with respect to analytical solution	11%	7.4%	3.7%

It can be seen that an error of 3.7% persists in the drag force even on a 100x100x100 mesh. Closer examination of the results reveals that the deviation is primarily due to the lack of adequate geometric resolution of the sphere, as seen in the Fig. 13.

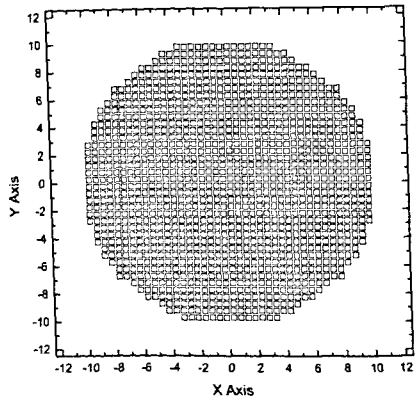

Fig. 13. Projection of IB cells on x-y plane for $100\times100\times100$ cell mesh for flow over sphere.

The above three verification problems indicate that the immersed boundary method developed here demonstrates good accuracy for the range of problems computed. The error of the IBM is found to be mainly related to how well the real solid surface is approximated by the IB cells/faces. This type of error can be significantly reduced by locally refining the mesh; this type of hanging-node refinement is accommodated straightforwardly in our unstructured mesh framework.

Validation Test: Flow Around Freely-Vibrating Cantilever

To validate the immersed boundary method, we consider the vibration of a microcantilever in a fluid, as shown in Fig. 14. Our objective is to predict the damping coefficient as a function of the ambient pressure assuming that the continuum limit is valid. Experimentally-measured damping coefficients for ambient pressures ranging from atmospheric to near-vaccuum are available in [23]. In the computations performed here, the motion of the material points is prescribed a priori; the fluid flow and the resulting damping coefficient are computed using IBM.

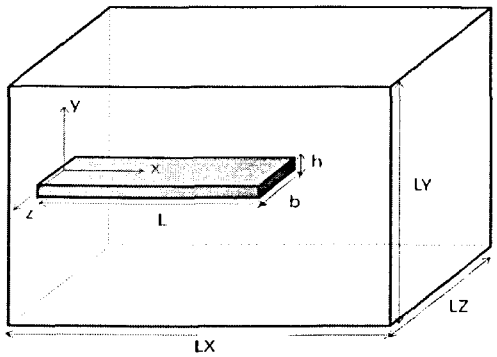

Fig. 14. An illustration of the vibrating cantilever system.

Damping Ratio. As the cantilever vibrates, it experiences a damping due to flow around it. The quality factor or Q factor is defined in terms of the ratio of the energy stored in the system to that of the energy dissipated in one cycle:

$$Q = 2\pi \times \frac{\text{Energy stored}}{\text{Energy dissipated per cycle}} \tag{19}$$

Higher Q indicates a lower rate of energy loss relative to the stored energy of the oscillator; consequently, the oscillations die out more slowly. The damping ratio, indicated by ζ, is related to the Q factor by

$$\zeta = \frac{1}{2Q} \tag{20}$$

Here the derivation of the damping ratio for cantilever is briefly described, and is taken from **[Error! Bookmark not defined.]**. We then describe the post-processing technique used to obtain the damping ratio.

For a one-degree-freedom damped harmonic oscillator as shown in Fig. 15, the governing ordinary differential equation is

$$m\ddot{x}(t) + c\dot{x}(t) + kx(t) = 0 \tag{21}$$

where m is mass, c is damping coefficient, and k is spring constant.

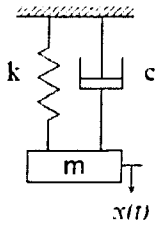

Fig. 15: An illustration of a one-degree-freedom harmonic vibrating and damped system.

We define the natural frequency of the harmonic oscillator as $\omega_0 = \sqrt{k/m}$ and the damping ratio $\zeta = \frac{c}{2\sqrt{km}} = \frac{c}{2m\omega_0}$. Assuming that the velocity takes the form $\dot{x}(t) = V \cdot e^{j\omega t}$ and substituting into Eq. (20) yields

$$j\omega m e^{j\omega t} + k/(j\omega) \cdot e^{j\omega t} + c e^{j\omega t} = 0 \tag{22}$$

Solving the above ODE yields the frequency of the oscillator as a function of the damping ratio:

$$\omega = \omega_0(\zeta \pm \sqrt{\zeta^2 - 1}) \tag{23}$$

We wish to define a damping ratio similar to that for the 1D oscillator case. According to Euler-Bernoulli beam theory, the governing equation for a 2D cantilever is

$$\rho_s h \frac{\partial^2 w(x,y,t)}{\partial t^2} + EI\nabla^4 w(x,y,t) = p(x,y,t) \tag{24}$$

where ρ_s is the cantilever density and h is the beam thickness. E is the elastic modulus and I is the second moment of area, w is the displacement and p is force per unit area on the cantilever.

