Accepted Manuscript

A fictitious domain/distributed Lagrange multiplier based fluid-structure interaction scheme with hierarchical B-Spline grids

C. Kadapa, W.G. Dettmer, D. Perić

PII: S0045-7825(15)00429-6
DOI: http://dx.doi.org/10.1016/j.cma.2015.12.023
Reference: CMA 10803

To appear in: *Comput. Methods Appl. Mech. Engrg.*

Received date: 11 August 2015
Revised date: 18 December 2015
Accepted date: 22 December 2015

Please cite this article as: C. Kadapa, W.G. Dettmer, D. Perić, A fictitious domain/distributed Lagrange multiplier based fluid-structure interaction scheme with hierarchical B-Spline grids, *Comput. Methods Appl. Mech. Engrg.* (2015), http://dx.doi.org/10.1016/j.cma.2015.12.023

A fictitious domain/distributed Lagrange multiplier based fluid-structure interaction scheme with hierarchical B-Spline grids

C. Kadapa*, W.G. Dettmer, D. Perić

Zienkiewicz Centre for Computational Engineering, College of Engineering, Swansea University, Fabian Way, Swansea, SA1 8EN, Wales, UK.

Abstract

We present a numerical scheme for fluid-structure interaction based on hierarchical B-Spline grids and fictitious domain/distributed Lagrange multipliers. The incompressible Navier-Stokes equations are solved over a Cartesian grid discretised with B-Splines. The fluid grid near the immersed solids is refined locally using hierarchical B-Splines. The immersed solid is modelled as geometrically-exact beam discretised with standard linear Lagrange shape functions. The kinematic constraint at the fluid-solid interface are enforced with distributed Lagrange multipliers. The unconditionally-stable and second-order accurate generalised-α method is used for integration in time for both the fluid and solid domains. A fully-implicit and fully-coupled solution scheme is developed by using Newton-Raphson method to solve the non-linear system of equations obtained with Galerkin weak formulation. First, the spatial and temporal convergence of the proposed scheme is assessed by studying steady and unsteady flow past a fixed cylinder. Then, the scheme is applied to several benchmark problems to demonstrate the efficiency and robustness of the proposed scheme. The results obtained with the present scheme are compared with the reference values.

Keywords: Fictitious domain methods; Immersed boundary methods; Fluid-structure interaction; Hierarchical B-Splines; Distributed Lagrange multipliers; Flow-induced vibrations.

*Corresponding author
Email address: c.kadapa@swansea.ac.uk (C. Kadapa)

1. Introduction

Fluid-structure interaction is a phenomenon frequently encountered in the fields of science and engineering. Many factors, such as properties of the fluid and structure, extent of deformations of the structure, and instabilities due to added-mass, influence the development and applicability of a numerical scheme for simulating FSI problems. In the traditional arbitrary Lagrangian-Eulerian (ALE) approach with body-fitted meshes, which is extensively studied and understood (see Chapter 14 in [1] and references therein), the fluid is solved on a body-fitted mesh which is either adjusted or re-meshed depending upon the extent of the deformations of the solid, see [29, 53]. However, ALE comes with several disadvantages: a.) it requires the generation of body-fitted meshes which is a cumbersome task for complicated geometries, b.) the fluid mesh needs to be updated or re-meshed depending upon the extent of solid deformations, c.) every re-meshing step involves a data-mapping from old mesh to the new mesh which is also prone to errors. Hence, the applicability of ALE formulation is limited by the ease of generating body-fitted meshes and the robustness of re-meshing algorithms. Moreover, for more demanding fluid-structure interaction applications involving topological changes, e.g., self-contacts in structural model, simulation of check-valves and multiple fibres submerged in flow, ALE formulation may fail because of zero-volume elements when the structural parts are in contact. Extending such numerical schemes for FSI based on body-fitted meshes to 3D problems where the solid undergoes extreme deformations is a challenging task, for which it is difficult to ensure the robustness of the scheme. To overcome these difficulties alternate solution strategies based on fixed Cartesian grids are being explored.

The immersed or embedded or non-body-fitted or Cartesian grid based methods are simpler, easy to implement and computationally more efficient than the methods based on body-fitted meshes for problems where the solids undergo huge deformations and/or topological changes and multiphase and mixing flows. In these type of methods the fluid is modelled in an Eulerian frame of reference and the solid is modelled in a Lagrangian frame. The solid, that may either be fixed or undergoing extreme deformations and/or topological

changes, is immersed into the fluid grid with discretisation that does not need to match the solid boundaries. The interface conditions at the fluid-solid interface are enforced via several techniques and it is this technique that distinguishes different immersed methods. To our knowledge, immersed boundary methods (IBM) introduced and pioneered by Peskin [46] is the first research work carried out in the direction of non-body-fitted meshes. In [46] and its variation [39, 41, 49, 52] the kinematic constraint at fluid-solid interface is enforced using body-force approach. The body-force is computed assuming that the Lagrange points are connected to artificial springs with high stiffness values. This method restricts the time steps to small values irrespective of whether the fluid solver is implicit or explicit. Later, immersed interface method (IIM) was introduced by [35, 36, 38, 44] in which derivatives in the cells cut the boundary of the immersed solid are modified in order to accommodate the jumps in velocity and/or pressure. Due to this modification process IIM is applicable only to FSI problems with bulky solids. Historically, in majority of the research work carried out with IBM and IIM the fluid problem is solved using finite-difference and finite-volume grids which lack local refinement capability.

IBM and IIM based on standard finite element meshes are studied in [4, 37]. Zhang and Gay [64], Yao et al. [61] and Zhang et al. [65] studied immersed finite element methods for fluid-structure interaction problems. However, the amount of research in such methods is limited and most of these methods still inherit the disadvantages of Peskin's immersed boundary method ([46]). For example, the way the interacting forces are computed and velocities are interpolated from fluid mesh to solid mesh and vice-versa, restricts the time steps to very small values.

Höllig [25, 26] used B-Splines for the first time in the context of immersed finite element methods and developed weighted extended B-Splines (WEB-Splines). Later, this concept was extended by [50, 51, 55] to FSI problems. Though this method seems to be promising to simulate FSI problems, it involves a basis function modification algorithm in order to tackle instabilities due to the presence of small cut-cells. Also, for problems involving thin structures this approach poses several difficulties in identifying and modifying the basis functions. To overcome these limitations, the fictitious domain method (FDM) pioneered

by Glowinski [15, 18–23, 43, 45] seems to be an efficient alternative. FDM is another class of embedded methods where the kinematic constraint at the fluid-solid interface is enforced using Lagrange multipliers. FDM offers several advantages over the classical IBM ([46]), IIM ([35]) and WEB-Spline method ([25]). While the kinematic constraint at the fluid-solid interface is applied weakly in IBM, it is applied strongly in FDM using Lagrange multipliers. Moreover, the Lagrange multipliers are tractions on the boundary of the immersed body which can be used directly for FSI problems. Furthermore, in FDM, there is no need to modify the basis functions of the fluid grid in order to ensure the cut-cell stabilisation as the fluid is solved everywhere in background fluid grid. So far, in the literature, the fluid grid in FDM is discretised with the standard Lagrange polynomials — Taylor-Hood or Crouziex-Raviart family elements [3] — with or without bubble functions [42]. In this work we propose a fictitious domain formulation for simulating FSI problems based on hierarchical B-Spline grid. The numerical scheme proposed in this paper can be considered as an extension of immersogeometric framework described by [33] in the sense that the non-uniform rational B-splines (NURBS) used in [30] to discretise the background fluid grid are replaced here with B-Splines to discretise the fluid problem and hierarchical B-Splines to enrich the fluid grid near immersed bodies. The main motivation behind using hierarchical B-Splines is that use of local refinement results in significant savings in computational time as already proven in [5, 56].

This research work is focussed on fluid-structure interaction phenomenon where the structures are usually thin, for example., heart-valves, filaments, flags, parachutes, pulp fibres. So, the flexible solid structure is modelled as infinitely thin line using geometrically-exact beam formulation (see Chapter 17 in [67]).

This paper is organised as follows. In Section 2, we give a brief overview of B-Splines and their hierarchical refinement. In Section 3 we give a short introduction and advantages of FDM. In Section 4 we describe the formulation for fluid-flexible solid interaction. In Section 5, the generalised-α time integration schemes are discussed. Section 6 gives a brief description of geometrically-exact formulation for the beam. Numerical examples are presented in Section 7. First, we study the conservation properties of the proposed scheme.

Then, we validate the proposed scheme by studying the steady and unsteady flow past a circular cylinder. Later, we study several benchmark examples using the proposed numerical scheme and analyse the simulation results obtained. We conclude this paper with Section 8 by drawing conclusions and pointing out directions for further research.

2. Hierarchical B-Splines

In this work we use hierarchical B-Splines for spatial discretisation of the Cartesian grid for the fluid. We present a brief overview of B-Splines and their hierarchical refinement in this section. For a detailed discussion on B-Splines the reader is suggested to refer to the standard books on non-uniform rational B-splines (NURBS) by Piegl and Tiller [47] and Rogers [48].

2.1. Univariate B-Splines

B-Splines are piecewise-continuous polynomial functions. For a given knot vector $\boldsymbol{\Xi} = \{\xi_0, \ldots, \xi_{n+a+1}\}$ and degree of polynomial a, the univariate B-Spline functions are evaluated by the recurrence relations,

$$N_{i,0}(\xi) = \begin{cases} 1 & \text{if } \xi_i \leq \xi \leq \xi_{i+1} \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

$$N_{i,p}(\xi) = \frac{\xi - \xi_i}{\xi_{i+p} - \xi_i} N_{i,p-1}(\xi) + \frac{\xi_{i+p+1} - \xi}{\xi_{i+p+1} - \xi_{i+1}} N_{i+1,p-1}(\xi) \tag{2}$$

A B-Spline basis function $N_{i,a}$ of degree a spans from knot ξ_i to knot ξ_{i+a+1} as shown in Fig. 1 and is at least $\mathcal{C}^{a-1}$ continuous in (ξ_i, ξ_{i+a+1}). Uniform B-Splines are considered throughout this paper, i.e. $\Delta\xi = \xi_{i+1} - \xi_i$ is constant throughout the domain (when there is no local refinement).

