Chapter 14
Hybrid Lagrangian–Eulerian Method-Based CFSD Development, Application, and Analysis

Namshad Thekkethil and Atul Sharma

14.1 Introduction

Fluid–structure dynamics (*FSD*)—a coupled interaction between fluid dynamics and structure dynamics—is one of the complex phenomena observed in nature and has led to the development of biomimetic-based engineering systems. Analysis of the FSD in the natural systems could lead to a better design of the biomimetic systems. Since experimental methods have several limitations with regard to physical model for the complex FSD phenomenon, computational methods can take the lead in the analysis of the experimentally challenging FSD problems. For the *computational fluid–structure dynamics* (*CFSD*), there are various types of methods that are based on independent advancements in *computational fluid dynamics* (*CFD*) and *computational structure dynamics* (*CSD*) along with a coupling between the CFD and CSD that can be either one-way or two-way. For a *one-way coupled CFSD*, the structure is rigid and subjected to a forced motion that is independent of fluid dynamic forces acting on the structure while the fluid flow depends on the kinematic conditions of the structure. For a *two-way coupled CFSD*, the fluid flow and motion and/or deformation of flexible/rigid structure are dependent on each other; the motion/deformation of the structure is caused by the fluid dynamic forces.

N. Thekkethil · A. Sharma (✉)
Indian Institute of Technology Bombay, Mumbai 400076, India
e-mail: atulsharma@iitb.ac.in

N. Thekkethil
e-mail: namshad.th@gmail.com

S. Roy et al. (eds.), *Immersed Boundary Method*, Computational Methods
in Engineering & the Sciences, https://doi.org/10.1007/978-981-15-3940-4_14

14.1.1 CFSD Development, Application, and Analysis

Computational fluid–structure dynamics involves the development of a software, its application for a fluid–structure dynamics problem to obtain scientifically exciting and engineering-relevant results, and analysis of the results for a unified cause-and-effect study (Sharma 2017). Historically, the Eulerian approach-based finite volume method (FVM) is usually preferred in CFD while CSD prefers Lagrangian approach-based finite element method (FEM). However, for CFSD, various combinations of Eulerian and Lagrangian methods are considered that are broadly classified into two approaches: monolithic and partitioned. The *monolithic approach* considers fluid and structure as a continuum and uses either Eulerian or Lagrangian approach throughout the domain, whereas the *partitioned approach* solves the fluid flow and the structure motion or/and deformation separately along with a coupling condition at the fluid–solid interface. The partitioned approach is further classified into three types: fully Lagrangian, arbitrary Lagrangian–Eulerian (ALE), and *hybrid Lagrangian–Eulerian* (*HLE*) methods.

Fully Lagrangian method (Belytschko and Kennedy 1975; Donea et al. 1976) considered both the fluid dynamics and structural dynamics in the Lagrangian system and was the first choice for CFSD. However, the Lagrangian method for fluid flow is limited to almost stationary fluid since the fluid flow leads to a distortion of the mesh. ALE method considers body-fitted mesh and involves dynamic meshing without the mesh distortion problems (Noh 1963). ALE methods are efficient for many classes of FSI problems; however, it is limited by the need to re-mesh and gets into the trouble of the mesh distortion at a larger deformation of the structure. The HLE methods are the best choice for large deformation of CFSD problems. It uses the Eulerian approach for CFD and the Lagrangian approach for CSD. The HLE method presented here was proposed in our recent work (Thekkethil and Sharma 2019) for both one-way and two-way coupled CFSD problems. The HLE method considers a physical law-based FVM (Sharma 2017) and a *level-set function-based immersed boundary method* (LS-IBM) for CFD and geometric nonlinear Galerkin FEM for CSD along with direct implementation of coupling conditions at the fluid–solid interface.

14.1.2 Immersed Boundary Method

A historical development for CFD simulation of flow across immersed complex-shaped body started with a finite difference method-based solution on a Cartesian grid that approximates the curved body as a stepped one. Later, a finite volume method-based solution on a body-fitted grid was proposed initially for a structured curvilinear grid and later for an unstructured grid. The FVM-based solution continued for many years; however, the progress in CFD application from flow across a stationary structure to a moving and/or deforming structure led to various numerical challenges in generating a *time-wise varying* body-fitted structured/unstructured grid. This led to a renewed interest in the application of Cartesian grid although with a new form—non-body-fitted Cartesian grid.

Immersed boundary method (*IBM*) (Peskin 2002) is a numerical methodology for finite difference method (FDM) or FVM-based CFSD development on a *non-body-fitted* and *fixed* Cartesian grid that involves a special treatment for implementation of fluid–solid interface boundary conditions and also for the CFD solution on the partially filled fluid cells. IBM gained popularity during the last few decades. The motion/deformation of the immersed structure results in certain Cartesian fluid cells (near the fluid–solid interface) to be partially or entirely filled with the solid at certain time instants. Depending on the numerical method to handle the change in the fluid cells near the moving interface, many IBMs are available in the literature that can be broadly classified into two types (Mittal and Iaccarino 2005): continuous forcing IBM and discrete forcing IBM (Mittal and Iaccarino 2005). The discrete forcing IBM is further classified based on the direct or indirect implementation of fluid–solid interface boundary conditions. A *sharp-interface* IBM (Udaykumar et al. 2001; Mittal et al. 2008) considers the physically realistic sharp fluid–solid interface, while a numerically diffused fluid–solid interface is considered in a *diffused interface* IBM (Pan 2006; Patel and Natarajan 2018). Depending on the strategy used for the application of fluid–solid interface boundary conditions, various sharp-interface methods are available in the literature, such as ghost-cell-based IBM (Majumdar et al. 2001; Mittal et al. 2008) and cut-cell-based IBM (Udaykumar et al. 2001). Both methods use a certain type of interpolation for the application of fluid–solid interface boundary conditions.

14.1.3 Outline of the Chapter

In this chapter, we present an HLE method-based CFSD development in Sect. 14.3 and its application for analysis of various types of one-/two-way coupled CFSD problems in Sect. 14.4. The HLE method (Thekkethil and Sharma 2019) involves FVM and LS-IBM for fluid dynamics and geometric nonlinear Galerkin FEM for structural dynamics and is based on a partitioned approach. The associated conservation laws and the fluid–solid coupling conditions are presented in Sect. 14.2.

14.2 CFSD: Conservation Laws and Fluid–Solid Coupling Conditions

For any FSD problem, conservation laws for fluid flow and structure dynamics need to be satisfied along with a continuity of stress and kinematics as the coupling condition at the fluid–solid interface. For fluid flow, mass and momentum conservation laws are considered in the Eulerian form while a Lagrangian form of momentum conservation law is considered for motion as well as deformation of the structure.

14.2.1 Mass and Momentum Conservation Laws for a Fluid Control Volume: Eulerian Form

For an *incompressible* fluid control volume (CV) with volume Ω^v and surface Γ^v (Fig. 14.1a), the Eulerian form of unsteady mass and momentum conservation laws is given for a negligible body force as

$$\text{Mass: } \mathrm{M}^v_{\text{out}} - \mathrm{M}^v_{\text{in}} = 0 \tag{14.1}$$

$$\text{Momentum: } \frac{\partial}{\partial t}\left(\mathcal{M}\vec{u}\right)^v + A^v_{\text{out}} - A^v_{\text{in}} = \vec{\mathrm{F}}^{\,v}_{s} \tag{14.2}$$

where M^v_{in} and $\mathrm{M}^v_{\text{out}}$ are the mass flow rates while A^v_{in} and A^v_{out} are the momentum flow rates entering and leaving the CV, respectively. Furthermore, $\mathcal{M}$ is the mass, $\vec{u}$ is the velocity, and $\vec{\mathrm{F}}^{\,v}_{s}$ is the surface force acting on the surface Γ^v of the control volume.

14.2.2 Momentum Conservation Law for a Solid Control Mass: Lagrangian Form

For a solid control mass with volume Ω^m and surface Γ^m (Fig. 14.1b), the Lagrangian form of momentum conservation law is given for a negligible body force as

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\mathcal{M}\vec{u}\right)^m = \vec{\mathrm{F}}^{\,m}_{s} \text{ where } \vec{u} = \frac{\mathrm{d}\vec{\mathrm{d}}}{\mathrm{d}t} \tag{14.3}$$

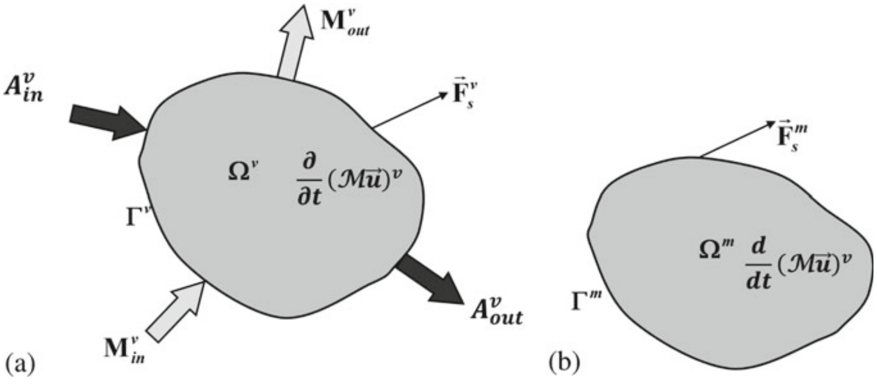


Fig. 14.1 **a** Mass and momentum conservation on a fluid *control volume* Ω^v with control surface Γ^v and **b** momentum balance in a solid *control mass* Ω^m with control surface Γ^m

Here, $\frac{\mathrm{d}}{\mathrm{d}t}\left(\mathcal{M}\overrightarrow{u}\right)^m$ is the rate of change of instantaneous momentum of the control mass, $\overrightarrow{\mathrm{F}}_s^m$ is the surface force acting on the surface Γ^m, and $\overrightarrow{d}$ is the displacement vector of the control mass.

14.2.3 Fluid–Solid Coupling Conditions

Coupled fluid dynamics and structural dynamics govern the fluid–solid interface dynamics. The coupling is obtained by continuity of kinematics and stress field at the interface, given as

$$\overrightarrow{u}_{f,\mathrm{int}} = \overrightarrow{u}_{s,\mathrm{int}} \text{ and } \sigma_{s,\mathrm{int}}.\hat{n} = \sigma_{f,\mathrm{int}}.\hat{n} \tag{14.4}$$

Here, the subscripts f and s represent the fluid and structure, respectively, and int represents the fluid–solid interface. The vector $\hat{n}$ represents the unit normal vector at the interface.

14.3 HLE Method-Based CFSD Development: Hybrid FEM-FVM-Based Numerical Methodology

The present HLE method (Thekkethil and Sharma 2019) uses a form of conservation law that is Lagrangian for structure dynamics and Eulerian for fluid dynamics, presented in the previous subsection. Furthermore, the derivation of the algebraic formulations for the present HLE method-based CFSD development considers a physical law-based finite volume method (Sharma 2017) and a Galerkin finite element method (Zienkiewicz et al. 1977) for the fluid and structure dynamics, respectively. The physical law-based FVM starts with a discrete form of conservation laws, proposed by Sharma (2017) in a recent textbook on CFD as compared to starting with the partial differential equations (PDEs) in almost all the other FVM books on CFD (Patankar 2018; Versteeg and Malalasekera 2007). Both the physical law-based FVM and the PDE-based FVM use the same approximations and, thus, result in the same algebraic formulation for CFD.

Numerical methodology for the FVM-based CFD development and FEM-based CSD development and the associated coupling for HLE method-based CFSD development are presented in separate subsections below.

14.3.1 CFD Development: Physical Law-Based FVM and Level-Set Function-Based Immersed Boundary Method

CFD development consists of five steps (Sharma 2017): grid generation, FVM-based algebraic formulation, solution methodology, computation of engineering parameters, and testing. The first three steps of the CFD development are presented in separate subsections below for the present level-set function-based immersed boundary method (LS-IBM). The present LS-IBM involves a level-set function-based direct implementation of fluid–solid interface boundary condition (Shrivastava et al. 2013); thus, it avoids any interpolation for the interfacial boundary conditions.

14.3.1.1 Cartesian Grid Generation

For the development of a CFD solver, the present LS-IBM considers a fixed Cartesian grid, as shown in Fig. 14.2. For flow across a non-Cartesian or complex-shaped structure, as seen in Fig. 14.2, the non-body-fitted Cartesian grid results in certain partially filled fluid *control volumes* (CVs) that require special treatment to ensure mass and momentum conservation laws and no-slip boundary conditions. The figure shows the various types of CVs for the Cartesian grid.