We assume that the cantilever velocity takes the form

$$\dot{w}(x,y,t) = A(x,y)e^{j\omega t} \tag{25}$$

where $A(x,y)$ is the mode shape. The force is written in the complex form

$$p(x,y,t) = [p_R(x,y) + j \cdot p_I(x,y)]e^{j\omega t} \tag{26}$$

Substituting Eq. (25) into Eq. (13) yields:

$$\begin{aligned} &j\omega[\rho h A(x,y) - p_I(x,y)/\omega]e^{j\omega t} + \\ &1/(j\omega) \cdot D_e \nabla^4 A(x,y)e^{j\omega t} = p_R(x,y)e^{j\omega t} \end{aligned} \tag{27}$$

Re-arrangement of Eq. (26) yields a form similar to Eq. (21)

$$j\omega M_{eq} e^{j\omega t} + K_{eq}/(j\omega) \cdot e^{j\omega t} + C_{eq} e^{j\omega t} = 0 \tag{28}$$

with

$$M_{eq} = \iint A(x,y)[\rho h A(x,y) - p_I(x,y)/\omega]dxdy \quad (29)$$

$$C_{eq} = -\iint A(x,y)p_R(x,y)dxdy \quad (30)$$

$$\zeta = \frac{C_{eq}}{2M_{eq}\omega_r} \quad (31)$$

The resonant frequency of the cantilever ω_r is given by experiments [**Error! Bookmark not defined.**].

If we neglect the mode shape variation by assuming $A(x,y)$ to be constant, the damping ratio may be simplied as

$$\zeta = \frac{-\iint p_R dxdy}{2\omega_r\left[\rho hLb - \iint (p_I/\omega)dxdy\right]} \quad (31)$$

The real and imaginary parts of the complex force are calculated by

$$p_R = \frac{1}{NT}\int_0^{NT} p(t)\times\cos(\omega_0 t)d\tau \quad (32)$$

$$p_I = \frac{1}{NT}\int_0^{NT} p(t)\times\sin(\omega_0 t)d\tau \quad (33)$$

where T is the period of the force signal and N is the number of cycles.

Problem Specification. A cantilever with width $b = 35\mu m$, thickness $h = 1.0\mu m$ and a length $L = 100\mu m$ is considered. The simulation domain is taken to be a rectangular box with dimensions $LX = 200\mu m$ $LY = 150\mu m$ and $LZ = 200\mu m$, as shown in Fig. 15. The outer boundaries are assumed to be stationary walls. The domain is filled with air at T=288K. At ambient pressure $P = 101325Pa$ and a temperature $T = 288K$, air properties take the values $\mu = 1.79\times10^{-5}kg/m\cdot s$ and $\rho = 1.225kg/m^3$. The variation of density with pressure is assumed to follow the ideal gas law. Computations are performed for ambient pressure values ranging from 1.1145x10^4 to 8.3593x10^4 Pa, corresponding to experiments described in [23].

In the simulation, the cantilever is assigned a normal velocity of $w = V_{max}\cos(\omega_0 t)$. From experimental mesurement, the amplitude of the velocity, V_{max} is small enough to keep the Reynolds number much less than unity. In keeping with this, we assume $V_{max} = 0.001m/s$ everywhere on the cantilever. The unsteady evolution of the flow is computed until a periodic steady state is reached. At this point, the resultant pressure force on the cantilever is computed each time step by integrating the fluid pressure on the IB faces enveloping the beam surface. The force signal is recorded as a function of time, and Eq. (31) is used to compute the damping coefficient. Fig. 16 shows one period of the prescribed velocity and the resultant force exerted on the cantilever.

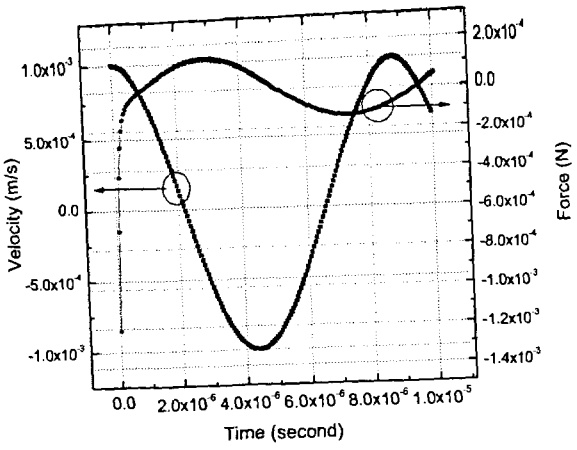

Fig. 16: One period of the prescribed velocity and the resultant force on the cantilever.