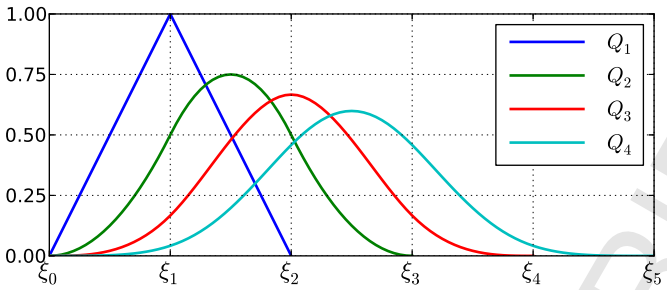


Figure 1: Univariate B-Spline basis functions.

2.2. B-Splines in higher dimensions

One of the remarkable advantages of B-Splines is the ease with which they can be extended to higher dimensions. This is done using tensor products. With $\mathbf{N}_\xi$, $\mathbf{N}_\eta$ and $\mathbf{N}_\zeta$ as the univariate B-Spline functions in ξ, η, and ζ parametric directions, respectively, the multivariate B-Spline basis functions in two- and three-dimensions are given as,

$$\mathbf{N}(\xi,\eta) = \mathbf{N}_\xi \otimes \mathbf{N}_\eta \qquad \text{in} \quad 2\text{D} \tag{3}$$

$$\mathbf{N}(\xi,\eta,\zeta) = \mathbf{N}_\xi \otimes \mathbf{N}_\eta \otimes \mathbf{N}_\zeta \qquad \text{in} \quad 3\text{D} \tag{4}$$

The support of a 2D B-Spline basis function that is of equal polynomial order in both parametric directions is illustrated in Fig. 2.

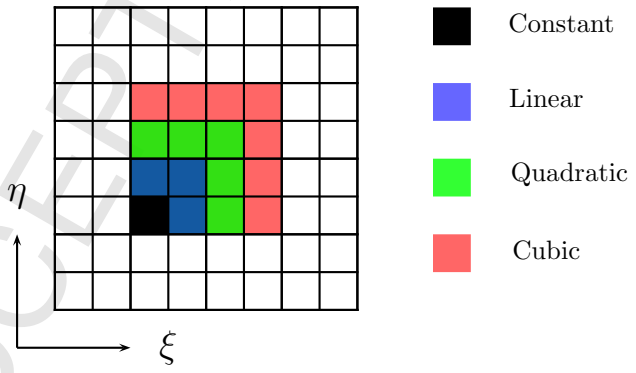


Figure 2: Support of a equal-order B-Spline basis function in 2D.

2.3. B-Spline subdivision and their two-scale relation

The most important property of the B-Spline functions in the context of the present work is their subdivision property which is also known widely as *two-scale* relation. According to this remarkable property, a single B-Spline function can be written as a sum of scaled and translated copies of itself. For a general B-Spline function N_a, the *two-scale* relation is written as,

$$N_a(\xi) = \sum_{i=0}^{a+1} \alpha_i N_a(2\xi - i) \tag{5}$$

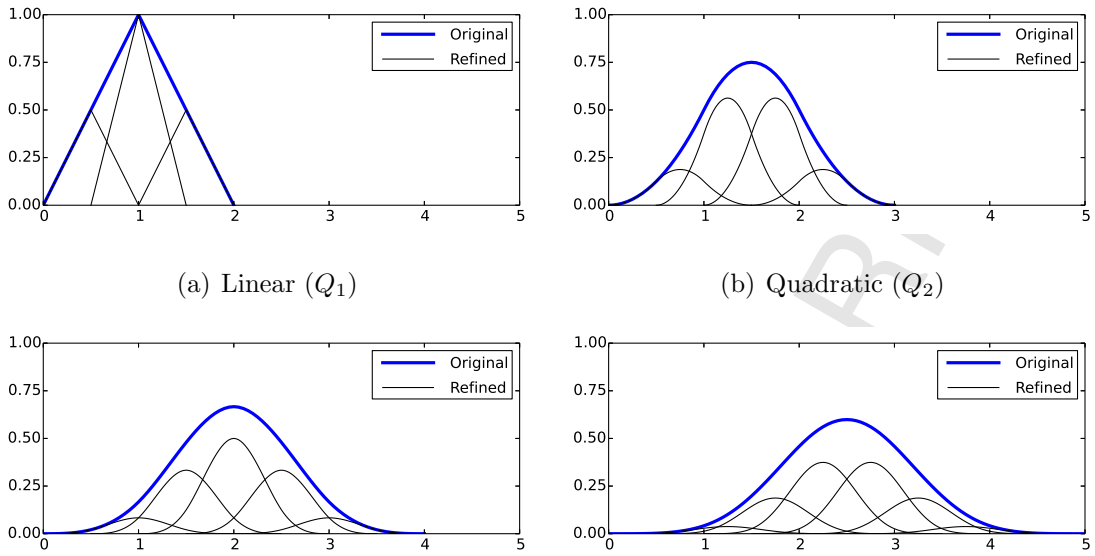
where, α_i are functions of Binomial coefficients, given as,

$$\alpha_i = \frac{1}{2^a} \binom{a+1}{i} \tag{6}$$

In the context of hierarchical refinement, this property can be restated as: a B-Spline function on a knot vector with knot span $\Delta\xi$ can be evaluated as a linear combination of B-Spline functions defined on a knot vector with knot span $\Delta\xi/2$. That is, B-Spline basis functions $\mathbf{N}_k(\xi)$ at level k can be written as a linear combination of B-Spline basis functions $\mathbf{N}_{k+1}(\xi)$ at level k+1. Mathematically,

$$\mathbf{N}_k(\xi) = \mathbf{S}\, \mathbf{N}_{k+1}(\xi) \tag{7}$$

where, $\mathbf{S}$ is the subdivision matrix which contains the coefficients α from Eq. (5). The *two-scale* relation is schematically illustrated in Fig. 3 for linear, quadratic, cubic and quartic B-Splines. This property of B-Spline functions is utilised to evaluate the B-Spline functions at coarse levels from those at refined levels, when performing numerical integration and post-processing.



(a) Linear (Q_1) (b) Quadratic (Q_2)

(c) Cubic (Q_3) (d) Quartic (Q_4)

Figure 3: Two-scale relation of the B-Spline functions.

2.4. Hierarchical refinement

The main motivation behind using hierarchical refinement is to improve the computational efficiency. This is achieved by *locally* refining the B-Spline grid near the immersed bodies thereby reducing the total number of DOF. The computational advantages of hierarchical refinement have already been demonstrated in [5, 56]. The hierarchical refinement of B-Splines is a difficult task involving complex algorithms, see [5, 56]. However, it can be implemented quite elegantly and robustly using *trees*, widely used data structures in computer science [10, 54], and the concepts of *templates* in the programming language C++ [58]. In this work we have adopted the same approach as described by Schillinger et al. [56] and suggest the reader to refer the same for the detailed description of the algorithms involved in hierarchical refinement of B-Splines.

The amount of time required to produce the local refinement is negligible when compared to the total computational time of the whole simulation. Moreover, in all of these examples presented in this paper the local refinement is performed only once and is kept constant throughout the simulation, which makes the time cost of local refinement even more insignificant.

3. Fictitious domain method

The fictitious domain methods (FDM) are a class of domain embedding methods used for numerical solutions of partial differential equations. These methods have been extensively studied in Glowinski [15, 18–23, 43, 45] for particle flows and other fluid-structure interaction phenomena. In FDM, the fluid domain is extended into the interior of the solid domain and the fluid equations are solved throughout the entire Cartesian grid. As illustrated in Fig. 4, a solid body Ω_b, with its boundary Γ_b that may or may not change in time, is placed over the top of a fluid domain Ω_f. As a result, the fluid mesh does not have to match at the fluid-solid interface and hence, a Cartesian grid can be used for the fluid domain. The fluid is modelled using Eulerian approach and the solid is represented with Lagrangian description. The kinematic constraint at the interface between fluid and the solid is enforced via Lagrange multipliers.

The advantages of FDM over their body-fitted counterparts can be summarized as:

- there is no need for complicated and time-consuming body-fitted meshes. So, the discretisation is easy and can be parallelised effectively.
- as there is no body-fitted mesh, complicated unstructured re-meshing, due to excessive displacements and distortions of the solid, is totally avoided.
- the numerical scheme is free of data-mapping errors that is otherwise present in body-fitted ALE schemes where re-meshing is required.
- properties of structured grids can be exploited in developing efficient parallel solvers, for example parallel multigrid preconditioners, when compared with unstructured grids.
- FSI problems with topological changes and mixing and multiphase flows can be simulated efficiently.

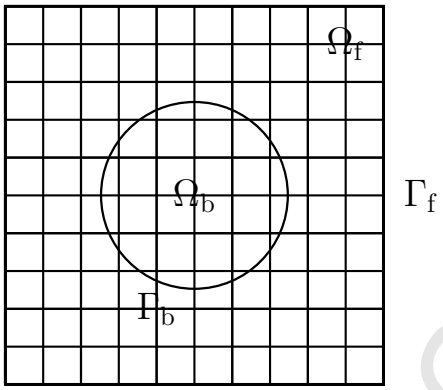


Figure 4: Fictitious domain method - schematic description.