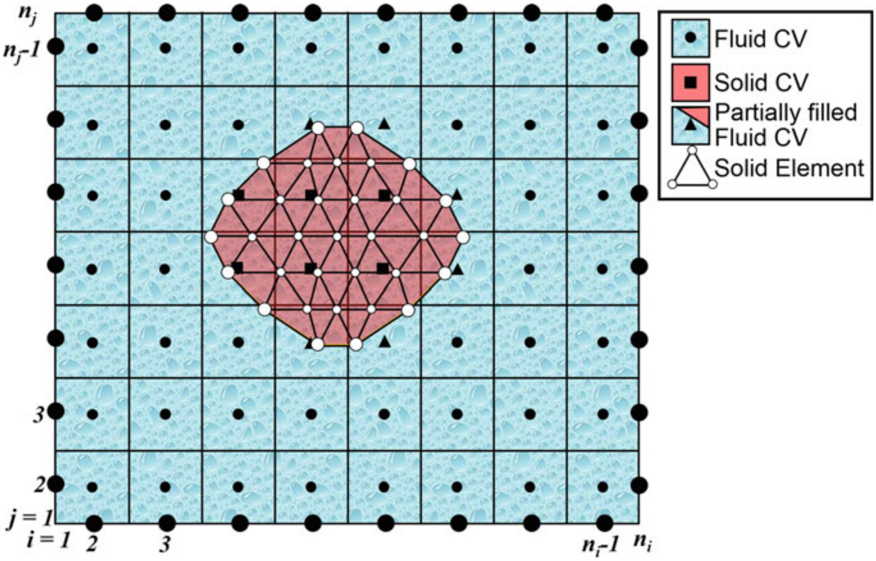


Fig. 14.2 Computational domain for a 2D FSI problem with a Lagrangian triangular mesh for structure immersed in the Eulerian Cartesian mesh in the complete domain

14.3.1.2 Physical Law-Based FVM

Figure 14.3a shows a computational stencil for a CV P whose neighbouring CVs are also fluid CVs. Considering the computational stencil, for a 2D incompressible flow, a mass and momentum conservation law-based FVM results in discrete mathematics-based approximated algebraic formulation, given (Sharma 2017) as

$$\left(m_{x,e}^{n+1} - m_{x,w}^{n+1}\right) \triangle y_P - \left(m_{y,n}^{n+1} - m_{y,s}^{n+1}\right) \triangle x_P = 0 \tag{14.5}$$

$$\begin{gathered}
\rho_f \frac{\phi_P^{n+1} - \phi_P^n}{\triangle t} \triangle V_P + A_{\phi,P}^{n+1} = D_{\phi,P}^{n+1} + S_{\phi,P}^{n+1} \\
\text{where } \mathrm{A}_{\phi,P} = \left[\left(m_{x,e}^{+}\phi_e^{+} + m_{x,e}^{-}\phi_e^{-}\right) - \left(m_{x,w}^{+}\phi_w^{+} + m_{x,w}^{-}\phi_w^{-}\right)\right] \triangle y_P \\
\left[\left(m_{y,n}^{+}\phi_n^{+} + m_{y,n}^{-}\phi_n^{-}\right) - \left(m_{y,s}^{+}\phi_s^{+} + m_{y,s}^{-}\phi_s^{-}\right)\right] \triangle x_P \\
D_{\phi,P} = \mu_f \left[\left(\frac{\phi_E - \phi_P}{\delta x_e} - \frac{\phi_P - \phi_W}{\delta x_w}\right) \triangle y_P + \left(\frac{\phi_N - \phi_P}{\delta y_n} - \frac{\phi_P - \phi_S}{\delta y_s}\right) \triangle x_P\right] \\
S_{u,P} = \left(p_w - p_e\right) \triangle y_P, S_{v,P} = \left(p_s - p_n\right) \triangle x_P
\end{gathered} \tag{14.6}$$

where m_x and m_y are the components of mass flux in x- and y-directions, respectively, and the superscript $n+1$ represents the time instant $(t + \triangle t)$.

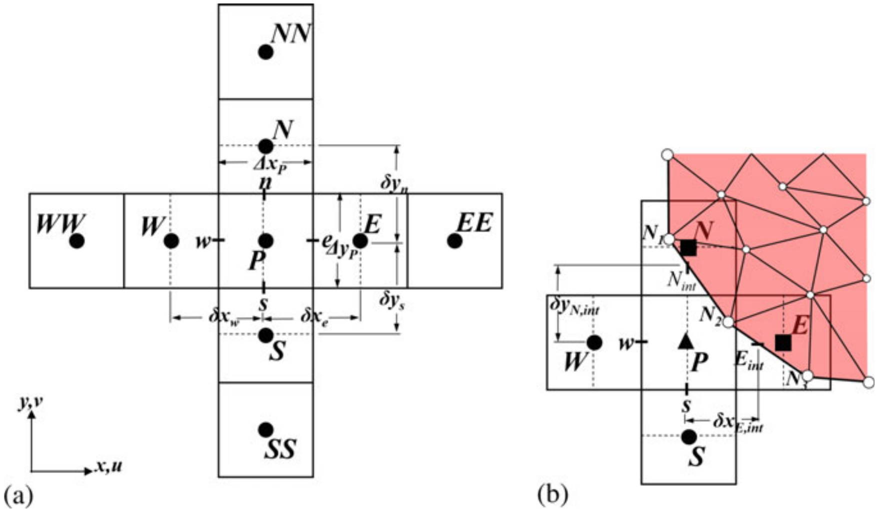


Fig. 14.3 Computational stencil for a fluid control volume with **a** all the neighbouring cells in the fluid and **b** north and east neighbours in the structure

For a *semi-implicit* solution methodology considered here, momentum equation considers the implicit time-level $(n + 1)$ for all the terms (advection A, diffusion D, source S). Furthermore, $\phi = u$ and $\phi = v$ in Eq. (14.6) correspond to x-momentum and y-momentum conservations, respectively. The advection terms $A_{u,P}$ and $A_{v,P}$ are the advection of x-momentum flow rate and y-momentum flow rate, while the diffusion terms $D^{\chi}_{u,P}$ and $D^{\chi}_{v,P}$ are the viscous forces in x- and y-directions, respectively. Furthermore, ρ_f is the density of the fluid, μ_f is the viscosity of the fluid, and p is the pressure acting on the surface of the CVs. Also, in the advection term, the mass flux in the positive and negative directions $m^+ = \max(m, 0)$ and $m^- = \min(m, 0)$, and u/v velocity at the face centre $\phi_{f=e,w,n,s} = w_D\phi_D + w_U\phi_U + w_{UU}\phi_{UU}$ is obtained using an advection scheme (Sharma 2017). Here, D, U, and UU correspond to the downstream, upstream, and upstream-of-upstream values, respectively, and the weights w_D, w_U, and w_{UU} for the first-order upwind (FOU), second-order upwind (SOU), and quadratic upstream interpolation for convective kinematics (QUICK) schemes are obtained from a distance-based extrapolation/interpolation scheme (Sharma 2017).

14.3.1.3 Solution Methodology: Semi-implicit Pressure Projection Method

A *semi-implicit pressure projection method* (SIPPM) on a co-located grid system is used for the unsteady solution of the algebraic formulation—Eq. (14.5) for mass and Eq. (14.6) for momentum conservation. The velocity field $\overrightarrow{u}^{n+1}_P$ at a new time-level $(n + 1)$ is obtained from the momentum conservation equation, while the mass conservation equation is converted into an algebraic equation for pressure (presented below), using a predictor–corrector method in the SIPPM. The predictor step involves prediction (represented by $*$ values) of velocity at cell centre $\overrightarrow{u}^*_P$ as well as normal velocities of mass fluxes at the face centres (u^*_e, u^*_w, v^*_n, and v^*_s). The predicted mass fluxes at the face centres are used to obtain the pressure p^{n+1}_P at the new time-level $(n + 1)$ from the pressure equation. Finally, the pressure field is used to obtain the velocity correction (represented by $'$ values) at the cell centres $\overrightarrow{u}'_P$ and then obtain $\overrightarrow{u}^{n+1}_P = \overrightarrow{u}^*_P + \overrightarrow{u}'_P$. Formulation of the algebraic equations for $\overrightarrow{u}^*_P$, u^*_f/v^*_f, p^{n+1}_P, and $\overrightarrow{u}'_P$ is presented below for the SIPPM.

Original Proposition:

Using Eq. (14.6) with $\phi = u$ or v, the velocity at the cell centre u^{n+1}_P and that at the east face centre u^{n+1}_e are given as

$$\rho_f \frac{u^{n+1}_P - u^n_P}{\Delta t} \Delta V_P + A^{n+1}_{u,P} = D^{n+1}_{u,P} + \left(p^{n+1}_w - p^{n+1}_e\right) \Delta y_P$$
$$\rho_f \frac{u^{n+1}_e - u^n_e}{\Delta t} \Delta V_e + A^{n+1}_{u,e} = D^{n+1}_{u,e} + \left(p^{n+1}_P - p^{n+1}_E\right) \Delta y_P \qquad (14.7)$$

Predictor step:

The projection method results in a velocity predictor equation for u_P^*/u_e^* obtained from the above equation, after dropping the pressure term, given as

$$
\begin{aligned}
\rho_f \frac{u_P^* - u_P^n}{\triangle t} \triangle V_P + A_{u,P}^* &= D_{u,P}^* \\
\rho_f \frac{u_e^* - u_e^n}{\triangle t} \triangle V_e + A_{u,e}^* &= D_{u,e}^*
\end{aligned}
\tag{14.8}
$$

From the above implicit equation, u_P^* is obtained after an iterative solution while u_e^* is approximated by linear interpolation of the neighbouring cell-centre predicted velocity, i.e. $u_e^* = \overline{u_P^*, u_E^*}$; similarly, $u_w^* = \overline{u_P^*, u_W^*}$, $v_n^* = \overline{v_P^*, v_N^*}$, and $v_s^* = \overline{v_P^*, v_S^*}$ where v_P^* is obtained from equation similar to Eq. (14.8). Note that u_e^* is not obtained from the above implicit equation.

Corrector step:

Subtracting Eq. (14.8) for u_e^* from Eq. (14.7) for u_e^{n+1}, we get an approximate velocity correction as

$$
u_e^{n+1} - u_e^* \approx \frac{\triangle t}{\rho_f} \frac{\left(p_E^{n+1} - p_P^{n+1}\right)}{\delta x_e}
\tag{14.9}
$$

$$
\Rightarrow m_{x,e}^{n+1} \approx m_{x,e}^* - \triangle t \frac{\left(p_E^{n+1} - p_P^{n+1}\right)}{\delta x_e}
\tag{14.10}
$$

The approximations in the above equation correspond to neglecting the velocity correction corresponding to the advection and diffusion terms—resulting in the semi-implicit equation (Patankar 2018) although the original proposition is fully implicit [Eq. (14.6)].

Algebraic formulation for pressure:

Equations similar to Eq. (14.10) can be obtained for the mass fluxes at the other face centres, and substituting from these equations to the mass conservation Eq. (14.5), we obtain the pressure equation as

$$
\begin{aligned}
a_P p_P^{n+1} &= a_E p_E^{n+1} + a_W p_W^{n+1} + a_N p_N^{n+1} + a_S p_S^{n+1} + b \\
&\quad \text{where } a_E = \frac{\triangle t \triangle y_P}{\delta x_e}, a_W = \frac{\triangle t \triangle y_P}{\delta x_w}, a_N = \frac{\triangle t \triangle x_P}{\delta y_n}, a_S = \frac{\triangle t \triangle x_P}{\delta y_s}, \\
a_P &= a_E + a_W + a_N + a_S, b = -S_{m,P}^* \\
&= -\left[\left(m_{x,e}^* - m_{x,w}^*\right) \triangle y_P + \left(m_{y,n}^* - m_{y,s}^*\right) \triangle x_P\right]
\end{aligned}
\tag{14.11}
$$

Special Treatment for Partially Filled Fluid CVs

Figure 14.3b shows a control volume P with its east and neighbouring north cells in the structure. The solution procedure is same as other cells except for the computation of advection, diffusion, and pressure terms on the faces whose adjoining CV is a solid CV. For the control volume P in Fig. 14.3b, considering the east and north interfaces as horizontal and vertical lines, the advection and diffusion fluxes at the east and north sides are computed at E_{int} and N_{int}, respectively (instead of e and f), using the values of velocity at the solid boundary. The advection and diffusion fluxes at E_{int} and N_{int} are given as

$$\begin{aligned} a^{n+1}_{\phi x,E,\text{int}} &= m^{n}_{x,E,\text{int}}\phi^{n+1}_{E,\text{int}},\ a^{n+1}_{\phi y,N,\text{int}} = m^{n}_{y,N,\text{int}}\phi^{n+1}_{N,\text{int}} \\ d_{\phi x,E,\text{int}} &= \mu_f\,\frac{\phi^{n+1}_{E,\text{int}}-\phi^{n+1}_P}{\delta x_{E,\text{int}}},\ d_{\phi y,N,\text{int}} = \mu_f\,\frac{\phi^{n+1}_{N,\text{int}}-\phi^{n+1}_P}{\delta y_{E,\text{int}}} \end{aligned} \tag{14.12}$$

where $\delta x_{E,\text{int}}$ and $\delta y_{N,\text{int}}$ are shown in Fig. 14.3b. The interface velocities ($\phi_{E,\text{int}}$ and $\phi_{N,\text{int}}$) and the mass fluxes at the east and north sides are obtained from the neighbouring solid grid points (at the interface) by linear interpolation, given as

$$\begin{aligned} \phi_{E,\text{int}} &= \overline{\phi_{N_2},\phi_{N_3}},\ \phi_{N,\text{int}} = \overline{\phi_{N_1},\phi_{N_2}} \\ m_{x,E,\text{int}} &= \rho_f u_{E,\text{int}},\ m_{y,N,\text{int}} = \rho_f v_{N,\text{int}} \end{aligned} \tag{14.13}$$

where N_1, N_2, and N_3 are solid nodes (defined for FEM), as shown in Fig. 14.3b. Similar linear interpolation is used to obtain $v_{E,\text{int}}$ and $v_{N,\text{int}}$, and the resulting $\overrightarrow{u}_{E,\text{int}}$ and $\overrightarrow{u}_{N,\text{int}}$ are used to obtain the advection fluxes [Eq. (14.12)] without using any advection scheme. The interface distances in the diffusion flux, $\delta x_{E,\text{int}}$ and $\delta y_{E,\text{int}}$ [Eq. (14.12) and Fig. 14.3b], are obtained using the level-set function ψ, given as

$$\delta x_{E,\text{int}} = \frac{x_E - x_P}{|\psi_E - \psi_P|}\,|\psi_P|\,,\ \delta y_{N,\text{int}} = \frac{y_N - y_P}{|\psi_N - \psi_P|}\,|\psi_P| \tag{14.14}$$