In Fig. 17, the measured quality factor for Mode 1 of cantilever vibrationis compared to predictions using the immersed boundary method. Computations using IBM are made only at pressures at which the continuum Navier-Stokes equaitons are valid. Table 2 shows the exact values computed, as well as the measured experimental values. Our computations are found to be within 5.4% of experiment. The experimental error has been estimated to be 1.0% in [23].

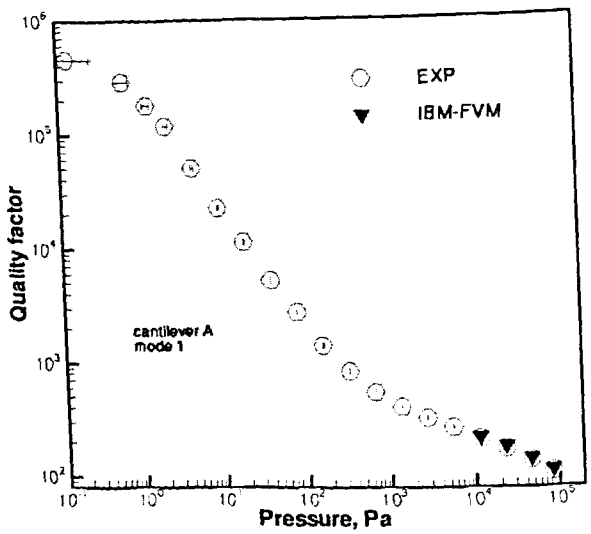

Fig. 18. Comparison of computed and measured quality factors for cantilever vibration in Mode 1.

Table 2. Quality factor comparison with experiments in [23].

Pressure (Pa)	11145	22531	45862	83593
Q (EXP)	211.86	169.46	135.87	107.76
Q (IBM-FVM)	219.42	178.62	140.19	111.84
Error	3.6%	5.4%	3.2%	3.8%

CONCLUSIONS

In this paper, an immersed boundary scheme for the computation of fluid-structure interaction has been described. The scheme employs an unstructured finite volume scheme for an incompressible flow, and a material point method for the deformation and motion of the solid. Only prescribed solid motion is considered in this initial implementation.

The computational scheme addresses issues specific to the implementation of the immersed boundary method within a non-staggered pressure-velocity framework. It is applied to a variety of fluid-structure interaction problems involving prescribed solid motion, and shown to perform satisfactorily. Research is underway to include two-way coupling between fluid and structure within this framework, and results from this effort will be reported in due course.

NOMENCLATURE

a_{ij} coefficient in discrete equation
A(x,y) mode shape
$\boldsymbol{A}$ outward-pointing area vector
b mass imbalance term in pressure correction equation
b cantilever width
c damping coefficient
C_{eq} equivalent damping coefficient
d momentum interpolation coefficient
ds centroid-to-centroid distance
$\boldsymbol{D}$ diffusion flux vector at cell face
$\mathbf{e}_\xi$ unit vector along line connecting cell centroids
E elastic module
$\mathbf{f}_b$ body force vector
F_D drag force
h cantilever thickness
I momentum
j imaginary unit
k spring constant
L cantilever length
$\dot{m}$ mass flow rate
M_{eq} equivalent mass
n time level
p pressure
p' pressure correction
Q quality factor
r distance between two points
$R_{m,total}$ total mass imbalance in fluid domain
Re Reynolds number
$\mathbf{S}$ source vector in momentum equation
$\boldsymbol{S}$ secondary gradient vector
S_m mass source in continuity equation
t time
T temperature
$\boldsymbol{V}$ velocity vector
$\boldsymbol{V}_b$ velocity vector of material point
$\tilde{V}_{f,n}$ momentum-interpolated face normal velocity
w cantilever displacement

Greek Symbols

α under-relaxation coefficient
μ fluid viscosity
ρ fluid density
$\boldsymbol{\tau}$ stress tensor
θ angle
ς damping ratio
ω frequency
ΔV cell volume

Subscripts and Superscripts

b immersed boundary face
f cell face
R real part of a complex number
I imaginary part of a complex number
0 associated with cell C0
1 associated with cell C1

ACKNOWLEDGMENTS

Support from the Department of Energy (National Nuclear Security Administration) under award no: DE-FC52-08NA28617 is gratefully acknowledged.