4. Formulation

4.1. Governing equations

4.1.1. Governing equations for the fluid

For an incompressible viscous fluid the initial-boundary value problem is stated as: Given $\mathbf{g}^f : \Omega^f \to \mathbb{R}^3$; $\bar{\mathbf{v}}^f : \Gamma_D^f \to \mathbb{R}^3$; and $\bar{\mathbf{t}}^f : \Gamma_N^f \to \mathbb{R}^3$, find velocity, $\mathbf{v}^f : \Omega \to \mathbb{R}^3$; and pressure, $p : \Omega \to \mathbb{R}$, such that:

$$\rho^f \frac{\partial \mathbf{v}^f}{\partial t} + \rho^f (\mathbf{v}^f \cdot \nabla)\mathbf{v}^f - \mu^f \Delta \mathbf{v}^f + \nabla p = \mathbf{g}^f \qquad \text{in} \quad \Omega^f \tag{8a}$$

$$\nabla \cdot \mathbf{v}^f = 0 \qquad \text{in} \quad \Omega^f \tag{8b}$$

$$\mathbf{v}^f = \bar{\mathbf{v}}^f \qquad \text{on} \quad \Gamma_D^f \tag{8c}$$

$$\mathbf{t}^f = \boldsymbol{\sigma}^f \cdot \mathbf{n}^f = \bar{\mathbf{t}}^f \qquad \text{in} \quad \Gamma_N^f \tag{8d}$$

$$\mathbf{v}^f(\cdot, 0) = \mathbf{v}_0^f \qquad \text{in} \quad \Omega^f \tag{8e}$$

where, ρ^f is the density of the fluid, μ^f is the dynamic viscosity of the fluid, $\mathbf{g}^f$ is the body force, $\mathbf{n}^f$ is the unit outward normal on the boundary, Γ^f, of Ω^f and the pseudo-stress $\boldsymbol{\sigma}^f$ is given by,

$$\boldsymbol{\sigma}^f = \mu \nabla \mathbf{v}^f - p\,\mathbf{I} \tag{9}$$

Γ_D^f is the part of the boundary of the domain Ω^f where Dirichlet boundary condition $\bar{\mathbf{v}}^f$ is applied and Γ_N^f is the part of the boundary of the domain Ω^f where Neumann boundary condition $\bar{\mathbf{t}}^f$ is applied. Here, $\Gamma^f = \Gamma_D^f \cup \Gamma_N^f$ and $\Gamma_D^f \cap \Gamma_N^f = \emptyset$, while $\mathbf{v}_0^f$ is the initial velocity of the fluid in the domain Ω^f.

4.1.2. Governing equations for the solid

The initial-boundary value problem of elasticity, in the current configuration, is stated as:

Given $\mathbf{g}^s : \Omega \to \mathbb{R}^3$; $\bar{\mathbf{d}}^s : \Gamma_D \to \mathbb{R}^3$; and $\bar{\mathbf{t}}^s : \Gamma_N \to \mathbb{R}^3$, find displacement $\mathbf{d}^s : \Omega \to \mathbb{R}^3$, such that:

$$\rho^s \frac{\partial^2 \mathbf{d}^s}{\partial t^2} + \nabla \cdot \boldsymbol{\sigma}^s = \mathbf{g}^s \qquad \text{in} \quad \Omega^s \tag{10a}$$

$$\mathbf{d}^s = \bar{\mathbf{d}}^s \qquad \text{on} \quad \Gamma_D^s \tag{10b}$$

$$\mathbf{t}^s = \boldsymbol{\sigma}^s \cdot \mathbf{n}^s = \bar{\mathbf{t}}^s \qquad \text{in} \quad \Gamma_N^s \tag{10c}$$

$$\mathbf{d}^s(\cdot, 0) = \mathbf{d}_0^s \qquad \text{in} \quad \Omega^s \tag{10d}$$

$$\dot{\mathbf{d}}^s(\cdot, 0) = \dot{\mathbf{d}}_0^s \qquad \text{in} \quad \Omega^s \tag{10e}$$

where, ρ^s is the density of the solid, $\mathbf{g}^s$ is the body force on the solid and $\boldsymbol{\sigma}^s$ is the Cauchy's stress, $\mathbf{n}^s$ is the unit outward normal on the boundary, Γ^s, of Ω^s, Γ_D^s is the part of the boundary of the domain Ω^s where Dirichlet boundary condition $\bar{\mathbf{d}}^s$ is applied and Γ_N^s is the part of the boundary of the domain Ω^s where Neumann boundary condition $\bar{\mathbf{t}}^s$ is applied. Here, $\Gamma^s = \Gamma_D^s \cup \Gamma_N^s$ and $\Gamma_D^s \cap \Gamma_N^s = \emptyset$, while $\mathbf{d}_0^s$ and $\dot{\mathbf{d}}_0^s$ are the initial displacement and initial velocity of the solid, respectively.

4.1.3. Governing equations at the interface

Two conditions have to be satisfied at the fluid-structure interface, denoted as Γ^{f-s}. The first one is the *no-slip* condition, which enforces that the fluid at interface between fluid and solid moves at the same velocity as the boundary of the solid. The second condition enforces the equilibrium of stresses along the interface. Mathematically, these two conditions

are written as,

$$\mathbf{v}^f = \mathbf{v}^s \qquad \text{on} \quad \Gamma^{f-s} \tag{11}$$

$$\mathbf{t}^f + \mathbf{t}^s = \mathbf{0} \qquad \text{on} \quad \Gamma^{f-s} \tag{12}$$

where, $\mathbf{t}^f$ and $\mathbf{t}^s$ are the tractions exerted by the fluid and solid, respectively, on the interface.

4.2. Weak formulation

Variational equations for FSI can now be written as: Find the fluid velocity $\mathbf{v}^f \in \mathcal{S}_{v^f}$ and pressure $p \in \mathcal{S}_p$, the structural velocities $\mathbf{v}^s \in \mathcal{S}_{v^s}$ and the Lagrange multiplier $\boldsymbol{\lambda} \in \mathcal{S}_\lambda$ such that for all weighting functions $\mathbf{w}^f \in \mathcal{V}_{v^f}$, $q \in \mathcal{V}_p$, $\mathbf{w}^s \in \mathcal{V}_{v^s}$ and $\boldsymbol{\varphi} \in \mathcal{V}_\lambda$

$$B^f(\{\mathbf{w}^f, q\}, \{\mathbf{v}^f, p\}) - F^f(\{\mathbf{w}^f, q\}) + \int_\Gamma \mathbf{w}^f \cdot \boldsymbol{\lambda}\, d\Gamma = 0 \tag{13}$$

$$B^s(\mathbf{w}^s, \mathbf{v}^s) - F^s(\mathbf{w}^s) - \int_\Gamma \mathbf{w}^s \cdot \boldsymbol{\lambda}\, d\Gamma = 0 \tag{14}$$

$$\int_\Gamma \boldsymbol{\varphi} \cdot (\mathbf{v}^f - \mathbf{v}^s)\, d\Gamma = 0 \tag{15}$$

where,

$$B^f(\{\mathbf{w}, q\}, \{\mathbf{v}, p\}) = \int_{\Omega^f} \mathbf{w} \cdot \rho^f \left(\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v} \right) d\Omega^f + \int_{\Omega^f} \boldsymbol{\varepsilon}(\mathbf{w}) : \boldsymbol{\sigma}^f(\mathbf{v}, p)\, d\Omega^f + \int_{\Omega^f} q\, \nabla \cdot \mathbf{v}\, d\Omega^f \tag{16}$$

$$F^f(\{\mathbf{w}, q\}) = \int_{\Omega^f} \mathbf{w} \cdot \mathbf{g}^f\, d\Omega^f + \int_{\Gamma^f_N} \mathbf{w} \cdot \bar{\mathbf{t}}^f\, d\Gamma^f_N \tag{17}$$

$$B^s(\mathbf{w}, \mathbf{v}) = \int_{\Omega^s} \mathbf{w} \cdot \rho^s \frac{\partial \mathbf{v}}{\partial t}|_X\, d\Omega^s + \int_{\Omega^s} \boldsymbol{\varepsilon}(\mathbf{w}) : \boldsymbol{\sigma}^s(\mathbf{v})\, d\Omega^s \tag{18}$$

$$F^s(\mathbf{w}) = \int_{\Omega^s} \mathbf{w} \cdot \mathbf{g}^s\, d\Omega^s + \int_{\Gamma^s_N} \mathbf{w} \cdot \bar{\mathbf{t}}^s\, d\Gamma^s_N \tag{19}$$

The variational formulations given by Eqs. (13) and (14) give the following Euler-Lagrange conditions on the fluid-structure interface Γ^{f-s}:

$$\boldsymbol{\lambda} = -\boldsymbol{\sigma}_f \mathbf{n}_f = \boldsymbol{\sigma}_s \mathbf{n}_s \tag{20}$$

with,

$$\mathbf{n}_f = -\mathbf{n}_s \tag{21}$$

5. Integration in time

In order to complete the discretisation of the weak forms a numerical time integration scheme has to be chosen. In the present work we use the generalised-α method for both the fluid and solid domains. The generalised-α scheme is first introduced by Chung and Hulbert [9] for second-order differential equations arising in structural dynamics. This scheme was later applied to incompressible Navier-Stokes by Jansen et al. [28]. The scheme has been proven to be unconditionally stable and second-order accurate for linear problems, see [9, 12, 28]. Also, this scheme allows the user to control the high-frequency damping using a single parameter, called as spectral radius and denoted usually by ρ_∞.

The total time-interval of interest $[0, T]$ is subdivided into time instants $0 = t_0 < t_1 < t_2 < \ldots t_N = T$ with time step size $\Delta t = t_{n+1} - t_n$.

5.1. Generalised-α method for the solid

With $\mathbf{d}^s_n$, $\mathbf{v}^s_n$ and $\mathbf{a}^s_n$ as the displacement, velocity and acceleration of a solid point at time instant t_n, the basic system of equations for the generalised-α method for the solid is given as,

$$\mathbf{d}^s_{n+1} = \mathbf{d}^s_n + \Delta t \mathbf{v}^s_n + \Delta t^2 \left((\frac{1}{2} - \beta^s)\mathbf{a}^s_n + \beta^s \mathbf{a}^s_{n+1} \right) \tag{22}$$

$$\mathbf{v}^s_{n+1} = \mathbf{v}^s_n + \Delta t \left((1-\gamma)\mathbf{a}^s_n + \gamma \mathbf{a}^s_{n+1} \right) \tag{23}$$

$$\mathbf{d}^s_{n+\alpha^s_f} = (1-\alpha^s_f)\,\mathbf{d}^s_n + \alpha^s_f\,\mathbf{d}^s_{n+1} \tag{24}$$

$$\mathbf{v}^s_{n+\alpha^s_f} = (1-\alpha^s_f)\,\mathbf{v}^s_n + \alpha^s_f\,\mathbf{v}^s_{n+1} \tag{25}$$

$$\mathbf{a}^s_{n+\alpha^s_m} = (1-\alpha^s_m)\,\mathbf{a}^s_n + \alpha^s_m\,\mathbf{a}^s_{n+1} \tag{26}$$

In this work, we choose velocity as the primary variable for the solid domain. So, Eqs. (22) and (23) are rearranged as,

$$\mathbf{d}^s_{n+1} = \mathbf{d}^s_n + \frac{\Delta t(\gamma^s - \beta^s)}{\gamma}\mathbf{v}^s_n + \frac{\Delta t^2(\gamma^s - 2\beta^s)}{2\gamma^s}\,\mathbf{a}^s_n + \frac{\Delta t \beta^s}{\gamma^s}\,\mathbf{v}^s_{n+1} \tag{27}$$

$$\mathbf{a}^s_{n+1} = \frac{\gamma^s - 1}{\gamma^s}\mathbf{a}^s_n + \frac{1}{\gamma^s \Delta t}(\mathbf{v}^s_{n+1} - \mathbf{v}^s_n) \tag{28}$$

Once the velocity $\mathbf{v}^s_{n+1}$ at time t_{n+1} is obtained the displacement $\mathbf{d}^s_{n+1}$ and acceleration $\mathbf{a}^s_{n+1}$ can be computed from Eqs. (27) and (28), respectively.