For the mass balance, the mass fluxes at east and north faces that corresponds to the interface values [Eq. (14.13)] are directly used; resulting mass balance equation, for the partially filled CV "P" (Fig. 14.3b), is given as

$$\begin{aligned} a_P p^{n+1}_P &= a_W p^{n+1}_W + a_S p^{n+1}_S + b \\ \text{where } a_P &= a_W + a_S,\ b = -S^{*}_{m,P} \\ &= -\left[\left(m^{n+1}_{x,E,\text{int}} - m^{*}_{x,w}\right)\triangle y_P + \left(m^{n+1}_{y,N,\text{int}} - m^{*}_{y,s}\right)\triangle x_P\right] \end{aligned} \tag{14.15}$$

The interface pressures ($P_{E,\text{int}}$ and $P_{N,\text{int}}$) are obtained from the pressure gradient boundary condition at the interface, given as

$$\frac{\partial p}{\partial n} = -\rho_f a_n \implies \nabla p.\hat{n} = -\rho_f\,\overrightarrow{a}\,.\hat{n} \tag{14.16}$$

where a_n is the normal acceleration at the interface. Using the level-set function at the interface, it is given (Shrivastava et al. 2013) as

$$\frac{\partial p}{\partial x}\frac{\partial \psi}{\partial x} + \frac{\partial p}{\partial y}\frac{\partial \psi}{\partial y} = -\rho_f\left(a_{x,\text{int}}\frac{\partial \psi}{\partial x} + a_{y,\text{int}}\frac{\partial \psi}{\partial y}\right) \tag{14.17}$$

where $a_{x,\text{int}}$ and $a_{y,\text{int}}$ are the accelerations in the x- and y-directions at the solid surface. Considering the solid boundary as horizontal and vertical lines, the above equation results in an approximated boundary condition for the pressure, given as

$$\frac{\partial p}{\partial x} \approx -\rho_f a_{x,\text{int}}, \ \frac{\partial p}{\partial y} \approx -\rho_f a_{y,\text{int}} \tag{14.18}$$

Thus, the pressure boundary conditions at the east and north faces of the cell P are given as

$$\begin{aligned} p_{E,\text{int}} &= p_P - \delta x_{E,\text{int}}\rho_f a_{x,E,\text{int}} \\ p_{N,\text{int}} &= p_P - \delta y_{N,\text{int}}\rho_f a_{y,N,\text{int}} \end{aligned} \tag{14.19}$$

where $a_{x,E,\text{int}}$ and $a_{y,N,\text{int}}$ are obtained by linear interpolation from the solid grid points [similar to Eq. (14.13)].

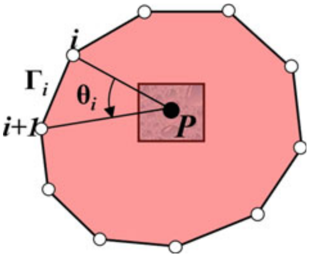
Calculation of Level-Set Function: A Geometric Method

Level-set function ψ (Sethian 1999) is a normal distance function with a change in sign across an interface. The sign of ψ is used to detect the Cartesian CVs that are in the fluid and also the partially filled CVs. Furthermore, its magnitude is used to calculate the diffusion fluxes [Eq. (14.14)] in the partially filled CVs. The sign of the level-set function is distinguished by using a winding number algorithm, and its magnitude is obtained by a minimum distance algorithm, proposed in our recent work (Thekkethil and Sharma 2019). The algorithms are presented here for a 2D solid body; however, it can be extended to 3D geometries also.

The surface of the structure Γ_s is divided into n_{ss} number of line segments Γ_i, as shown in Fig. 14.4, where $i = 1, 2, \ldots, n_{ss}$. For the endpoints in the line segments, the position vectors are specified as $\left[\vec{x}_1, \vec{x}_2, \ldots, \vec{x}_{n_{ss}}\right]$. For any cell P in the Cartesian domain (Fig. 14.4), the sign of the level-set function is computed using the winding number algorithm, which is better than many other methods in this category such as line tracing algorithm, that fails if the shape of the structure is complex. In the winding number algorithm, the counterclockwise angle subtended by each line segment with the cell P is added to obtain the winding number. For the cell P with position vector $\vec{x}$ in Fig. 14.4, the angle subtended by the line segment Γ_i is given as

$$\theta_i = \cos^{-1}\left[\frac{\left(\vec{x}_i - \vec{x}\right).\left(\vec{x}_{i+1} - \vec{x}\right)}{\left|\vec{x}_i - \vec{x}\right|\left|\vec{x}_{i+1} - \vec{x}\right|}\right] \tag{14.20}$$

Fig. 14.4 For a fluid cell in the Cartesian system, computation of angle between two consecutive grid points on the structure surface



The winding number ω_n for the cell P is computed as

$$\omega_{n,P} = \frac{1}{2\pi}\sum_{i=1}^{n_{ss}} \theta_i \tag{14.21}$$

For any cell P inside the structure, $\omega_{n,P}$ will be 1 while it will be 0 for a cell P in the fluid.

For finding the magnitude of the level-set function, the minimum distance algorithm is used. For each Cartesian cell P, the shortest distance from each line segment Γ_i (Fig. 14.4) is calculated as

$$\begin{aligned} |\psi|_{P,i} &= \left|\overrightarrow{x}_d\right| \\ \text{where } \overrightarrow{x}_d &= \begin{cases} \overrightarrow{x} - \overrightarrow{x}_i & \text{if } t < 0 \\ \overrightarrow{x} - \overrightarrow{x}_{i+1} & \text{if } t > 1 \\ \overrightarrow{x} - \left(\overrightarrow{x}_i + t\left[\overrightarrow{x}_{i+1} - \overrightarrow{x}_i\right]\right) & \text{if } 0 \le t \le 1 \end{cases} \\ t &= \frac{\left(\overrightarrow{x} - \overrightarrow{x}_i\right).\left(\overrightarrow{x}_{i+1} - \overrightarrow{x}_i\right)}{\left|\overrightarrow{x}_{i+1} - \overrightarrow{x}_i\right|^2} \end{aligned} \tag{14.22}$$

The magnitude of the level-set function is computed as the minimum of the shortest distance from each line segment, given as

$$|\psi|_P = \min\left(|\psi|_{P,1}, |\psi|_{P,2}, \ldots, |\psi|_{P,n_{ss}}\right) \tag{14.23}$$

For a 3D geometry, a 3D winding algorithm (Jacobson et al. 2013) can be used to find whether the point lies inside or outside the structure. The 3D geometry surface can be divided into a finite number of elements. For each Cartesian cell, similar to θ_i in 2D, the solid angle can be calculated for each element on the surface. Summation of the solid angles with all the elements gives a measure of the winding number. For finding the magnitude of the level-set function, similar to $|\psi|_{P,i}$ in 2D, the shortest distance from each element can be computed, and the minimum of the shortest distance gives the magnitude of the level-set function.

Solution algorithm:

For the present HLE method, the CFD solution is obtained by solving the equations presented in Sect. 14.3.1.3, for the completely as well as partially filled CVs. The solution algorithm to obtain the velocity and pressure at a new $(n+1)$th time step from the old nth time instant is given as follows:

1. Initialise the velocity, pressure, and level-set field in the domain as per the initial configuration variables.
2. Compute the level-set function ψ using Eqs. (14.21) and (14.23).
3. Solve Eq. (14.8) for u^* and v^*.
4. Predict $S^*_{m,P}$ [Eq. (14.11)] required for the pressure equation.
5. Solve the pressure equation, Eq. (14.11).
6. Calculate the corrected mass flux m_f^{n+1} [Eq. (14.10)].
7. Solve Eq. (14.7) for u_P^{n+1} and v_P^{n+1} using mass conserving mass flux $m_{f^{n+1}}$ and linearly interpolated pressure ($p_w = \overline{p_P, p_W}$ and $p_e = \overline{p_P, p_E}$).
8. Set $n = n + 1$ and repeat steps $2 - 8$ for the next time step. Continue up to certain stopping criterion of the transient simulation.

14.3.2 CSD Development: Geometric Nonlinear Galerkin FEM-Based Numerical Methodology for Structural Dynamics

For the structural dynamics involving large deformation, a geometric nonlinear Galerkin FEM-based algebraic formulation is used here to convert the momentum conservation equation [Eq. (14.3)] to a system of linear algebraic equations for displacement vector $\vec{\mathrm{d}}$. Similar to the CFD development in the previous section, CSD development is presented below in separate subsections for grid generation, FEM-based algebraic formulation, and solution methodology.

14.3.2.1 Unstructured Grid Generation

For the development of a CSD solver, the present geometric nonlinear Galerkin FEM-based numerical methodology considers a fixed body-fitted unstructured grid, as shown in Fig. 14.2. The figure shows that the grid generation involves dividing the solid into several control masses, considered triangular here. The control masses are called as *elements,* and the grid points at the vertices of the elements are called as *nodes* in FEM, represented by unfilled circles in Fig. 14.2. Although the triangular elements, along with the nodes, move substantially during large deformation, a fixed node/element is considered in the geometric nonlinear Galerkin FEM presented in the next subsection. This involves defining the deformation vector $\vec{\mathrm{d}}$ at the various solid nodes, with reference to the initial ($t = 0$) node configuration, shown in Fig. 14.2.

14.3.2.2 Geometric Nonlinear Galerkin Finite Element Method

In order to solve the Lagrangian form of the momentum conservation law [Eq. (14.5)] for the deformation vector $\overrightarrow{\mathrm{d}}$ at the various nodes (Fig. 14.2), the algebraic formulation for a three-node triangular element is presented here. For the volume Ω^e and surface Γ^e of the element, the instantaneous momentum conservation [Eq. (14.5)] for the element e is given as

$$\left(\frac{\mathrm{d}^2}{\mathrm{d}t^2}\int\limits_{\Omega^e}\rho_s\overrightarrow{\mathrm{d}}^e\mathrm{d}\Omega^e\right)^{n+1}=\int\limits_{\Gamma_c^e}\sigma_c^{n+1}.\hat{n}\mathrm{d}\Gamma_c^e=\int\limits_{\Gamma_{\mathrm{in}}^e}\left(\mathbb{D}^{e,n+1}\mathrm{S}^{e,n+1}\right).\hat{n}\mathrm{d}\Gamma_{in}^e \tag{14.24}$$

where the L.H.S of the above equation corresponds to the unsteady term $\frac{\mathrm{d}}{\mathrm{d}t}\left(\mathcal{M}\overrightarrow{u}\right)$ and R.H.S to the surface force term $\overrightarrow{\mathrm{F}}_s^e$. Also, note that the above equation for the force is first represented with reference to the current (instantaneous) configuration $\overrightarrow{\mathrm{F}}_{s,c}^e$ and then with reference to the initial ($t=0$) configuration $\overrightarrow{\mathrm{F}}_{s,\mathrm{in}}^e$. Here, ρ_s is the density of the solid and $\overrightarrow{\mathrm{d}}^e$ is the displacement vector of the element. $\overrightarrow{\mathrm{F}}_{s,c}^e$ is presented above as the surface integral of the Cauchy stress stress σ that is with reference to the deformed or current configuration. Its conversion with reference to the initial configuration (with surface area Γ_{in}^e) results in a product of $\mathbb{D}^e$ and S^e and corresponds to *deformation gradient* and *second Piola–Kirchhoff's stress*, respectively. They are given in 2D Cartesian coordinate system as

$$\mathbb{D}^e=I+\left(\nabla\overrightarrow{\mathrm{d}}^e\right)^T=\begin{bmatrix}1+d_{x,x}^e & d_{x,y}^e\\ d_{y,x}^e & 1+d_{y,y}^e\end{bmatrix}\text{ and }\mathrm{S}^e=\begin{bmatrix}S_{xx}^e & S_{xy}^e\\ S_{yx}^e & S_{yy}^e\end{bmatrix} \tag{14.25}$$

Here, the components of S^e are presented in a matrix form as $\mathbf{S^e}=\mathrm{D}\mathbf{E}^e$, where $\mathbf{S^e}$ is the *element stress matrix,* D is the *stress–strain relationship matrix*, and $\mathbf{E}^e$ is the element *Green strain matrix*. Using St. Venant–Kirchhoff's model for a plain-strain case, the matrix form of stress–strain relationship is given as

$$\mathbf{S^e}=\mathbf{D}\mathbf{E}^e\Rightarrow\begin{bmatrix}S_{xx}^e\\ S_{yy}^e\\ S_{xy}^e\end{bmatrix}=\begin{bmatrix}C_1 & C_2 & 0\\ C_2 & C_1 & 0\\ 0 & 0 & C_3\end{bmatrix}\begin{bmatrix}d_{x,x}^e+0.5\left[\left(d_{x,x}^e\right)^2+\left(d_{y,x}^e\right)^2\right]\\ d_{y,y}^e+0.5\left[\left(d_{x,y}^e\right)^2+\left(d_{y,y}^e\right)^2\right]\\ d_{x,y}^e\left(1+d_{x,x}^e\right)+d_{y,x}^e\left(1+d_{y,y}^e\right)\end{bmatrix} \tag{14.26}$$

where $C_1={}^{E(1-\nu_s)}/_{(1+\nu_s)(1-2\nu_s)}$, $C_2={}^{E\nu_s}/_{(1+\nu_s)(1-2\nu_s)}$, and $C_3={}^{E}/_{2(1+\nu_s)}$. Here, E is Young's modulus and ν_s is Poisson's ratio of the solid material. Further, the suffix after the comma for d^e above represents the derivative, i.e. $d_{x,x}^e=\frac{d}{dx}d_x^e$.