REFERENCES

1. T.E. Tezduyar, S. Sathe and K.Stein, "Solution techniques for the fully discretized equations in computation of fluid-structure interactions with space-time formulations," Computer Methods in Applied Mechanics and Engineering, **195**, 5743-5732, 2006.
2. I. Demirdzic and S. Muzaferija, "Numerical methods for fluid flow, heat transfer, and stress analysis using unstructured moving meshes with cells of arbitrary topology," Computer Methods in Applied Mechanics and Engineering, **125**, 235-255, 1995.
3. C.W. Hirt and B.D. Nichols, "Volume of Fluid (VOF) methods for the dynamics of free boundaries," Journal of Computational Physics, **39**, 201-225, 1981.

4. W.J. Rider and D.B. Kothe, "Reconstructing volume tracking," Journal of Computational Physics, **141**, 112-152, 1998.
5. J.A. Sethian and P. Smereka, "Level set methods for fluid interfaces," Annual Review of Fluid Mechanics, **35**, 341-372, 2003.
6. H.S. Udaykumar, R. Mittal, P. Rampunggoon and P. Khanna, "A sharp interface Cartesian grid method for simulating flows with complex moving boundaries," Journal of Computational Physics, **174**, 345-380, 2001.
7. R. Mittal and G. Iaccarino, "Immesed Boundary Methods," Annual Review of Fluid Mechanics, **37**, 239-261, 2005.
8. C.S. Peskin, "The fluid dynamics of hear valves: experimental, theoretical and computational methods," Annual Review of Fluid Mechanics, **14**, 235-259, 1981.
9. J. Mohd-Yusof, "Combined immersed boundaries/b-spline methods for simulations of flows in complex geometries, Annual Research Briefs, Center for Turbulence Research,317-328, 1997.
10. E.A. Fadlun, R. Verzicco, P. Orlandi, and J. Mohd-Yusof, "Combined immersed boundary finite-difference methods for three-dimensional complex flow situations," Journal of Computational Physics, **161**, 37-60, 2000.
11. A. Gilmanov and F. Sotiropoulos, " A hybrid Cartesian/immersed boundary method for simulating flows with 3d geometrically complex moving bodies," Journal of Computational Physics, **207**,457-492, 2005.
12. A. Gilmanov and S. Acharya, "A hybrid immersed boundary and material point method for simulating 3D fluid-structure interaction problems," International Journal for Numerical Methods in Fluids, **56**, 2151-2177, 2008.
13. M.D. de Tullio, P. De Palma, G. Iaccarino, G. Pascazio, and M. Napolitano, "An immersed boundary method for compressible flows using local grid refinement," Journal of Computational Physics, **225**, 2098-2117, 2007.
14. S.V. Patankar, Numerical Heat Transfer and Fluid Flow, Hemisphere, 1980.
15. S.R. Mathur and J.Y.Murthy, "A pressure-based method for unstructured meshes," Numerical Heat Transfer, **31** (2), 195-216, 1997.
16. D. Sulsky, S-J. Zhou and H.L. Schreyer, "Application of a particle-in-cell method to solid mechanics," Computer Physics Communications, **87**, 236-252, 1995.
17. C. M. Rhie and W. L. Chow, "Numerical study of the turbulent flow past an airfoil with trailing edge separation," AIAA Journal, **21**(11), 1523-1532, 1983.
18. R.L. Lohner, "Some useful data structures for the generation of unstructured grids," Communications in Applied Numerical Methods, **4**, 123-135, 1998.
19. T.J. Barth, "Aspects of unstructured grids and finite volume solvers for the Euler and Navier-Stokes equations," Special Course on Unstructured Grid Methods for Advection-Dominated Flows, AGARD Report 787, 1992.
20. J. Kim, D. Kim and H. Choi, "An immersed-boundary finite volume method for simulation of flow in complex geometries," Journal of Computational Physics, **79**, 317-327, 2001.
21. F. Muldoon and S.Acharya, "A divergence-free interpolation scheme for the immersed boundary method," International Journal for Numerical Methods in Fluids, **56**, 1845-1884, 2008.
22. I.G. Currie, Fundamentals of Fluid Mechanics, McGraw-Hill, New York, 1974.
23. B. Rahul, T. Ryan, A. Alina, S. Hartono and A. Raman, "Unified theory of gas damping of flexible microscantilevers at low ambient pressures", Applied Physics Letter, 94, 163117-163119, 2009.