Chung and Hulbert [9] have shown that this method is unconditionally stable and second-order accurate for the combination of parameters,

$$\alpha^s_m = \frac{2 - \rho^s_\infty}{\rho^s_\infty + 1}, \qquad \alpha^s_f = \frac{1}{1 + \rho^s_\infty} \tag{29}$$

$$\gamma^s = \frac{1}{2} + \alpha^s_m - \alpha^s_f, \qquad \beta^s = \frac{1}{4}(1 + \alpha^s_m - \alpha^s_f)^2 \tag{30}$$

The amount of high-frequency dissipation can controlled by choosing spectral radius $\rho^s_\infty \in [0, 1]$.

5.2. Generalised-α method for the fluid

Similarly, with $\mathbf{v}^f_n$ and $\mathbf{a}^f_n$ as velocity and acceleration of a fluid at time instant t_n, the basic system of equations for the generalised-α method for the fluid is given as,

$$\mathbf{v}^f_{n+1} = \mathbf{v}^f_n + \Delta t \left((1 - \gamma^f)\mathbf{a}^f_n + \gamma^f \mathbf{a}^f_{n+1}\right) \tag{31}$$

$$\mathbf{v}^f_{n+\alpha^f_f} = (1 - \alpha^f_f)\,\mathbf{v}^f_n + \alpha^f_f\,\mathbf{v}^f_{n+1} \tag{32}$$

$$\mathbf{a}^f_{n+\alpha^f_m} = (1 - \alpha^f_m)\,\mathbf{v}^f_n + \alpha^f_m\,\mathbf{a}^f_{n+1} \tag{33}$$

By choosing fluid velocity $\mathbf{v}^f$ as the primary variable Eq. (31) can be rearranged as,

$$\mathbf{a}^f_{n+1} = \frac{1}{\gamma^f \Delta t}\,(\mathbf{v}^f_{n+1} - \mathbf{v}^f_n) + \frac{\gamma^f - 1}{\gamma^f}\,\mathbf{a}^f_n \tag{34}$$

Once the velocity $\mathbf{v}^f_{n+1}$ at time t_{n+1} is obtained the acceleration $\mathbf{a}^f_{n+1}$ can be computed from Eq. (34). In order to achieve the second-order accuracy the combination of parameters is,

$$\alpha^f_m = \frac{1}{2}\,\frac{3 - \rho^f_\infty}{1 + \rho^f_\infty}, \quad \alpha^f_f = \frac{1}{1 + \rho^f_\infty}, \quad \gamma^f = \frac{1}{2} + \alpha^f_m - \alpha^f_f \tag{35}$$

Similar to the time discretisation for solids, the numerical damping can be controlled by choosing the spectral radius $\rho_\infty^f \in [0, 1]$.

5.3. Interpolation in time at the interface

During the overall time-stepping algorithm the Lagrange multipliers $\boldsymbol{\lambda}$ have to be evaluated at the time instants $t_{n+\alpha_f^s}$ and $t_{n+\alpha_f^f}$. Using the same parameters as in generalised-α methods for the fluid and the solid, we can write,

$$\boldsymbol{\lambda}_{n+\alpha_f^f} = (1 - \alpha_f^f)\, \boldsymbol{\lambda}_n + \alpha_f^f\, \boldsymbol{\lambda}_{n+1} \tag{36}$$

$$\boldsymbol{\lambda}_{n+\alpha_f^s} = (1 - \alpha_f^s)\, \boldsymbol{\lambda}_n + \alpha_f^s\, \boldsymbol{\lambda}_{n+1} \tag{37}$$

In order to avoid mapping of data from $t_{n+\alpha_f^s}$ to $t_{n+\alpha_f^f}$ and vice-versa we choose same value of spectral radius for both the fluid and the solid domains. This renders,

$$\alpha_f^s = \alpha_f^f \tag{38}$$

$$t_{n+\alpha_f^s} = t_{n+\alpha_f^f} \tag{39}$$

$$\boldsymbol{\lambda}_{n+\alpha_f^s} = \boldsymbol{\lambda}_{n+\alpha_f^f} \tag{40}$$

With this choice, all the computations can be performed at one time instant $t_{n+\alpha_f^s} = t_{n+\alpha_f^f}$. For situations when time instants $t_{n+\alpha_f^s}$ and $t_{n+\alpha_f^f}$ are different, the forces at the interface have to be interpolated consistently. Failing to do so will result in instabilities and loss of accuracy of the overall FSI scheme. For detailed discussion on this topic we refer the reader to Joosten et al. [30, 31].

5.4. Discretisation

By taking the approximations for the solution variables and their corresponding test functions as,

$$\mathbf{v}_f = \mathbf{N}_{v_f} \bar{\mathbf{v}}_f, \quad \mathbf{w}_f = \mathbf{N}_{w_f} \bar{\mathbf{w}}_f \tag{41}$$

$$p = \mathbf{N}_p \bar{\mathbf{p}}_f, \quad q = \mathbf{N}_q \bar{\mathbf{q}}_f \tag{42}$$

$$\mathbf{v}_s = \mathbf{N}_{v_s} \bar{\mathbf{v}}_s, \quad \mathbf{w}_s = \mathbf{N}_{w_s} \bar{\mathbf{w}}_s \tag{43}$$

$$\boldsymbol{\lambda} = \mathbf{N}_\lambda \bar{\boldsymbol{\lambda}}, \quad \boldsymbol{\phi} = \mathbf{N}_\phi \bar{\boldsymbol{\phi}} \tag{44}$$

and using the Newton-Raphson scheme to solve the non-linear system of equations resulting from the weak-formulation, we obtain a matrix system of the form,

$$\begin{bmatrix} \mathbf{K}_{v^f v^f} & \mathbf{K}_{v^f p} & \mathbf{K}_{v^f \lambda} & \mathbf{0} \\ \mathbf{K}_{p v^f} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{K}_{\lambda v^f} & \mathbf{0} & \mathbf{0} & \mathbf{K}_{\lambda v^s} \\ \mathbf{0} & \mathbf{0} & \mathbf{K}_{v^s \lambda} & \mathbf{K}_{v^s v^s} \end{bmatrix} \begin{Bmatrix} d\bar{\mathbf{v}}^{\mathbf{f}} \\ d\bar{\mathbf{p}} \\ d\bar{\boldsymbol{\lambda}} \\ d\bar{\mathbf{v}}^{\mathbf{s}} \end{Bmatrix} = - \begin{Bmatrix} \mathbf{R}_{\mathbf{v}^{\mathbf{f}}} \\ \mathbf{R}_p \\ \mathbf{R}_{\boldsymbol{\lambda}} \\ \mathbf{R}_{\mathbf{v}^{\mathbf{s}}} \end{Bmatrix} \tag{45}$$

where, $\mathbf{N}_{v_f}$, $\mathbf{N}_p$, $\mathbf{N}_\lambda$ and $\mathbf{N}_{v_s}$ are the basis functions for the fluid velocity, fluid pressure, Lagrange multipliers and solid velocity, respectively. Here, $\mathbf{N}_{w_f}$, $\mathbf{N}_q$, $\mathbf{N}_\phi$ and $\mathbf{N}_{w_s}$ are the corresponding test functions for the fluid velocity, fluid pressure, Lagrange multipliers and solid velocity, respectively. In this work $\mathbf{N}_{v_f}$, $\mathbf{N}_{w_f}$, $\mathbf{N}_p$ and $\mathbf{N}_q$ are B-Splines; $\mathbf{N}_\lambda$ and $\mathbf{N}_\phi$ are Dirac delta functions; and $\mathbf{N}_{v_s}$ and $\mathbf{N}_{w_s}$ are linear Lagrange polynomials.

6. Geometrically exact formulation for the beam

In this work we model the solid as an infinitely thin line formulated with geometrically-exact beam described in Zienkiewicz and Taylor [67]. This formulation takes into account large displacements and finite rotations but assumes that the resulting strains are small. Assuming that the beam is aligned with the X-axis and deforms in X-Y plane, the deformed position from Fig. 5 can be written as,

$$\begin{aligned} x &= X + u + Y\sin\beta \\ y &= v + Y\cos\beta \end{aligned} \tag{46}$$

where, u and v are the X- and Y-displacements and β is the angle between the normal to the beam cross-section and X-axis.

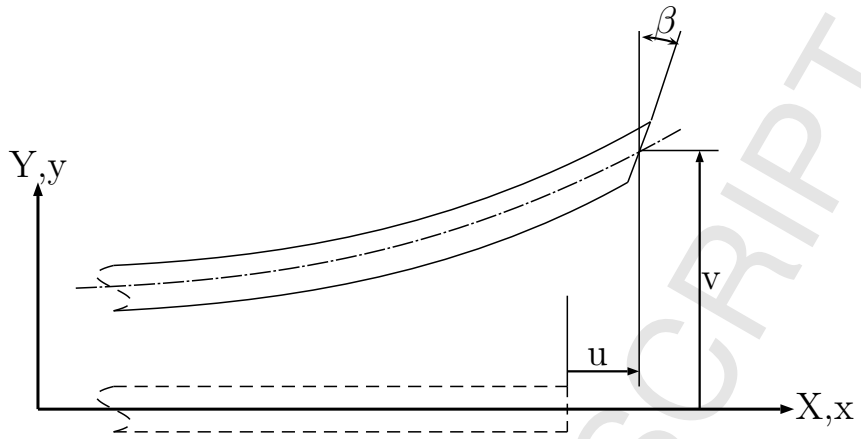


Figure 5: Beam formulation - original and deformed configurations.