Using the Gauss divergence theorem, Eq. (14.24) is given as

$$\rho_{s,\text{in}}\left(\frac{\mathrm{d}^2}{\mathrm{d}t^2}\int\limits_{\Omega^e_{in}}\overrightarrow{\mathrm{d}}^{\,e}d\Omega^e_{\text{in}}\right)^{n+1}=\int\limits_{\Omega^e_{\text{in}}}\nabla.\left(\mathbb{D}^{e,n+1}\mathrm{S}^{e,n+1}\right)d\Omega^e_{\text{in}} \tag{14.27}$$

Using a bilinear interpolation with $\overrightarrow{\mathrm{d}}^{\,e}=a\overrightarrow{x}+b\overrightarrow{y}+c$ for the element e, the constants a, b, and c are obtained as $f\left(\overrightarrow{x}_1,\overrightarrow{x}_2,\overrightarrow{x}_3,\overrightarrow{\mathrm{d}}^{\,e}_1,\overrightarrow{\mathrm{d}}^{\,e}_2,\overrightarrow{\mathrm{d}}^{\,e}_3\right)$, which after certain rearrangements results in a function form of the displacement vector for the element e as

$$\overrightarrow{\mathrm{d}}^{\,e}=N^e_1\overrightarrow{\mathrm{d}}^{\,e}_1+N^e_2\overrightarrow{\mathrm{d}}^{\,e}_2+N^e_3\overrightarrow{\mathrm{d}}^{\,e}_3$$
$$\text{where, }\begin{bmatrix}N^e_1\\N^e_2\\N^e_3\end{bmatrix}=\frac{1}{\begin{vmatrix}1&x^e_1&y^e_1\\1&x^e_2&y^e_2\\1&x^e_3&y^e_3\end{vmatrix}}\begin{bmatrix}x^e_2y^e_3-x^e_3y^e_2+\left(y^e_2-y^e_3\right)x+\left(x^e_3-x^e_2\right)y\\x^e_3y^e_1-x^e_1y^e_3+\left(y^e_3-y^e_1\right)x+\left(x^e_1-x^e_3\right)y\\x^e_1y^e_2-x^e_2y^e_1+\left(y^e_1-y^e_2\right)x+\left(x^e_2-x^e_1\right)y\end{bmatrix} \tag{14.28}$$

Here, N^e_1, N^e_2, and N^e_3 are called as *shape functions* of the element e with respect to the nodes 1, 2, and 3, respectively (Fig. 14.5). Substituting $\overrightarrow{\mathrm{d}}^{\,e}$ from Eqs. (14.28) to (14.27), we get

$$\rho_{s,\text{in}}\left[\frac{\mathrm{d}^2}{\mathrm{d}t^2}\int\limits_{\Omega^e_{\text{in}}}\left(N^e_1\overrightarrow{\mathrm{d}}^{\,e}_1+N^e_2\overrightarrow{\mathrm{d}}^{\,e}_2+N^e_3\overrightarrow{\mathrm{d}}^{\,e}_3\right)\mathrm{d}\Omega^e_{\text{in}}\right]^{n+1}=\left[\int\limits_{\Omega^e_{\text{in}}}\nabla.\left(\mathbb{D}^e\mathrm{S}^e\right)\mathrm{d}\Omega^e_{\text{in}}\right]^{n+1} \tag{14.29}$$

Nodal Equations:

For conversion of the above element equation to a nodal equation, the geometric nonlinear Galerkin FEM (as compared to other FEMs) involves multiplication of the

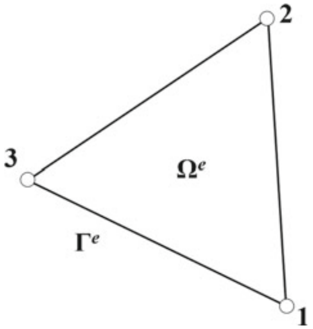


Fig. 14.5 A triangular element with three nodes considered for the geometric nonlinear Galerkin FEM-based algebraic formulation for an element "*e*"

integrand of the above equation with the shape functions of the corresponding nodes. Using this operation, Eq. (14.29) results in a nodal for node i as

$$\begin{aligned}
&\rho_{s,\text{in}}\left[\frac{\mathrm{d}^2}{\mathrm{d}t^2}\int\limits_{\Omega^e_{\text{in}}}\left(N_1^e\overrightarrow{\mathrm{d}}{}_1^e+N_2^e\overrightarrow{\mathrm{d}}{}_2^e+N_3^e\overrightarrow{\mathrm{d}}{}_3^e\right)N_i^e\mathrm{d}\Omega^e_{\text{in}}\right]^{n+1}\\
&=\left[\int\limits_{\Gamma^e_{\text{in}}}\left(\mathbb{D}^e\mathrm{S}^e\right)N_i^e.\hat{n}\mathrm{d}\Gamma^e_{\text{in}}\right]^{n+1}-\left[\int\limits_{\Omega^e_{\text{in}}}\left(\mathbb{D}^e\mathrm{S}^e\right).\nabla\left(N_i^e\right)\mathrm{d}\Omega^e_{\text{in}}\right]^{n+1}
\end{aligned}\tag{14.30}$$

where the first surface integral term (obtained after applying the Gauss divergence theorem) on the R.H.S above represents the surface force acting on node i, $\overrightarrow{F}{}_i^e$. Furthermore, substituting $\mathbb{D}^e$ and S^e from Eq. (14.25), the integrand of the second term in the R.H.S of the above equation is presented in matrix form. They are given as

$$\begin{aligned}
\left[\int\limits_{\Gamma^e_{\text{in}}}\left(\mathbb{D}^e\mathrm{S}^e\right)N_i^e.\hat{n}\mathrm{d}\Gamma^e_{\text{in}}\right]^{n+1}&=\begin{bmatrix}F^e_{xi}\\F^e_{yi}\end{bmatrix},\\
\left(\mathbb{D}^e\mathrm{S}^e\right).\nabla\left(N_i^e\right)&=\begin{bmatrix}1+d^e_{x,x} & d^e_{x,y}\\ d^e_{y,x} & 1+d^e_{y,y}\end{bmatrix}\begin{bmatrix}S^e_{xx} & S^e_{xy}\\ S^e_{yx} & S^e_{yy}\end{bmatrix}\begin{bmatrix}N^e_{i,x}\\N^e_{i,y}\end{bmatrix}\\
&=\left(\left[\mathbf{B}_i^{L,e}\right]^T+\left[\mathbf{B}_i^{NL,e}\right]^T\right)\mathbf{S}^e
\end{aligned}\tag{14.31}$$

where $\mathbf{B}_i^{L,e}$ and $\mathbf{B}_i^{NL,e}$ are the *linear* and *nonlinear deformation matrices* for node i, given as

$$\mathbf{B}_i^{L,e}=\begin{bmatrix}N^e_{i,x} & 0\\ 0 & N^e_{i,y}\\ N^e_{i,y} & N^e_{i,x}\end{bmatrix},\ \mathbf{B}_i^{NL,e}=\begin{bmatrix}N^e_{i,x}d^e_{x,x} & N^e_{i,x}d^e_{y,x}\\ N^e_{i,y}d^e_{x,y} & N^e_{i,y}d^e_{y,y}\\ N^e_{i,x}d^e_{x,y}+N^e_{i,y}d^e_{x,x} & N^e_{i,x}d^e_{y,y}+N^e_{i,y}d^e_{y,x}\end{bmatrix}\tag{14.32}$$

Here, the suffix after the comma for N_i^e above represents the derivative, i.e. $N^e_{i,x}=\frac{d}{dx}N_i^e$. Further, the element stress matrix $\mathbf{S}^e$ in Eq. (14.31) is represented in terms of $\mathbf{B}^{L,e}$ and *modified deformation matrix* $\mathbf{Bm}^e$, by substituting Eq. (14.28) for $\overrightarrow{\mathrm{d}}{}^e$ into Eq. (14.26) for $\mathbf{S}^e$, given as

$$\begin{aligned}
&\mathbf{S}^e=\mathrm{D}\left(\mathbf{B}^{\mathrm{L,e}}+\mathbf{Bm}^{\mathrm{e}}\right)\mathbf{d}^e\\
&\text{where }\mathbf{B}^{L,e}=\left[\mathbf{B}_1^{L,e}\ \mathbf{B}_2^{L,e}\ \mathbf{B}_3^{L,e}\right]\text{ and }\mathbf{Bm}^e=\left[\mathbf{Bm}_1^e\ \mathbf{Bm}_2^e\ \mathbf{Bm}_3^e\right]
\end{aligned}\tag{14.33}$$

where $\mathbf{Bm}_i^e$ and $\mathbf{d}^e$ (nodal displacement vector of the element e) are given as

$$\mathbf{Bm}_i^e = 0.5\begin{bmatrix} N_{i,x}^e d_{x,x}^e & N_{i,x}^e d_{y,x}^e \\ N_{i,y}^e d_{x,y}^e & N_{i,y}^e d_{y,y}^e \\ 2N_{i,y}^e d_{x,x}^e & 2N_{i,x}^e d_{y,y}^e \end{bmatrix}, \mathbf{d}^e = \begin{bmatrix} d_{x1}^e \\ d_{y1}^e \\ d_{x2}^e \\ d_{y2}^e \\ d_{x3}^e \\ d_{y3}^e \end{bmatrix} \tag{14.34}$$

Substituting Eqs. (14.31)–(14.30), taking all the terms to the R.H.S, and using *generalised Newmark algorithm* (Zienkiewicz et al. 1977) [with a second-degree polynomial approximation for time variation and second-order accuracy (GN22)], the *residual vector* $\mathbf{\Psi}$ of an element e with respect to node i for $(n+1)$th time instant is given as

$$\mathbf{\Psi}_i^{e,n+1} = \begin{bmatrix} F_{xi}^e \\ F_{yi}^e \end{bmatrix} - \left[\int\limits_{\Omega_{\text{in}}^e} \left(\left[\mathbf{B}_i^{L,e}\right]^T + \left[\mathbf{B}_i^{NL,e}\right]^T\right)\mathbf{S}^e d\Omega_{\text{in}}^e\right]^{n+1} - \\ \frac{2\rho_{s,\text{in}}}{\triangle t^2}\int\limits_{\Omega_{\text{in}}^e}\begin{bmatrix} N_1^e & 0 & N_2^e & 0 & N_3^e & 0 \\ 0 & N_1^e & 0 & N_2^e & 0 & N_3^e \end{bmatrix} N_i^e d\Omega_{\text{in}}^e \left[\mathbf{d}^{e,n+1} - \mathbf{d}^{e,n} + \triangle t \times \mathbf{u}^{e,n}\right] = 0 \tag{14.35}$$

where $\mathbf{u}^{e,n}$ is the nodal velocity vector of the element e at nth time instant. Since the above equation is nonlinear, an iterative method is used for the solution. Using the *Newton–Raphson method*, the residual vector at $(k+1)$th iterative step is obtained by a Taylor series expansion, given as

$$\mathbf{\Psi}_i^{e,n+1,k+1} \approx \mathbf{\Psi}_i^{e,n+1,k} + \frac{\partial \mathbf{\Psi}_i^{e,n+1,k}}{\partial \mathbf{d}^{e,n+1}} d\mathbf{d}^{e,n} = 0 \tag{14.36}$$

where $d\mathbf{d}^{e,n}$ is the increment to the displacement vector, given as

$$d\mathbf{d}^{e,n} = \mathbf{d}^{e,n+1,k+1} - \mathbf{d}^{e,n+1,k} \tag{14.37}$$

Thus, the final equation for the three nodes of a triangular element—called as *nodal equation*—is given as

$$\mathbf{K}_i^e d\mathbf{d}^{e,n} = \mathbf{\Psi}_i^{e,n+1k} \text{ where } \mathbf{K}_i^e = -\frac{\partial \mathbf{\Psi}_i^{e,n+1,k}}{\partial \mathbf{d}^{e,n+1}} \text{ and } i = 1, 2, 3 \tag{14.38}$$

Here, $\mathbf{K}_i^e$ is the *element stiffness matrix* for the element e with respect to node i and the associated derivative of $\mathbf{\Psi}_i^{e,n+1,k}$ [Eq. (14.35)] with respect to $\mathbf{d}^{e,n+1}$, given as

$$\mathbf{K}_i^e = 0 + \left(\mathbf{K}_{m,i}^e + \mathbf{G}_i^e\right) + \frac{2\mathbf{M}_i^e}{\triangle t^2} \tag{14.39}$$

where the derivative of Eq. (14.35) with respect to $\mathbf{d}^{e,n+1}$ for the surface force $\overrightarrow{F}^e_i$ is zero, differentiation in parts for the stress term results in the terms shown above inside the bracket, and that for the unsteady term results in $2\mathbf{M}^e_i/\triangle t^2$. $\mathbf{K}^e_{m,i}$, $\mathbf{G}^e_i$, and $\mathbf{M}^e_i$ in the above equation are called as *material tangent matrix*, *geometric stiffness matrix*, and *mass matrix*, respectively. The material tangent matrix is given as

$$\mathbf{K}^e_{m,i} = \int\limits_{\Omega^e_{\text{in}}} \left(\left[\mathbf{B}^{L,e}_i\right]^T + \left[\mathbf{B}^{NL,e}_i\right]^T \right)^{n+1} \frac{d}{d\mathbf{d}^{e,n+1}} \mathbf{S}^{e,n+1} d\Omega^e_{\text{in}} \tag{14.40}$$

Substituting $\mathbf{S}^e$ from Eq. (14.33) and using $\frac{d}{d\mathbf{d}^{e,n+1}} \left(\left[\mathbf{B}^{L,e}_i\right]^T \right) = 0$, we get

$$\begin{aligned} \frac{d}{d\mathbf{d}^{e,n+1}} \mathbf{S}^{e,n+1} &= \mathrm{D}\mathbf{B}^{e,n+1} \text{ where } \mathbf{B}^e = \mathbf{B}^{L,e} + \mathbf{B}^{NL,e} \\ \Longrightarrow \mathbf{K}^e_{m,i} &= \left(\left[\mathbf{B}^e_i\right]^T \right)^{n+1} \mathrm{D}\mathbf{B}^{e,n+1} \Omega^e_{\text{in}} \end{aligned} \tag{14.41}$$

Further, using $\frac{d}{d\mathbf{d}^{e,n+1}} \left(\left[\mathbf{B}^{L,e}_i\right]^T \right) = 0$, the geometric stiffness matrix $\mathbf{G}^e_i$ is given as

$$\begin{aligned} \mathbf{G}^e_i &= \int\limits_{\Omega^e_{\text{in}}} \frac{d}{d\mathbf{d}^{e,n+1}} \left(\left[\mathbf{B}^{L,e}_i\right]^T + \left[\mathbf{B}^{NL,e}_i\right]^T \right)^{n+1} \mathbf{S}^{e,n+1} d\Omega^e_{\text{in}} \\ \Rightarrow \mathbf{G}^e_i &= \begin{bmatrix} G^e_{i,1} & 0 & G^e_{i,2} & 0 & G^e_{i,3} & 0 \\ 0 & G^e_{i,1} & 0 & G^e_{i,2} & 0 & G^e_{i,3} \end{bmatrix} \Omega^e_{\text{in}} \text{ where} \\ G^e_{i,j} &= N^e_{i,x} S^{e,n+1}_{xx} N^e_{j,x} + N^e_{i,x} S^{e,n+1}_{xy} N^e_{j,y} + N^e_{i,y} S^{e,n+1}_{xy} N^e_{j,x} + N^e_{i,y} S^{e,n+1}_{yy} N^e_{j,y} \end{aligned} \tag{14.42}$$

The mass matrix $\mathbf{M}^e_i$ is given as

$$\mathbf{M}^e_i = \rho_{s,\text{in}} \int\limits_{\Omega^e_{\text{in}}} \begin{bmatrix} N^e_1 & 0 & N^e_2 & 0 & N^e_3 & 0 \\ 0 & N^e_1 & 0 & N^e_2 & 0 & N^e_3 \end{bmatrix} N^e_1 d\Omega^e_{\text{in}} \tag{14.43}$$

Element equations:

Combining the nodal equations for all the three nodes [Eq. (14.38)], the system of equations for an element e is presented in matrix form as

$$\mathbf{K}^e d\mathbf{d}^{e,n,k} = \mathbf{\Psi}^{e,n+1,k} \text{ where } \mathbf{K}^e = \begin{bmatrix} \mathbf{K}^e_1 \\ \mathbf{K}^e_2 \\ \mathbf{K}^e_3 \end{bmatrix} \text{ and } \mathbf{\Psi}^e = \begin{bmatrix} \mathbf{\Psi}^e_1 \\ \mathbf{\Psi}^e_2 \\ \mathbf{\Psi}^e_3 \end{bmatrix} \tag{14.44}$$

Here, $\mathbf{K}^e$ is the *element stiffness matrix* and $\mathbf{\Psi}^{e,n+1,k}$ is the residual of the element e for $(n+1)$th time step, given as

$$\boldsymbol{\Psi}^{e,n+1,k} = \begin{bmatrix} F^e_{x1} \\ F^e_{y1} \\ F^e_{x2} \\ F^e_{y2} \\ F^e_{x3} \\ F^e_{y3} \end{bmatrix} - \left(\left[\mathbf{B}^e\right]^T\right)^{n+1} \mathbf{S}^{e,n+1}\Omega^e_{\text{in}} - \frac{2\mathbf{M}^e}{\triangle t^2}\left[\mathbf{d}^{e,n+1} - \mathbf{d}^{e,n} + \triangle t \times \mathbf{u}^{e,n}\right],$$

$$\mathbf{K}^e = \mathbf{K}^e_m + \mathbf{G}^e + \frac{2\mathbf{M}^e}{\triangle t^2} \tag{14.45}$$

For the element e, the various matrices in the above equation are given as

$$\mathbf{K}^e_m = \left(\left[\mathbf{B}^e\right]^T\right)^{n+1} \mathrm{D}\left[\mathbf{B}^e\right]^{n+1}\Omega^e_{\text{in}},$$

$$\mathbf{G}^e = \Omega^e_{\text{in}} \begin{bmatrix} G^e_{1,1} & 0 & G^e_{1,2} & 0 & G^e_{1,3} & 0 \\ 0 & G^e_{1,1} & 0 & G^e_{1,2} & 0 & G^e_{1,3} \\ G^e_{2,1} & 0 & G^e_{2,2} & 0 & G^e_{2,3} & 0 \\ 0 & G^e_{2,1} & 0 & G^e_{2,2} & 0 & G^e_{2,3} \\ G^e_{3,1} & 0 & G^e_{3,2} & 0 & G^e_{3,3} & 0 \\ 0 & G^e_{3,1} & 0 & G^e_{3,2} & 0 & G^e_{3,3} \end{bmatrix}, \mathbf{M}^e = \frac{\rho_{s,\text{in}}\Omega^e_{\text{in}}}{12} \begin{bmatrix} 2&0&1&0&1&0 \\ 0&2&0&1&0&1 \\ 1&0&2&0&1&0 \\ 0&1&0&2&0&1 \\ 1&0&1&0&2&0 \\ 0&1&0&1&0&2 \end{bmatrix} \tag{14.46}$$

14.3.2.3 Solution Methodology: Global Equations

For CSD development, solution methodology corresponds to solution of a global system of linear algebraic equations that is obtained by a summation of the element equation [Eq. (14.44) for n_e number of elements], given as

$$\sum_{e=1}^{n_e} \mathbf{K}^e d\mathbf{d}^{e,n,k} = \sum_{e=1}^{n_e} \boldsymbol{\Psi}^{e,n+1,k} \Rightarrow \mathbf{K}^G d\mathbf{d}^{G,n,k} = \boldsymbol{\Psi}^{G,n+1,k} \tag{14.47}$$

where $\mathbf{K}^G$ is the *global stiffness matrix*, $d\mathbf{d}^G$ is the *global displacement-increment vector*, and $\boldsymbol{\Psi}^G$ is the *global residual vector*. The global $\mathbf{K}^G$, $d\mathbf{d}^G$, and $\boldsymbol{\Psi}^G$ are expressed in terms of the respective elemental equations, given as

$$K^G_{2i+p-2,2j+q-2} = \sum_{e=1}^{n_e} K^e_{2l+p-2,2m+q-2}, \quad \text{if } R^{e,l} = i;\ R^{e,m} = j;$$

$$\text{for} \quad \begin{aligned} i &= 1, 2, \ldots, n_s; j = 1, 2, \ldots, n_s \\ l &= 1, 2, 3; m = 1, 2, 3 \\ p &= 1, 2; q = 1, 2 \end{aligned} \tag{14.48}$$

$$d\mathsf{d}^{G,n}_{2i+p-2} = d\mathsf{d}^{e,n}_{2l+p-2}, \text{ if } i = R_{e,l} \text{ , for } \begin{array}{c} e = 1, 2, \ldots, n_e \\ i = 1, 2, \ldots, n_s \\ l = 1, 2, 3 \\ p = 1, 2 \end{array} \tag{14.49}$$

$$\Psi^{G,n+1,k}_{2i+p-2} = \mathsf{f}^{G}_{2i+p-2} - \sum_{e=1}^{n_e} \Psi^{e,n+1,k}_{2l+p-2} \text{ if } i = R^{e,l} \text{ , for } \begin{array}{c} i = 1, 2, \ldots, n_s \\ l = 1, 2, 3 \\ p = 1, 2 \end{array} \tag{14.50}$$

where f^{G}_{2i+p-2} represents the external forces acting on the node i (in x-direction for $p = 1$ and y-direction for $p = 2$), i.e. $\mathsf{f}^{G}_{2i+p-2} = \sum_{e=1}^{n_e} f^{e}_{2l+p-2}$. Here, $\mathbf{K}^G$, $d\mathbf{d}^G$, and $\boldsymbol{\Psi}^G$ are computed with respect to the global node numbering from 1 to n_s. In order to relate global node numbering with the element node numbering, a node relationship matrix is defined as

$$R^{e,l} = 3e - 3 + l \text{ where } e = 1, 2, \ldots, n_e; l = 1, 2, 3 \tag{14.51}$$

The global displacement vector $\mathbf{d}^G$ is also related to the element displacement vector $\mathbf{d}^e$ in the same way. For the present iterative step, $\mathbf{d}^{G,n+1,k+1}$ is obtained as

$$\mathbf{d}^{G,n+1,k+1} = \mathbf{d}^{G,n+1,k} + d\mathbf{d}^{G,n,k} \tag{14.52}$$

Solution algorithm:

1. Generate unstructured mesh, resulting in the position vector of all the nodes and global as well as local node numbering.
2. Compute node relationship matrix $R^{e,l}$ [Eq. (14.51)].
3. Initialise $\mathbf{d}^G = 0$ and compute the geometric parameters.
4. Assume $\mathbf{d}^{G,n+1,k} = \mathbf{d}^{G,n}$ and set $k = 1$.
5. Update the displacement matrix $\mathbf{d}^{G,n+1,k+1} = \mathbf{d}^{G,n+1,k}$ and set $k = k + 1$.
6. Compute global stiffness matrix $\mathbf{K}^G$ [Eq. (14.48)] and global residual vector $\boldsymbol{\Psi}^{G,n+1,k}$ [Eq. (14.50)].
7. Solve Eq. (14.47) to obtain $d\mathbf{d}^{G,n,k}$ and update $\mathbf{d}^{G,n+1,k+1}$ using Eq. (14.52).
8. Check for the convergence. If $\left|\Psi^{n+1,k+1}\right| < \varepsilon \left|\Psi^{n+1,1}\right|$, continue to next time step and go to step 2, else set $k = k + 1$ and go to step 2.

14.3.3 Implicit Coupling Between CFD and CSD Solvers

The fluid dynamics and structural dynamics co-occur in an FSI problem. Thus, both CFD solver and CSD solver are coupled. The coupling is achieved by using the continuity condition presented in Sect. 14.2.3, which can be either explicit or implicit. Explicit coupling leads to a time lag between the fluid and structural solver, while

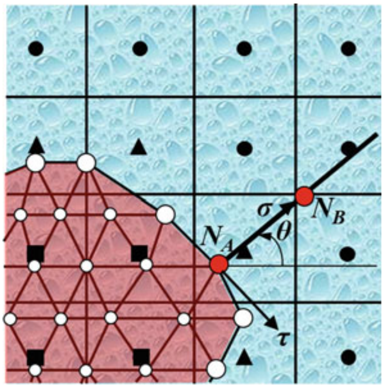


Fig. 14.6 Normal and shear stress on a node N_A on the solid surface

an iterative procedure is used in the implicit coupling to remove the time lag. The implicit coupling has better numerical stability characteristics and is essential for large deformation; thus, the implicit coupling is used in the present HLE method (Thekkethil and Sharma 2019) and presented below.

For the fluid domain, the body velocity obtained from the CSD solver is used as the boundary condition for the two-way coupled CFSD problem. For the CSD solver, the fluid dynamic forces obtained from the CFD solver are used as the boundary condition on the surface of the solid, for the two-way coupled CFSD problem. Figure 14.6 shows the fluid dynamics forces acting on a node N_A on the solid surface. The normal and shear stresses acting on the node are given as follows:

$$\sigma = \left[-p + \mu_f \frac{\partial u_n}{\partial n}\right]_{N_A}; \tau = \left[\mu_f \frac{\partial u_\tau}{\partial n}\right]_{N_A} \tag{14.53}$$

Here, the pressure p at node N_A is computed using quadratic interpolation from the neighbouring nodes. The normal derivatives of normal and tangential velocities at node N_A are computed as

$$\left[\frac{\partial u_n}{\partial n}\right]_{N_A} = \frac{u_{n,N_B} - u_{n,N_A}}{\delta}; \left[\frac{\partial u_\tau}{\partial n}\right]_{N_B} = \frac{u_{\tau,N_B} - u_{\tau,N_A}}{\delta} \tag{14.54}$$

where N_B is a point along the normal at the node N_A at a distance δ, which is equal to the finest grid size considered in the fluid domain. The normal and tangential velocities at N_B, u_{n,N_B}, and u_{τ,N_B} are computed using quadratic interpolation from the nearest fluid cells. From the normal and tangential stresses, the stresses along x- and y-directions are obtained as

$$\sigma_x = \sigma \cos\theta + \tau \sin\theta; \sigma_y = -\sigma \sin\theta + \tau \cos\theta \tag{14.55}$$

where θ is the angle of the normal at node N_A, given as

$$\theta = \tan^{-1}\left[\frac{\left(\frac{\partial\psi}{\partial y}\right)}{\left(\frac{\partial\psi}{\partial x}\right)}\right] \tag{14.56}$$

For the implicit coupling between the fluid and structural solvers, the solution is obtained iteratively until a convergence criterion is achieved for the interface variables, i.e. x-position x_{int}, y-position y_{int}, x-velocity u_{int}, y-velocity v_{int}, x-acceleration $a_{x,\text{int}}$, and y-acceleration $a_{y,\text{int}}$. The convergence criteria correspond to maximum of residual that is given as

$$\mathbf{R} = \max\left(R_{x_{\text{int}}}^{n+1,\text{new}}, R_{y_{\text{int}}}^{n+1,\text{new}}, R_{u_{\text{int}}}^{n+1,\text{new}}, R_{v_{\text{int}}}^{n+1,\text{new}}, R_{a_{x,\text{int}}}^{n+1,\text{new}}, R_{a_{y,\text{int}}}^{n+1,\text{new}}\right) < \varepsilon \tag{14.57}$$

Here, $R_{\chi_{\text{int}}}^{n+1,\text{new}}$ is the root mean square of the residuals of all the interface nodes for the present iteration, given as

$$R_{\chi_{\text{int}}}^{n+1,\text{new}} = \sqrt{\frac{1}{n_s}\sum_{i=1}^{n_{s,\text{int}}}\left(r_{\chi_{int,i}}^{n+1,\text{new}}\right)^2}, \text{ where } r_{\chi_{int,i}}^{n+1,\text{new}} = \chi_{int,i}^{n+1,\text{new}} - \chi_{int,i}^{n+1,\text{old}} \tag{14.58}$$