For the displacements in Eq. 46, the deformation gradient becomes,

$$\mathbf{F} = \begin{bmatrix} 1 + \frac{\partial u}{\partial X} + Y\cos\beta\frac{\partial \beta}{\partial X} & \sin\beta & 0 \\ \frac{\partial w}{\partial X} - Y\sin\beta\frac{\partial \beta}{\partial X} & \cos\beta & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{47}$$

Using the deformation gradient in Eq. 47, the two non-zero components of the Green-Lagrange strain tensor ($\mathbf{E} = \frac{1}{2}(\mathbf{F}^T\mathbf{F} - \mathbf{I})$), ignoring the quadratic terms in Y, are,

$$E_{XX} = \frac{\partial u}{\partial X} + \frac{1}{2}\left(\left(\frac{\partial u}{\partial X}\right)^2 + \left(\frac{\partial w}{\partial X}\right)^2\right) + Y\frac{\partial \beta}{\partial X}\left(\cos\beta\left(1 + \frac{\partial u}{\partial X}\right) - \sin\beta\frac{\partial w}{\partial X}\right) \tag{48}$$

$$E_{XY} = \frac{1}{2}\left(\sin\beta\left(1 + \frac{\partial u}{\partial X}\right) + \cos\beta\frac{\partial w}{\partial X}\right) \tag{49}$$

Finite element formulation for the beam can be developed using variational statements based on the Green-Lagrange tensor and the second Piola-Kirchhoff stress. For the detailed description of the formulation the reader is suggested to refer to Chapter 17 in [67].

7. Numerical examples

It is an established fact that the mixed Galerkin formulation for incompressible Navier-Stokes with equal-order interpolation for velocity and pressure violates LBB condition and

needs to be stabilised in order to obtain numerical solutions that are free from spurious oscillations. However, our experience shows that the use of pressure stabilisation along with Lagrange multipliers to enforce the kinematic constraint along the immersed boundary affects the conservation of mass and hence leads to erroneous results. Hence, we first address the issue of conservation using an example of steady flow over a fixed cylinder placed inside a narrow channel. Then, we validate our numerical scheme by studying the flow over a fixed circular cylinder at Reynolds numbers 20, 40, 100 and 200 and compare the drag coefficient (C_D), lift coefficient C_L and Strouhal number St with the reference values. Finally, several fluid-structure interaction benchmark tests are presented.

In this work, the direct solver PARDISO [2] is used to solve the global matrix system in Eq. 45. For all the problems involving unsteady flow, a spectral radius value of $\rho_\infty = 0.8$ (same value for both fluid and solid for FSI examples) is used .

7.1. Conservation test

In this example we demonstrate the conservation properties of the proposed numerical scheme by studying flow over a cylinder in a narrow channel. This example was studied in [8] to investigate and improve the conservation properties of least-squares finite element formulation. The geometry of the problem is as shown in Fig. 6. The density and viscosity of the fluid are $\rho^f = 1.0$ and $\mu^f = 0.1$, respectively. The boundary conditions are: $(v_x, v_y) = (1.0, 0.0)$ on the entire outer boundary and $(v_x, v_y) = (0, 0)$ on the surface of the cylinder. The boundary of the cylinder is discretised with 200 equally spaced points. Simulations are performed on an uniform mesh of 300×100 elements, shown in Fig. 7, with linear (Q_1), quadratic (Q_2) and cubic (Q_3) B-Splines. Two sets of simulations are performed, one without any pressure stabilisation and the other with PSPG stabilisation [57], in order to assess the conservation.

The contour plots of pressure presented in Fig. 8 show that equal-order B-Splines for velocity and pressure, without any pressure stabilisation, results in spurious oscillations in the pressure field for linear and cubic B-Splines. In contrast, for quadratic B-Splines the pressure field is smooth. Use of of pressure stabilisation alleviates this problem and results

in smooth pressure field, as expected.

However, the contour plots of X-velocity shown in Fig. 9 indicate that pressure stabilisation significantly affects the amount of flow though the openings BC and DE (see Fig. 6) and reduces the peak velocity of flow across the openings. The reduction in flow through the openings is balanced by the significant spurious flow inside the cylinder, and this can be confirmed from the profiles of X-velocity along the vertical line AC (in Fig. 6) illustrated in Fig. 10. In order to assess the accuracy, volumetric flow rates across the opening BC in Fig. 6 are computed and compared with the theoretical value. Computed flow rates, tabulated in Table 1, clearly indicate that pressure stabilisation used along with Lagrange multipliers to enforce kinematic constraints significantly affects mass conservation. The same behaviour has also been observed in [33], wherein the stabilisation parameter is scaled-down in the vicinity of the immersed bodies by using an ad hoc parameter. This scaling parameter and also the extent of the fluid domain in the vicinity of immersed bodies in which the stabilisation parameter has to be lowered needs to be chosen carefully so that the results obtained are accurate. Because of these difficulties associated with the scaling of the stabilisation parameter and also because the pressure field obtained with equal-order quadratic B-Splines is smooth enough (Fig. 8(c)) we decide to use equal-order B-Splines without any pressure stabilisation for all the examples presented in this paper. Even though this combination of velocity-pressure is *inf-sup* unstable the numerical results obtained match well with the reference values in all the examples that have been attempted..

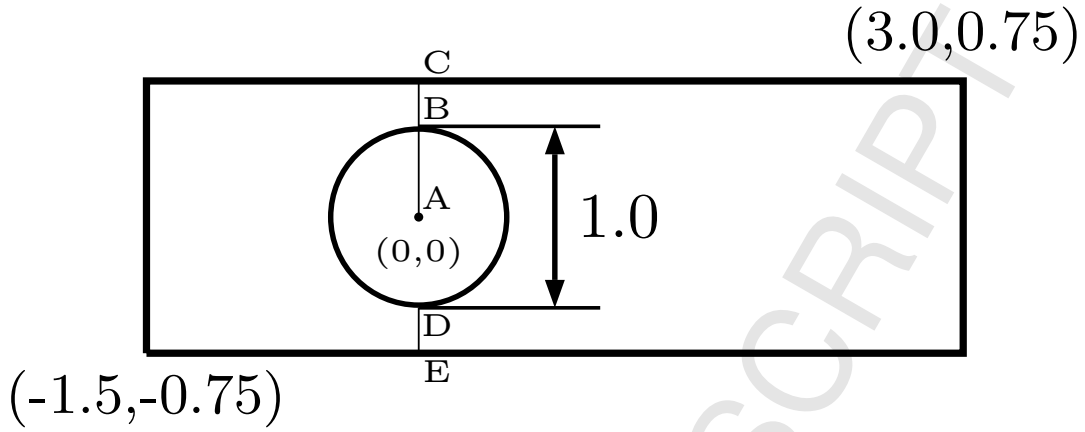


Figure 6: Conservation test: geometry of the problem.

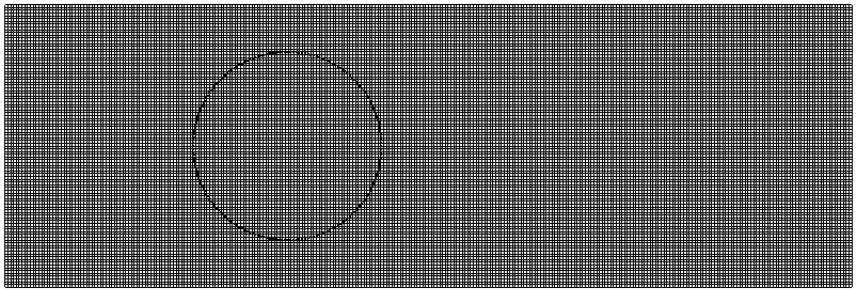
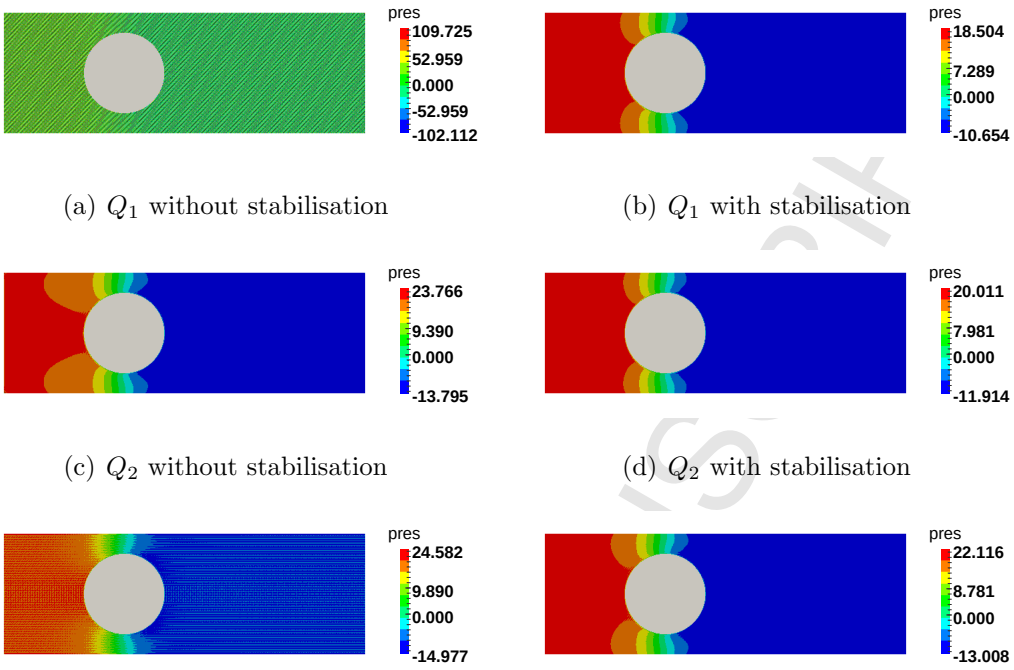


Figure 7: Conservation test: mesh used for the simulations.