Here, the superscripts represent the new and old iterations. For each iteration, the interface variables are updated using an under-relaxation factor to ensure convergence. For faster convergence, *Aitken's acceleration method* (Degroote et al. 2010) is used for the under-relaxation factor after certain (three here) iterative steps, given as

$$\omega_{\chi_{\text{int}}}^{n+1,\text{new}} = \omega_{\chi_{\text{int}}}^{n+1,\text{old}} \frac{\left(\mathbf{r}_{\chi_{\text{int}}}^{n+1,\text{old}}\right)^T\left(\mathbf{r}_{\chi_{\text{int}}}^{n+1,\text{new}} - \mathbf{r}_{\chi_{\text{int}}}^{n+1,\text{old}}\right)}{\left\|\mathbf{r}_{\chi_{\text{int}}}^{n+1,\text{new}} - \mathbf{r}_{\chi_{\text{int}}}^{n+1,\text{old}}\right\|^2} \tag{14.59}$$

where ω is the under-relaxation factor and $\mathbf{r}_{\chi_{\text{int}}}$ is the interface residual vector. Using the under-relaxation factor, the updated interface variables are obtained as

$$\chi_{\text{int},i}^{n+1,\text{new}} = \chi_{\text{int},i}^{n+1,\text{old}} + \omega_{\chi_{\text{int}}}^{n+1,\text{new}} r_{\chi_{\text{int},i}}^{n+1,\text{new}} \tag{14.60}$$

For a two-way coupled CFSD problem, Fig. 14.7 shows a flow chart for the implicit coupling between the CFD and CSD solvers. For the first iteration, the interface position, velocity, and acceleration in the present time instant are considered equal to that at the previous time instant for the CFD solver. After the solution of CFD solver, the structure equations are solved using the forces obtained at the fluid–solid interface. Using the solution obtained from the CSD solver, the interface variables are updated with an under-relaxation factor [Eq. (14.60)]. Further, the fluid flow is solved using the updated interface variables. The procedure is continued until convergence

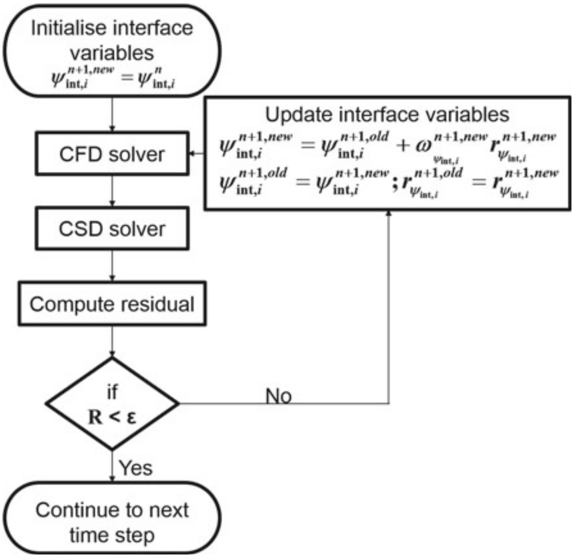

Initialise interface variables
$\psi_{\mathrm{int},i}^{n+1,new} = \psi_{\mathrm{int},i}^{n}$
CFD solver
CSD solver
Compute residual
if R < ε
No
Yes
Continue to next time step
Update interface variables
$\psi_{\mathrm{int},i}^{n+1,new} = \psi_{\mathrm{int},i}^{n+1,old} + \omega_{\psi_{\mathrm{int},i}}^{n+1,new} r_{\psi_{\mathrm{int},i}}^{n+1,new}$
$\psi_{\mathrm{int},i}^{n+1,old} = \psi_{\mathrm{int},i}^{n+1,new}; r_{\psi_{\mathrm{int},i}}^{n+1,old} = r_{\psi_{\mathrm{int},i}}^{n+1,new}$

Fig. 14.7 Flow chart for the implicit coupling between the CFD and CSD solvers, for a two-way coupled CFSD

is obtained for the residual [Eq. (14.57)]. An order of accuracy study was presented in our recent study (Thekkethil and Sharma 2019), where the order of accuracy of the present HLE method was demonstrated as second order.

14.4 HLE Method-Based CFSD Application and Analysis

Our HLE method-based CFSD application and analysis are presented here in separate subsections for rigid and flexible structure-based FSD problems.

14.4.1 CFSD Application and Analysis for Fluid–Rigid Structure Dynamics

For the one-way coupled fluid–rigid structure dynamics, CFSD application and analysis are presented here first for 2D flow across a transversely oscillating cylinder and 2D as well as 3D hydrodynamics study on fish-like propulsion of fish-like undulating foil. For the fish-like locomotion, the 2D study is presented for both tethered propulsion and self-propulsion of a fish-like pitching/undulating NACA0012 hydrofoil; the 3D study is presented for tethered propulsion of a batoid fish-like locomotion. The tethered propulsion is simulated by a constant velocity u_{∞}-based free-stream cross-flow, while a time-wise varying velocity $u_{\infty}(t)$ is used for the self-propulsion; both the velocities correspond to the propulsion velocity u_p of the foil that is constant $u_p = u_\infty$ for the tethered propulsion and time-varying $u_p\,(t) = u_\infty\,(t)$ for the self-propulsion. Here, $u_p\,(t)$ is obtained from the instantaneous thrust force, using Newton's II law of motion. The non-dimensional computational set-ups for all the one-way coupled CFSD problems are shown in Fig. 14.8.

14.4.1.1 Free-Stream Flow Across a Transverse Oscillating Cylinder

The transverse oscillating circular cylinder in a free-stream flow is a classical benchmark problem to test numerical methods for fluid flow across moving solid. The transverse oscillation is given as $y_e = A_e \sin\,(2\pi f_e t)$, where A_e is the amplitude and f_e is the frequency of oscillation. The non-dimensional parameters for the problem are the Reynolds number $\mathrm{Re} = \rho_f u_\infty D/\mu_f$, the non-dimensional amplitude A_e/D, and the frequency ratio f_e/f_o. Here, f_o is the natural frequency of vortex shedding.

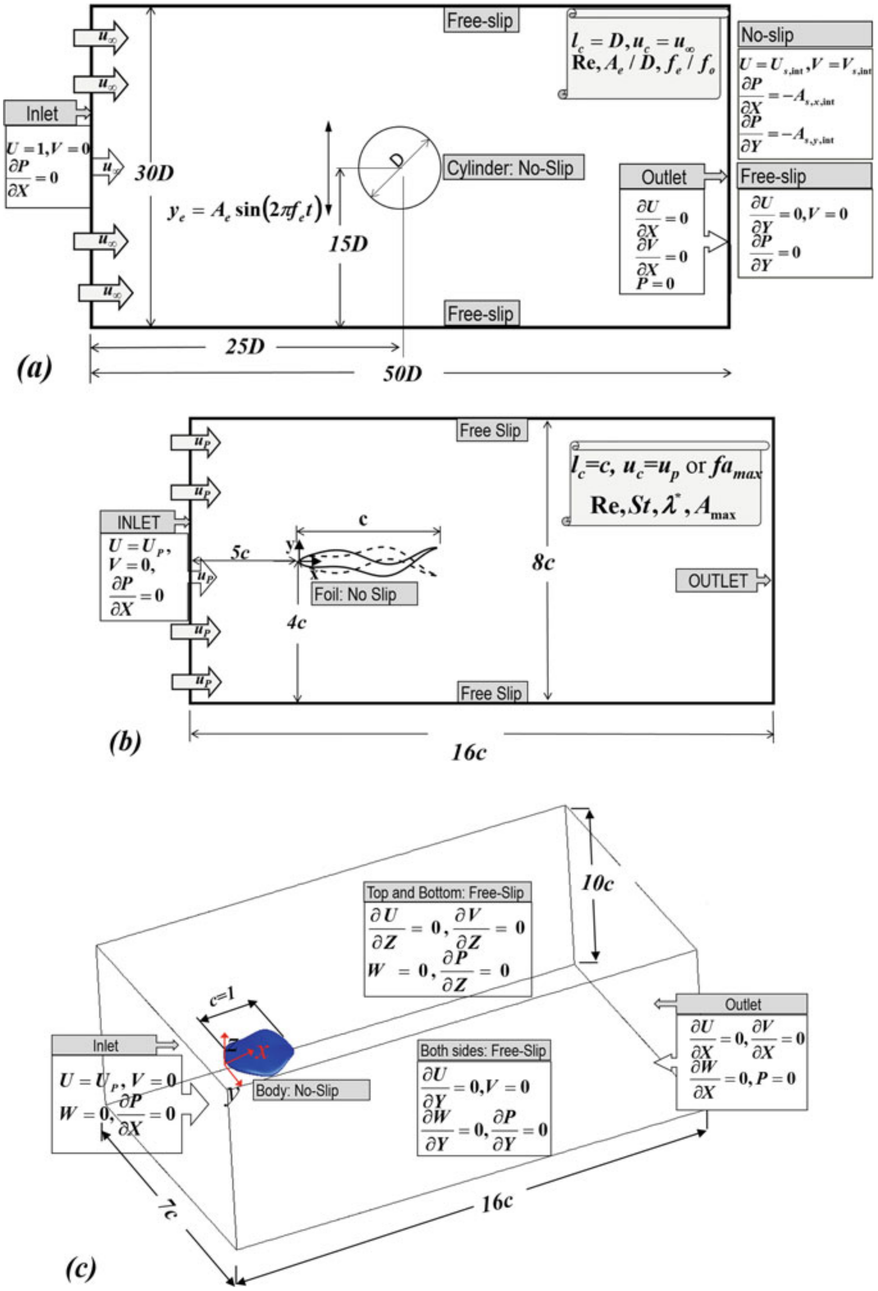
For $\mathrm{Re} = 185$, $A_e/D = 0.2$, and $f_e/f_o = 1.0$, Fig. 14.9 shows an excellent agreement between our and published (Guilmineau and Queutey 2002) results for vorticity contours and streamlines. Furthermore, our results for mean thrust coefficient $C_{Tm} = 0.432$ and RMS value of lift force coefficient $C_{Lrms} = 1.548$ match very well with respective values of 0.410 and 1.503 reported in the literature (Guilmineau and Queutey 2002).

14.4.1.2 2D Hydrodynamic Study for Tethered Propulsion and Self-propulsion of Anguilliform and Carangiform Fishes-Like Undulating Hydrofoil

LS-IBM-based hydrodynamic analysis of fishes-like tethered propulsion study of a 2D NACA0012 hydrofoil was presented in our recent study (Thekkethil et al. 2018). A fish body is modelled by the foil of chord length c, and a unified kinematic model was proposed. The model is based on the wavelength λ of a travelling wave moving along the foil. The travelling wave-based unified kinematics is represented by a lateral displacement of the centreline of the foil $\triangle y$, given as

$$\triangle y = a(x)\sin\left(\frac{2\pi x}{\lambda} - 2\pi f t\right) \text{ where } a\,(x) = a_{\max}\frac{x}{c} \tag{14.61}$$

The wave equation consists of amplitude $a(x)$ (varying from head to tail of the foil), wavelength λ, and frequency f of the travelling wave. A linear amplitude variation is considered from head to tail, with maximum amplitude at the tail as $a_{\max}$. The non-dimensional parameters for the problem are the non-dimensional wavelength λ^* $(\equiv \lambda/c)$, non-dimensional frequency St $\left(\equiv 2fa_{\max}/u_p\right)$, non-dimensional maximum amplitude $A_{\max}$ $(\equiv a_{\max}/c)$, and Reynolds number $\mathrm{Re}_{u_p} = \rho_f u_p c/\mu_f$. The unified kinematic model [Eq. (14.61)] represents various types of fishes-like kinematics—anguilliform fishes-like kinematics for the smaller non-dimensional wavelength $(\lambda^* < 1)$, caudal fin motion thunniform fishes-like kinematics for the



Free-slip
$l_c = D, u_c = u_\infty$
$\mathrm{Re}, A_e / D, f_e / f_o$
No-slip
$U = U_{s,\mathrm{int}}, V = V_{s,\mathrm{int}}$
$\frac{\partial P}{\partial X} = -A_{s,x,\mathrm{int}}$
$\frac{\partial P}{\partial Y} = -A_{s,y,\mathrm{int}}$
u_∞
Inlet
$U = 1, V = 0$
$\frac{\partial P}{\partial X} = 0$
30D
D
Cylinder: No-Slip
$y_e = A_e \sin(2\pi f_e t)$
15D
Outlet
$\frac{\partial U}{\partial X} = 0$
$\frac{\partial V}{\partial X} = 0$
$P = 0$
Free-slip
$\frac{\partial U}{\partial Y} = 0, V = 0$
$\frac{\partial P}{\partial Y} = 0$
Free-slip
25D
50D
(a)
Free Slip
$l_c = c, u_c = u_p$ or fa_{max}
$\mathrm{Re}, St, \lambda^*, A_{\max}$
u_P
INLET
$U = U_P,$
$V = 0,$
$\frac{\partial P}{\partial X} = 0$
c
5c
y
x
Foil: No Slip
4c
8c
OUTLET
Free Slip
(b)
16c
Top and Bottom: Free-Slip
$\frac{\partial U}{\partial Z} = 0, \frac{\partial V}{\partial Z} = 0$
$W = 0, \frac{\partial P}{\partial Z} = 0$
10c
c=1
z
x
y
Inlet
$U = U_P, V = 0$
$W = 0, \frac{\partial P}{\partial X} = 0$
Body: No-Slip
Both sides: Free-Slip
$\frac{\partial U}{\partial Y} = 0, V = 0$
$\frac{\partial W}{\partial Y} = 0, \frac{\partial P}{\partial Y} = 0$
Outlet
$\frac{\partial U}{\partial X} = 0, \frac{\partial V}{\partial X} = 0$
$\frac{\partial W}{\partial X} = 0, P = 0$
7c
16c
(c)

Fig. 14.8 Non-dimensional computational set-up for **a** free-stream flow across a transversely oscillating circular cylinder, **b** tethered/self-propulsion of fish-like undulating 2D NACA0012 hydrofoil, and **c** tethered propulsion of 3D batoid fish-like body

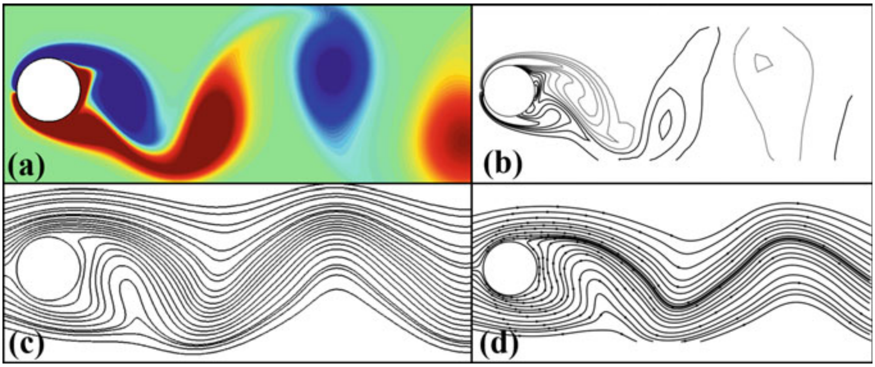


Fig. 14.9 **a**, **b** Streamlines and **c**, **d** pressure contours obtained from the **a**, **c** LS-IBM and **b**, **d** literature (Guilmineau and Queutey 2002), for the transverse oscillating cylinder in a free-stream flow at a time instant corresponding to maximum upward displacement of the cylinder, for constant $\mathrm{Re} = 185$, $A_e/D = 0.2$, and $f_e/f_o = 1.0$

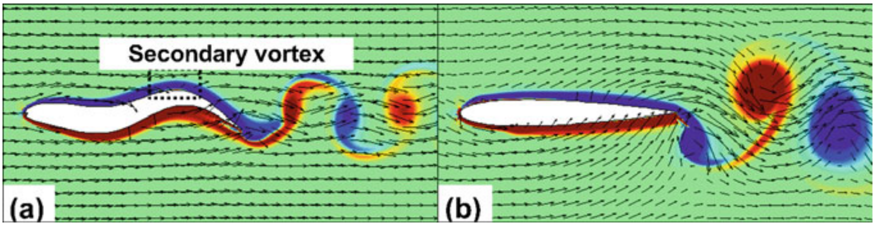


Fig. 14.10 Instantaneous vorticity contour and velocity vector during **a** anguilliform fishes-like undulation at $\lambda^* = 0.8$ and **b** carangiform fishes-like pitching at $\lambda^* = \infty$ for tethered propulsion of NACA0012 hydrofoil at $\mathrm{St} = 0.4$, $A_{\max} = 0.1$, and $\mathrm{Re}_{u_p} = 5000$

larger wavelength ($\lambda^* \to \infty$), and hypothetical fishes-like kinematics that is a combination of the anguilliform and thunniform fishes-like kinematics for the intermediate values of λ^*.

Figure 14.10 shows a reverse von Karman vortex street as a signature of thrust generation. Further, for smaller λ^* as compared to larger λ^*, the vortices are weaker and laterally stretched as compared to larger λ^*. The flow pattern results in a larger thrust force (efficiency) for larger (smaller) λ^*-based carangiform (anguilliform) fishes-like kinematics (Thekkethil et al. 2018).

For self-propelled anguilliform and carangiform fishes-like locomotion in our recent study (Thekkethil 2019) at a constant Reynolds number based on the frequency $\mathrm{Re}_f \left(\equiv \rho_f f a_{\max} c/\mu_f\right)$, Fig. 14.11 shows a temporal variation of vorticity contours and velocity vectors. For the initial time duration, the figure shows that λ^* results in a dipole formation with a strong lateral jet flow that leads to a larger hydrodynamic force and maximum stream-wise acceleration of the foil. Further, the figure shows a

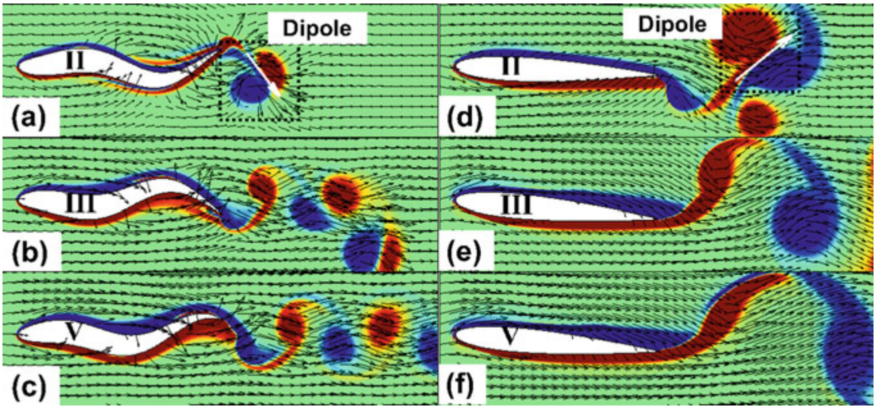


Fig. 14.11 Temporal variation of vorticity contours and velocity vectors during second, third, and fifth cycles of self-propulsion of NACA0012 hydrofoil for **a**–**c** anguilliform fishes-like undulation with $\lambda^* = 0.8$ and **d**–**f** carangiform fishes-like pitching with $\lambda^* = \infty$, at $Re_f = 1000$ and $A_{max} = 0.1$

decrease in the jet strength with time (due to the increase in the vortex spacing) that leads to a reduction in the hydrodynamic force. At the dynamic steady state, a zero net thrust force is obtained, resulting in a constant propulsion velocity.

14.4.1.3 3D Hydrodynamics Study for Tethered Propulsion of a Batoid Fishes-Like Body

Hydrodynamic analysis of various types of 3D batoid fishes-like locomotion was presented in our recent work (Thekkethil 2019). The batoid type of fishes uses a bat-like flapping of pectoral fin along with fishes-like undulation of body. The combined motion leads to a 3D kinematics. Figure 14.12 shows the shape of the batoid-like body considered in our recent study (Thekkethil 2019). The body has a hydrofoil cross section in the x-z plane with chord length c in the x-direction and an elliptical cross section in x-y and y-z planes with a span of b in the y-direction. The kinematics is a combination of the wavy motion in the x-z plane and symmetric pitching (a bird-like flapping) motion in the y-z plane. The combination of motions in the x-z and y-z planes can be represented by the transverse displacement of the body with respect to the x-y plane in dimensional form as

$$\triangle z = \frac{a_{max}}{cb/2} x|y| \sin\left[2\pi\left(\frac{x}{\lambda} - ft\right)\right] \text{ where, } x = [0, c] \text{ and } y = [-b/2, b/2] \tag{14.62}$$

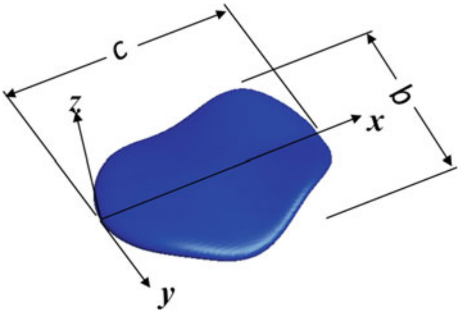


Fig. 14.12 3D view of the batoid-like body

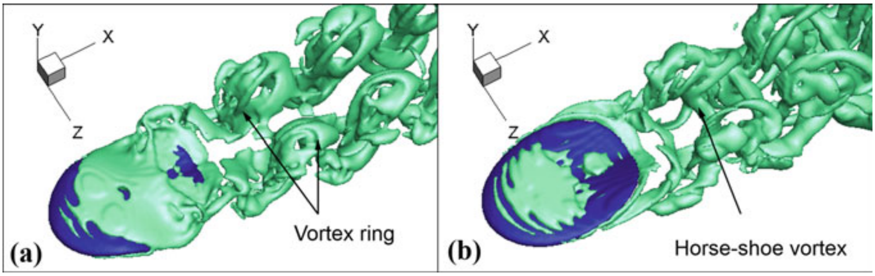


Fig. 14.13 Instantaneous Q-criterion-based vortex structure during a tethered propulsion of the batoid fishes-like body for **a** $\lambda^* = 0.8$ and **b** $\lambda^* = 4.0$, at $A_R = 0.75$, St $= 0.5$, $A_{\max} = 0.15$, and $\mathrm{Re}_{u_p} = 10{,}000$

Here, $a_{\max}$ is the maximum possible amplitude at $x = c$ and $|y| = b/2$. The wavelength of undulation λ and frequency of undulation f are similar to the 2D hydrofoil. For the 3D fishes-like locomotion, aspect ratio A_R ($\equiv b/c$) is an additional non-dimensional parameter. The various λ^* and A_R represent different types of batoid fishes-like locomotion.

For tethered propulsion of or constant propulsion-velocity-based free-stream flow across various types of 3D batoid fishes-like undulating hydrofoils, Fig. 14.13 shows instantaneous Q-criterion-based vortex structure. The figure shows a double pair of vortex rings with each pair on the front and backside connected by the vortex contrail for smaller λ^*. The two vortex rings are formed due to the symmetric pitching motion on both sides of the plane of symmetry. For the larger wavelength ($\lambda^* = 4.0$), a horseshoe vortex structure connecting the two vortex rings is present in addition to the vortex rings. This results in larger hydrodynamic forces for the larger λ^*.

14.4.2 CFSD Application and Analysis for Fluid–Flexible Structure Dynamics

As discussed above, for the fluid–flexible structure dynamics, the interaction between fluid dynamics and structural dynamics is two-way coupled—the fluid flow and structure motion/deformation are dependent on each other. In this section, the application of the present HLE method is presented for three problems: first, a lid-driven cavity-based flow across a flexible plate; second, a Poiseuille flow across a rigid cylinder with a flexible splitter plate; and third, tethered-propulsion-based free-stream flow across a flexible hydrofoil. The first problem is a computationally efficient benchmark problem, recently proposed by us (Thekkethil and Sharma 2019), the second problem is also a commonly used benchmark problem, and the third problem is an extension of our study on hydrodynamics during fishes-like locomotion. The non-dimensional computational set-ups for the two-way coupled CFSD problems are shown in Fig. 14.14.

14.4.2.1 Lid-Driven Cavity Flow-Based Benchmark Problem for Fluid–Flexible Structure Dynamics

We recently proposed (Thekkethil and Sharma 2019) a computationally efficient and easy-to-set up lid-driven cavity flow-based benchmark problem along with benchmark solutions for the two-way coupled FSD. The problem considers a square lid-driven cavity, with cavity length L, and both top and bottom wall act as a lid moving with a constant velocity. A flexible plate, of length $0.5L$ and thickness $0.05L$ hinged at the centre of the cavity, gets deformed due to the lid-driven cavity flow-based hydrodynamic force. In addition to the Reynolds number Re, the two-way coupled FSD problem considers the non-dimensional Young's modulus E^*, density ratio ρ_r, and Poisson's ratio ν_s as non-dimensional governing parameters.

Figure 14.15 shows the steady-state streamlines and pressure as well as vorticity contours. The lid-driven flow creates circular flows near the top and bottom boundaries of the cavity, which results in symmetric bending of the plate, as shown in the figure. The computational time taken for this problem is very small as compared to many benchmark problems reported in the literature.

14.4.2.2 Poiseuille Flow Across a Flexible Splitter Plate Behind a Cylinder

The problem corresponds to a hydrodynamically fully developed flow across a rigid circular cylinder of diameter D with a flexible splitter plate (of a thickness of $0.2D$ and length $3.5D$ attached behind it) in a channel. The problem was first proposed by Turek and Hron (2006) and is widely used as a benchmark problem in the literature.