(a) Q_1 without stabilisation

(b) Q_1 with stabilisation

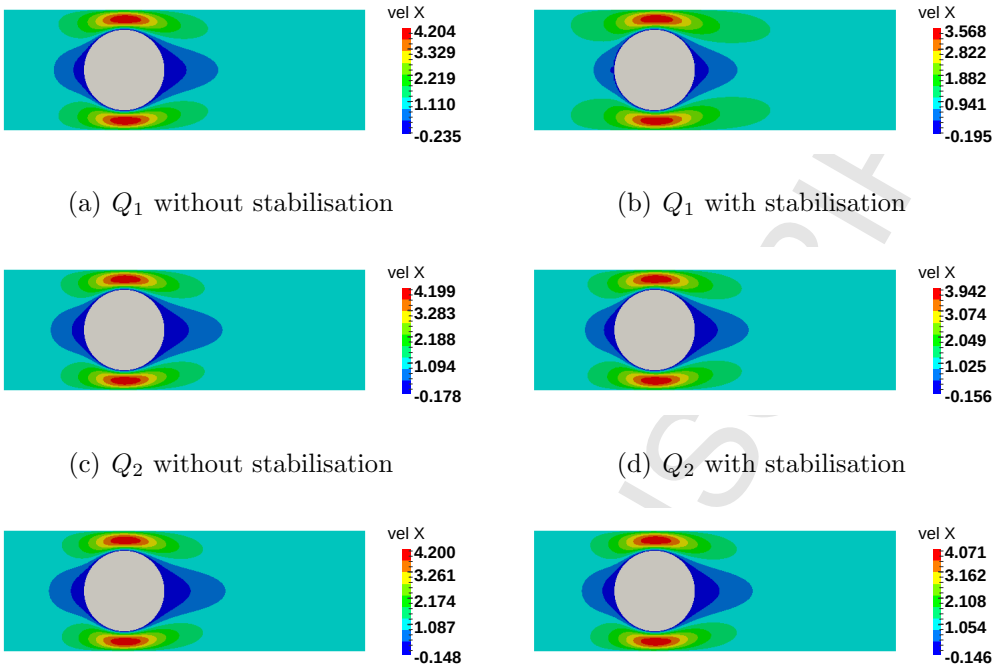
(c) Q_2 without stabilisation

(d) Q_2 with stabilisation

(e) Q_3 without stabilisation

(f) Q_3 with stabilisation

Figure 8: Conservation test: contour plots of pressure without and with PSPG stabilisation for linear (Q_1), quadratic (Q_2) and cubic (Q_3) B-Splines.



(a) Q_1 without stabilisation (b) Q_1 with stabilisation

(c) Q_2 without stabilisation (d) Q_2 with stabilisation

(e) Q_3 without stabilisation (f) Q_3 with stabilisation

Figure 9: Conservation test: contour plots of X-velocity without and with PSPG stabilisation for linear (Q_1), quadratic (Q_2) and cubic (Q_3) B-Splines.

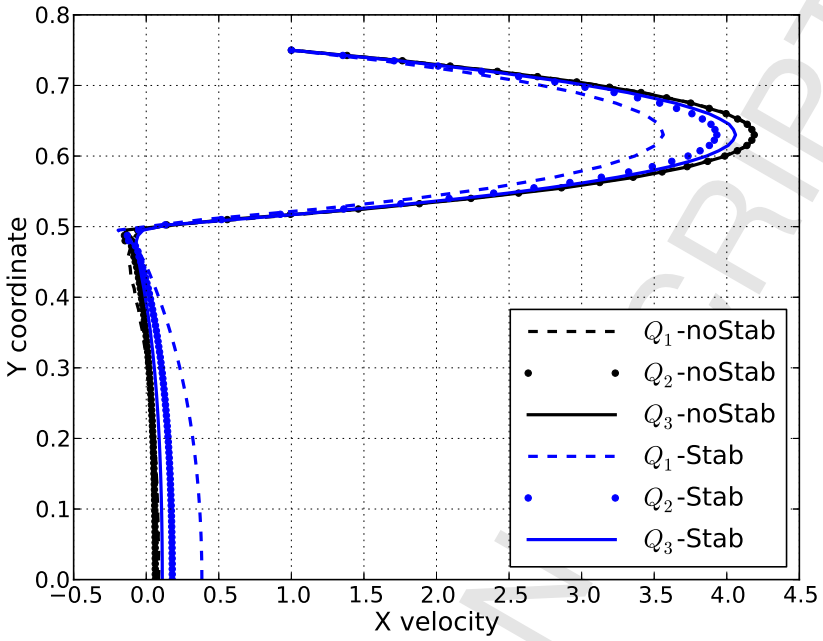


Figure 10: Conservation test: X-velocity profiles along the vertical line AC in Fig. 6, without (Q_a-noStab) and with PSPG stabilisation (Q_a-Stab) for linear (Q_1), quadratic (Q_2) and cubic (Q_3) B-Splines.

B-Spline degree	with stabilisation		without stabilisation	
	flow rate	% error	flow rate	% error
Q_1	0.6326	15.65	0.745	0.67
Q_2	0.6983	6.89	0.746	0.53
Q_3	0.7222	3.71	0.746	0.53

Table 1: Conservation test: comparison of flow rates across BC. Due to symmetric nature of the problem the theoretical value of flow rate across BC is 0.75.

7.2. Flow past a fixed cylinder

The geometry and boundary conditions of the problem are as shown in Fig. 11(a). The fluid domain is refined with hierarchical B-Splines around the cylinder as shown in Figs. 11(b) and 11(c). First, this problem is studied with steady Navier-Stokes for $Re = 20$ and $Re = 40$ with different levels of hierarchical refinements in order to check the convergence with respect to spatial discretisation. The computed values of C_D, presented in Table. 2 along with the reference values, indicate that the accuracy improves with the refinement. The table also contains the information about the number of points used to represent the boundary of the cylinder (NBP) and total degree of freedom (DOF) in each model. Contour plots of pressure, vorticity and streamlines obtained with Level-4 mesh for $Re = 20$ and $Re = 40$ are shown in Fig. 12. Vector plots of Lagrange multipliers obtained with Level-1 to Level-4 meshes are shown in Fig. 13. The plots indicate that the the multiplier field becomes smooth with refinement. We believe that this effect is direct manifestation of the *inf-sup* stability of velocity-multiplier field.

Later, we studied the flow for $Re = 100$ and $Re = 200$ to check the accuracy of results for unsteady flows. These studies are performed on Level-3 and Level-4 meshes, with $\Delta t = 0.1$ and $\Delta t = 0.05$ for each mesh, and the results are tabulated in Table. 3. Figs. 14 and 15 show the evolution of C_D and C_L, for $Re = 100$ and $Re = 200$, respectively. Evolution of C_L for $Re = 100$ and $Re = 200$ with Level-4 mesh with $\Delta t = 0.1$ and $\Delta t = 0.05$ is shown in Fig. 16. Snapshots of pressure and streamlines at two different time instants for $Re = 100$ with Level-3 mesh with $\Delta t = 0.1$ are presented in Fig.17. These results obtained with the proposed numerical scheme are well within range of those observed in the literature. We believe that the accuracy can be improved further by extending the zone of refinement around the cylinder and also using smaller time steps. However, we would like to emphasize that such extensive studies are not the main focus of this paper.

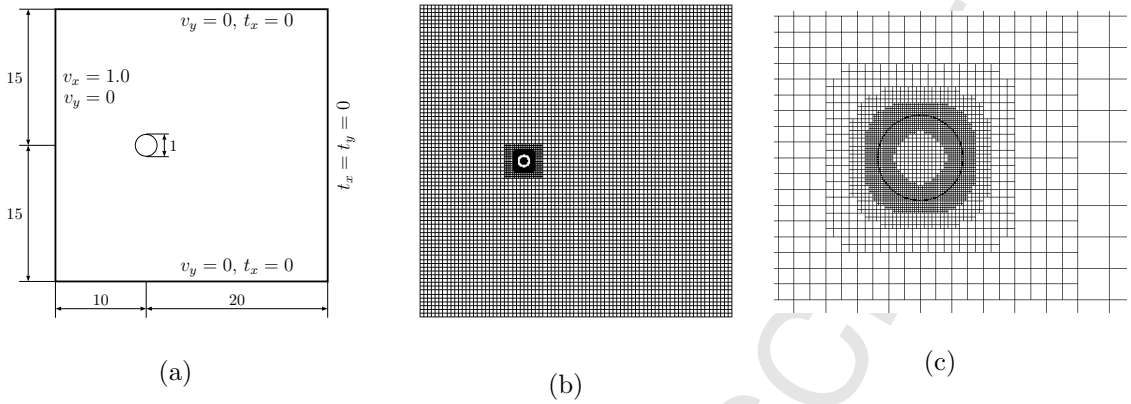


Figure 11: Flow past a fixed circular cylinder: a.) geometry and boundary conditions, b.) B-Spline mesh and c.) hierarchical refinement near the cylinder.

	NBP	DOF	C_D for $Re = 20$	C_D for $Re = 40$
Calhoun [7]	-	-	2.19	1.62
Russell and Wang [52]	-	-	2.13	1.60
Linnick and Fasel [39]	-	-	2.06	1.61
Present (Level-0)	10	20687	2.09	1.40
Present (Level-1)	20	21403	2.21	1.68
Present (Level-2)	40	22559	2.18	1.63
Present (Level-3)	80	25051	2.15	1.61
Present (Level-4)	160	29855	2.15	1.60

Table 2: Flow past a fixed circular cylinder: drag coefficient (C_D) for $Re = 20$ and $Re = 40$ with different levels of hierarchical refinement.