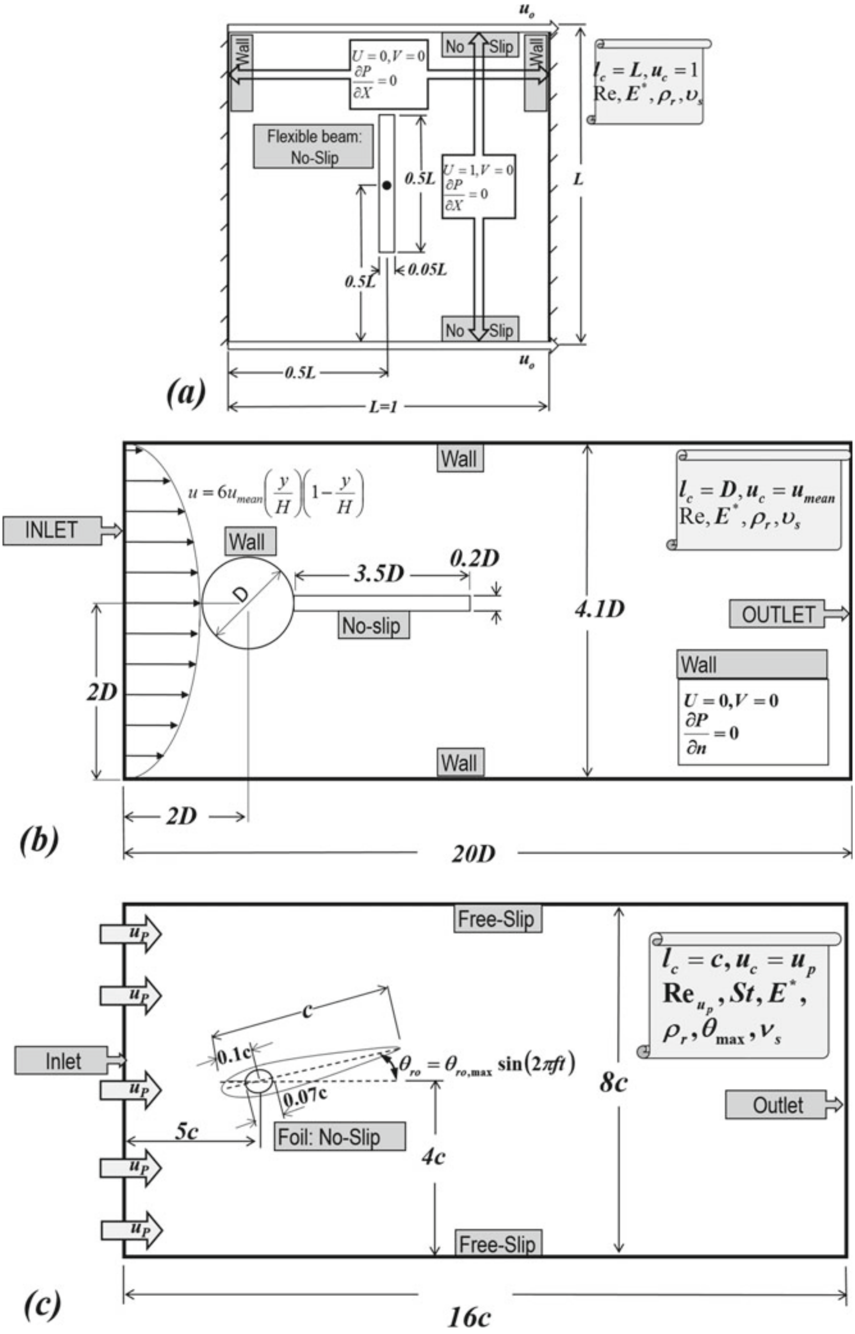

u_o
Wall
Wall
$U=0, V=0$
$\frac{\partial P}{\partial X}=0$
No Slip
$l_c = L, u_c = 1$
$\mathrm{Re}, E^*, \rho_r, \upsilon_s$
Flexible beam: No-Slip
0.5L
$U=1, V=0$
$\frac{\partial P}{\partial X}=0$
L
0.5L
0.05L
No Slip
u_o
0.5L
(a)
L=1
Wall
$u = 6u_{mean}\left(\frac{y}{H}\right)\left(1-\frac{y}{H}\right)$
$l_c = D, u_c = u_{mean}$
$\mathrm{Re}, E^*, \rho_r, \upsilon_s$
INLET
Wall
D
3.5D
0.2D
4.1D
OUTLET
No-slip
Wall
$U=0, V=0$
$\frac{\partial P}{\partial n}=0$
2D
Wall
2D
(b)
20D
Free-Slip
u_P
$l_c = c, u_c = u_p$
$\mathrm{Re}_{u_p}, St, E^*,$
$\rho_r, \theta_{\max}, \nu_s$
c
Inlet
0.1c
$\theta_{ro} = \theta_{ro,\max}\sin(2\pi ft)$
8c
0.07c
Outlet
5c
Foil: No-Slip
4c
Free-Slip
(c)
16c

Fig. 14.14 Non-dimensional computational set-up for **a** the benchmark problem on the classical lid-driven cavity flow-induced deformation of a hinged vertical plate, **b** a rigid circular cylinder with a flexible splitter plate in a Poiseuille flow, and **c** tethered propulsion of structurally flexible hydrofoil subjected to pitching motion

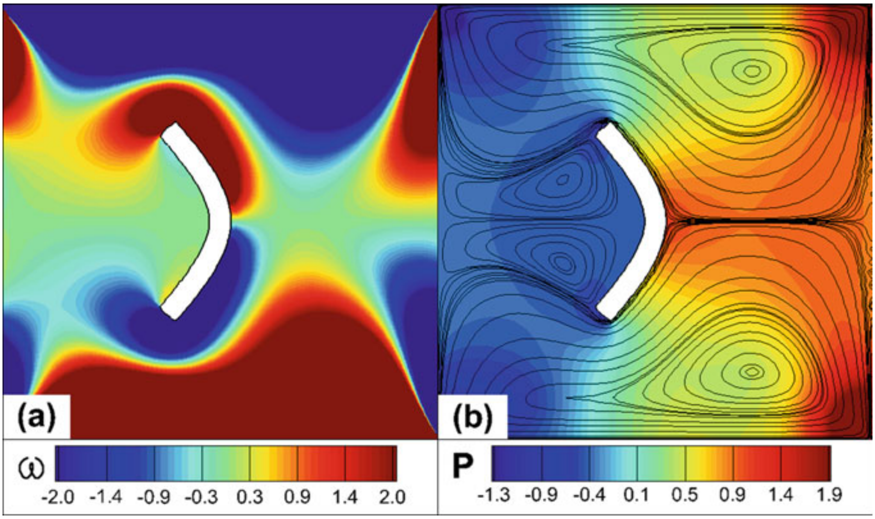


Fig. 14.15 Steady-state **a** vorticity contours and streamlines and **b** pressure contours, for hinged plate in a top and bottom lid-driven cavity at Re = 100, $E^* = 100$, $\rho_r = 10$, and $\nu_s = 0.3$

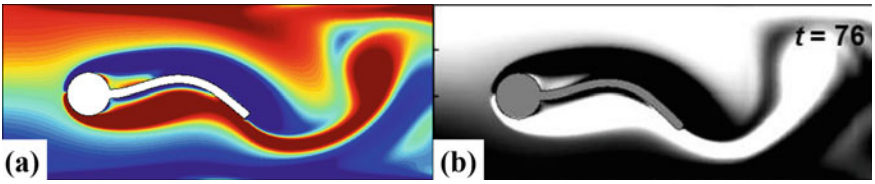


Fig. 14.16 Instantaneous vorticity contour obtained from the **a** LS-IBM and **b** literature (Bhardwaj and Mittal 2012), for the channel flow across flexible splitter plate attached behind rigid circular cylinder at a time instant $t = 76$, for constant Re = 100, $E^* = 1400$, $\rho_r = 10$, and $\nu_s = 0.4$

Figure 14.16 shows an excellent agreement between our (Thekkethil 2019) results and published (Bhardwaj and Mittal 2012) results for a periodic state. Due to the time-wise periodic hydrodynamic forces acting on the body, the plate is subjected to vibration and the periodic state is obtained after a certain number of vortex shedding cycles.

14.4.2.3 2D Hydrodynamics Study for Tethered Propulsion of a Fish-Like Pitching Flexible Hydrofoil

The arrangement of the flexible hydrofoil is shown in Fig. 14.14c. This was proposed in an experimental work (Marais et al. 2012) and was studied numerically in our recent work (Thekkethil 2019).

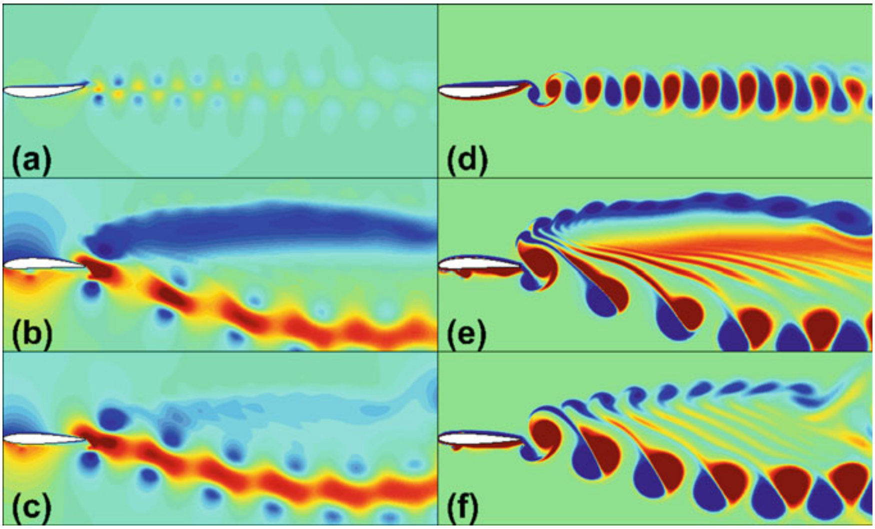


Fig. 14.17 Instantaneous **a**–**c** stream-wise velocity contours and **d**–**f** vorticity contours for the free-stream flow across pitching flexible hydrofoil with **a**, **d** largely flexible ($E^* = 5000$), **b**, **e** moderately flexible ($E^* = 30{,}000$), and rigid ($E^* = \infty$) hydrofoils, at $\mathrm{St} = 0.5$, $\mathrm{Re} = 5000$, $\theta_{\mathrm{ro,max}} = 8^o$, $\rho_r = 1.0$, and $\nu_s = 0.4$

For the tethered-propulsion-based free-stream flow across the flexible pitching hydrofoil, Fig. 14.17 shows the vorticity contours and stream-wise velocity contours. The figure shows a single straight-jet flow (straight reverse von Karman vortex street) for the largely flexible foil, inclined jet flow (inclined von Karman vortex street) for the rigid foil, and inclined jet flow along with a straight-wake (inclined reverse von Karman vortex street with straight von Karman vortex street) for the moderately flexible foil. The moderate (large) flexibility results in maximum (minimum) thrust generation.

14.5 Closure

This chapter is presented in two parts: first, CFSD development, and second, CFSD application and analysis. For the first part on CFSD development, a detailed numerical methodology in two-dimensional Cartesian coordinate system is presented for the partitioned approach-based hybrid Lagrangian–Eulerian (HLE) method. The methodology is based on physical law-based FVM and level-set-based IBM for CFD development and geometric nonlinear Galerkin FEM-based CSD development, along with an implicit coupling between the CFD and CSD solvers that is numerically stable for large deformation. The second part demonstrates the HLE method-based CFSD application (with the help of computational set-up) and analysis (of hydrodynamic results) on a variety of rigid and flexible structure-based one-way and two-way coupled CFSD problems, respectively.

References

Belytschko T, Kennedy JM (1975) Finite element study of pressure wave attenuation by reactor fuel subassemblies. J Press Vessel Technol 97(3):172–177

Bhardwaj R, Mittal R (2012) Benchmarking a coupled immersed-boundary-finite-element solver for large-scale flow-induced deformation. AIAA J 50(7):1638–1642

Degroote J, Haelterman R, Annerel S, Bruggeman P, Vierendeels J (2010) Performance of partitioned procedures in fluid-structure interaction. Comput Struct 88(7–8):446–457

Donea J, Fasoli-Stella P, Giuliani S (1976) Finite element solution of transient fluid-structure problems in Lagrangian coordinates. Technical report

Guilmineau E, Queutey P (2002) A numerical simulation of vortex shedding from an oscillating circular cylinder. J Fluids Struct 16(6):773–794

Jacobson A, Kavan L, Sorkine-Hornung O (2013) Robust inside-outside segmentation using generalized winding numbers. ACM Trans Graph (TOG) 32(4):33

Majumdar S, Iaccarino G, Durbin P(2001) Rans solvers with adaptive structured boundary nonconforming grids. In: Annual research briefs, Center for Turbulence Research, Stanford University, pp 353–466

Marais C, Benjamin T, José EW (2012) Godoy-Diana R (2012) Stabilizing effect of flexibility in the wake of a flapping foil. J. Fluid Mech. 710:659–669

Mittal R, Iaccarino G (2005) Immersed boundary methods. Annu Rev Fluid Mech 37:239–261

Mittal R, Dong H, Bozkurttas M, Najjar FM, Vargas A, Von Loebbecke A (2008) A versatile sharp interface immersed boundary method for incompressible flows with complex boundaries. J Comput Phys 227(10):4825–4852

Namshad T (2019) Computational fluid-structure dynamics development and its application for analysis of various types of 2D/3D fishes-like kinematics, propulsion, & flexibility. Ph.D. thesis, Indian Institute of Technology Bombay

Noh WF (1963) CEL: a time-dependent, two-space-dimensional, coupled Eulerian-Lagrange code. Technical report, Lawrence Radiation Laboratory, University of California, Livermore

Pan D (2006) An immersed boundary method for incompressible flows using volume of body function. Int J Numer Methods Fluids 50(6):733–750

Patankar S (2018) Numerical heat transfer and fluid flow. CRC Press, Boca Raton

Patel JK, Natarajan G (2018) Diffuse interface immersed boundary method for multi-fluid flows with arbitrarily moving rigid bodies. J Comput Phys 360:202–228

Peskin CS (2002) The immersed boundary method. Acta Numer 11:479–517

Sethian JA (1999) Level set methods and fast marching methods: evolving interfaces in computational geometry, fluid mechanics, computer vision, and materials science, vol 3. Cambridge University Press, Cambridge

Sharma A (2017) Introduction to computational fluid dynamics: development, application and analysis. Wiley & Athena, UK; Ane Books Pvt. Ltd. New Delhi

Shrivastava M, Agrawal A, Sharma A (2013) A novel level set-based immersed-boundary method for CFD simulation of moving-boundary problems. Numer Heat Transf B 63(4):304–326

Thekkethil N, Sharma A (2019) Level set function based immersed interface method and benchmark solutions for fluid flexible-structure interaction. Int J Numer Methods Fluids 91(3):134–157

Thekkethil N, Sharma A, Agrawal A (2018) Unified hydrodynamics study for various types of fishes-like undulating rigid hydrofoil in a free stream flow. Phys Fluids 30(7):077107

Turek S, Hron J (2006) Proposal for numerical benchmarking of fluid-structure interaction between an elastic object and laminar incompressible flow. In: Fluid-structure interaction. Springer, Berlin, pp 371–385

Udaykumar HS, Mittal R, Rampunggoon P, Khanna A (2001) A sharp interface Cartesian grid method for simulating flows with complex moving boundaries. J Comput Phys 174(1):345–380

Versteeg HK, Malalasekera W (2007) An introduction to computational fluid dynamics: the finite volume method. Pearson Education

Zienkiewicz OC, Taylor RL, Nithiarasu P, Zhu JZ (1977) The finite element method, vol 3. McGraw-Hill, London