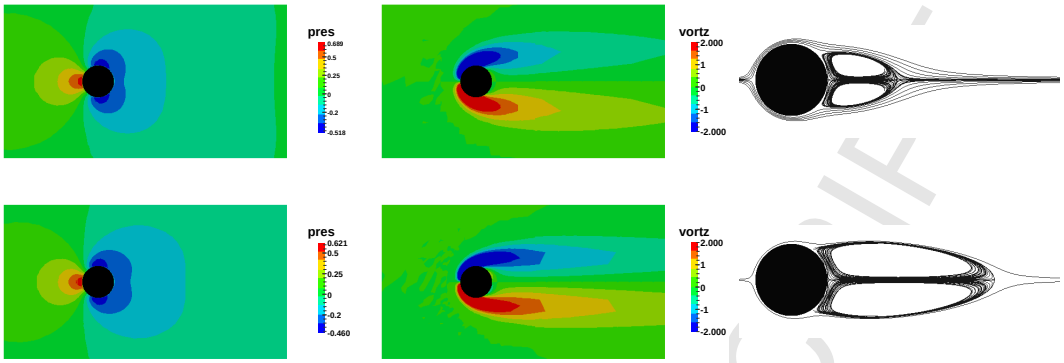
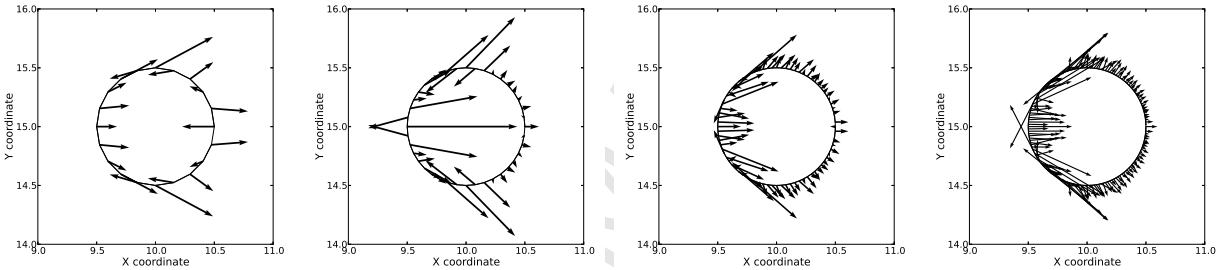
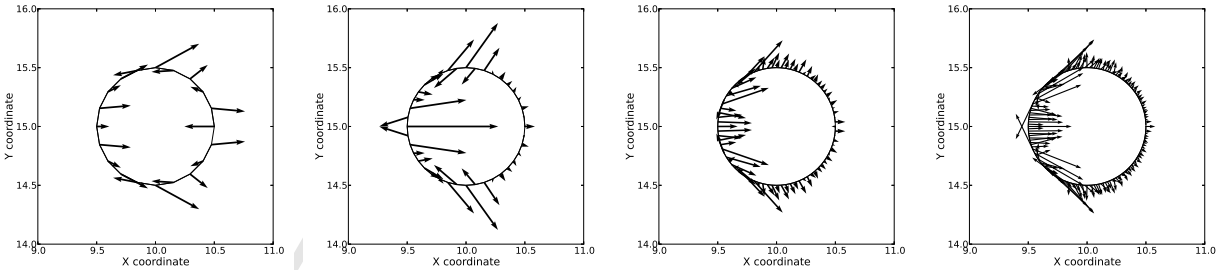


Figure 12: Flow past a fixed circular cylinder: contour plots. top: $Re = 20$. bottom: $Re = 40$. left: pressure, center: vorticity, right: streamlines.



(a) Level-1 (b) Level-2 (c) Level-3 (d) Level-4

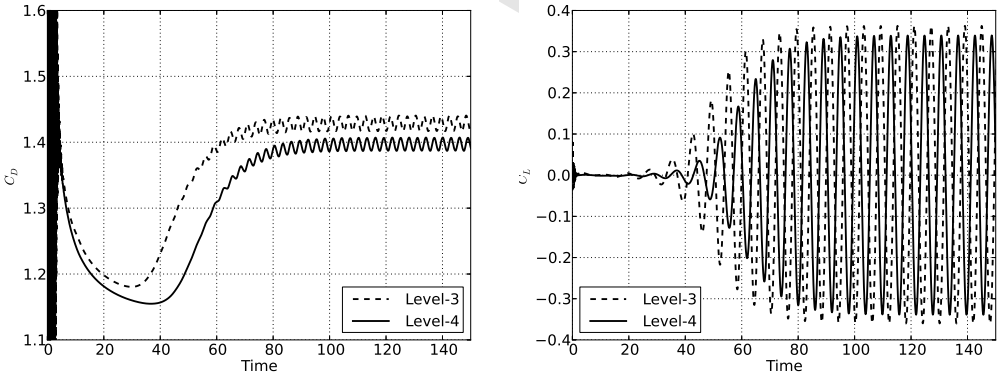


(e) Level-1 (f) Level-2 (g) Level-3 (h) Level-4

Figure 13: Flow past a fixed circular cylinder: vector plots of Lagrange multipliers for $Re = 20$ (top row) and $Re = 40$ (bottom row) with Level-1, Level-2, Level-3 and Level-4 meshes. Scaling factors of 2, 4, 8 and 16 are used for plotting the multipliers obtained with Level-1, Level-2, Level-3 and Level-4 meshes, respectively.

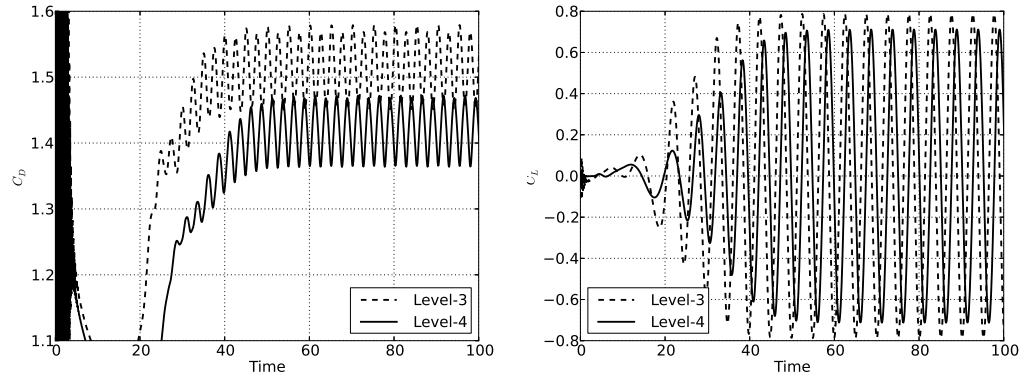
Data	$Re = 100$			$Re = 200$		
	C_D	C_L	St	C_D	C_L	St
Braza et al. [6]	1.36	±0.250	-	1.40	±0.750	-
Liu et al. [40]	1.35	±0.339	0.165	1.31	±0.690	0.192
Calhoun [7]	1.33	±0.298	0.175	1.17	±0.668	0.202
Russell and Wang [52]	1.38	±0.300	0.169	1.29	±0.500	0.195
Le et al. [34]	1.37	±0.323	0.160	1.34	±0.430	0.187
Kamensky et al. [33]	1.39	±0.341	0.170	1.38	±0.706	0.200
Present (Level-3 $\Delta t = 0.1$)	1.42	±0.360	0.173	1.51	±0.789	0.203
Present (Level-3 $\Delta t = 0.05$)	1.42	±0.362	0.171	1.51	±0.788	0.196
Present (Level-4 $\Delta t = 0.1$)	1.39	±0.339	0.165	1.42	±0.711	0.200
Present (Level-4 $\Delta t = 0.05$)	1.39	±0.339	0.166	1.42	±0.711	0.194

Table 3: Flow past a fixed circular cylinder: C_D, C_L and St for $Re = 100$ and $Re = 200$.



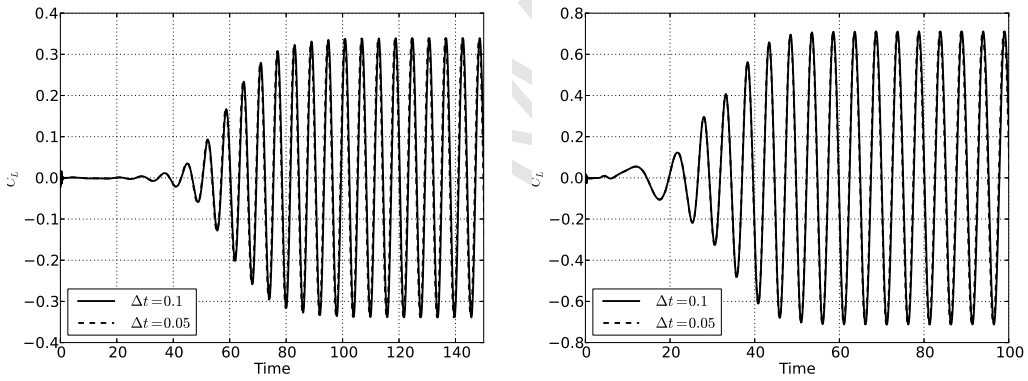
(a) Drag coefficient (C_D) (b) Lift coefficient (C_L)

Figure 14: Flow past a fixed circular cylinder: evolution of C_D and C_L for $Re = 100$ with Level-3 and Level-4 meshes with $\Delta t = 0.1$.



(a) Drag coefficient (C_D) (b) Drag coefficient (C_L)

Figure 15: Flow past a fixed circular cylinder: evolution of C_D and C_L for $Re = 200$ with Level-3 and Level-4 meshes with $\Delta t = 0.1$.



(a) $Re = 100$ (b) $Re = 200$

Figure 16: Flow past a fixed circular cylinder: evolution of C_L for $Re = 100$ and $Re = 200$ with Level-4 mesh for different time steps.

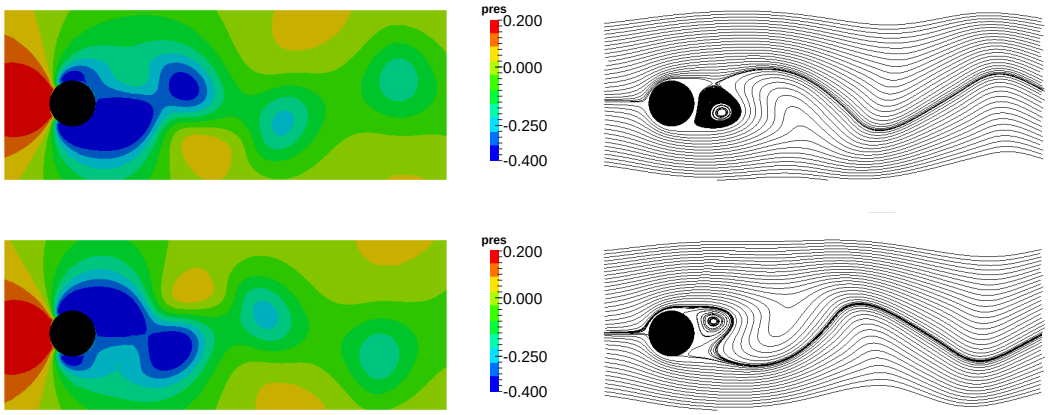


Figure 17: Flow past a fixed circular cylinder: contour plots of pressure (left) and streamlines (right) at two different time instants for $Re = 100$ with Level-3 mesh with $\Delta t = 0.1$.

7.3. Two flapping leaves

This problem, introduced by [17] and studied by [24, 33, 60], is an idealised two-dimensional model of mitral valve. The geometry and boundary conditions of the problem are as shown in Fig. 18. The problem consists of two leaf valves of equal length fixed to the boundaries of a 2D channel. These valves are subjected to sinusoidally varying horizontal velocity profile on the inlet, given by,

$$v_{in} = 5\, y(1.61 - y)\,(1.1 + \sin(2\pi t)) \tag{50}$$

No-slip boundary conditions are applied on the top and bottom sides of the channel and the outlet is chosen to be traction-free. The material properties of the fluid and the solid are same as those proposed in [17]. Density and viscosity of the fluid are $\rho^f = 100$ and $\mu^f = 10$, respectively. The thickness of the valve is $h = 0.0212$. Young's modulus of the valve is $E = 5 \times 10^7$ and Poisson's ratio is $\nu = 0.4$.

In this work, this problem is studied with three different levels of hierarchical B-Spline refinement to assess the convergence with respect to spatial discretisation. The portion of the fluid grid where the leaves undergo deformation is refined with hierarchical B-Splines as shown in Fig. 19. Each leaf in Level-k mesh is modelled with 10×2^k beam elements. In order to assess temporal convergence of the presented numerical scheme, each of the

discretisations is studied with two different time-steps. Figs. 20 and 21 show the X- and Y-displacement of free end of a leaf for Level-0 and Level-2 meshes obtained with $\Delta t = 0.01$ and $\Delta t = 0.005$. These graphs indicate that there is negligible difference between the numerical results obtained with the two different time steps. Fig. 22 shows the evolution of X- and Y-displacement of the leaf tip for all the four meshes with $\Delta t = 0.005$. Clearly, the solution converges as the mesh is refined. Contour plots of X-velocity, pressure and vorticity at time instant $t = 0.5$ are presented, respectively, in Figs. 23, 24, and 25. These plots also show that the solution improves as the mesh is refined. Moreover, the important thing to notice from those plots is that the pressure obtained is sufficiently smooth, except near the immersed boundaries, and the oscillations in pressure disappear with mesh refinement. Eventhough the velocity-pressure combination is *inf-sup* unstable the overall quality of pressure obtained in this work is superior to that reported in [33].

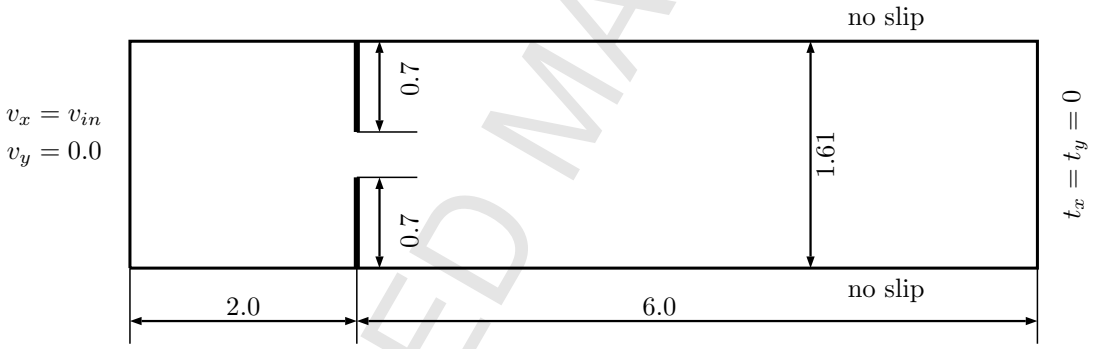
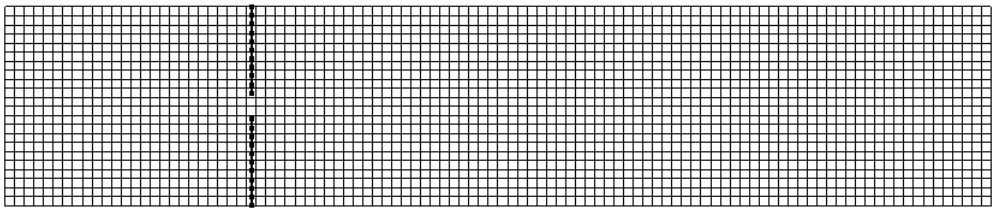
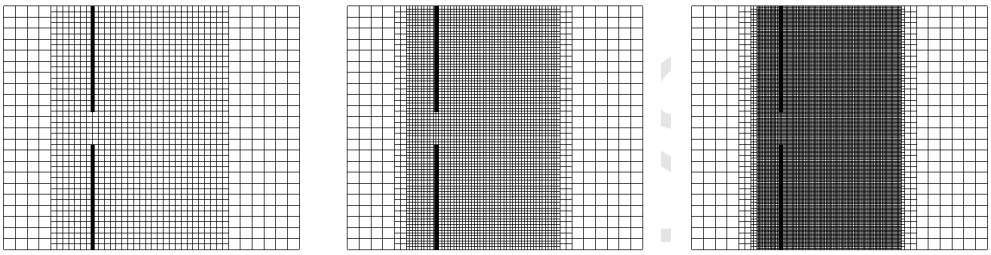


Figure 18: Two flapping leaves: geometry and boundary conditions.



(a) Level-0 mesh. 10 beam elements for each leaf. (7598).



(b) Level-1 mesh. (10626) (c) Level-2 mesh. (21014) (d) Level-3 mesh. (59988)

Figure 19: Two flapping leaves: hierarchical refinements used for the analysis. The numbers in the brackets indicate total DOFs in the model.

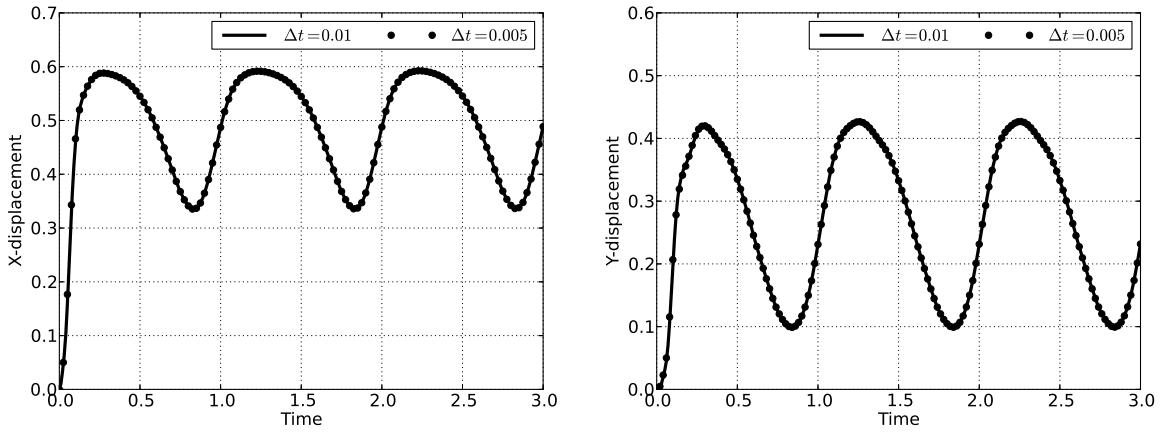


Figure 20: Two flapping leaves: tip displacements for Level-0 mesh with different time steps.

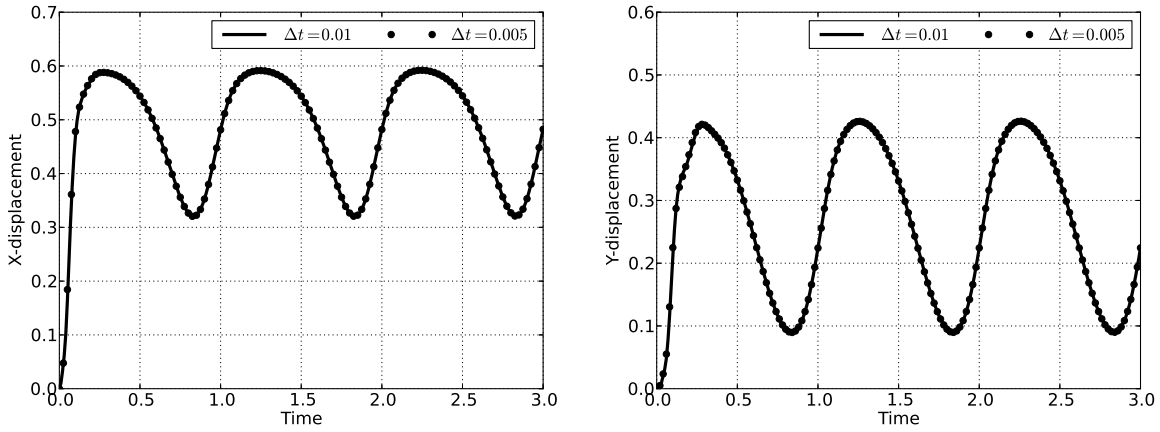


Figure 21: Two flapping leaves: tip displacements for Level-2 mesh with different time steps.

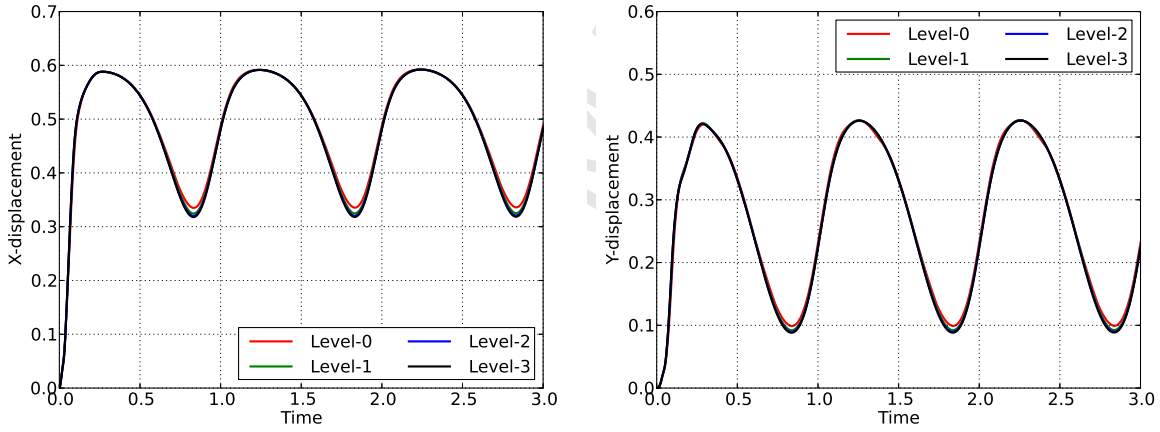
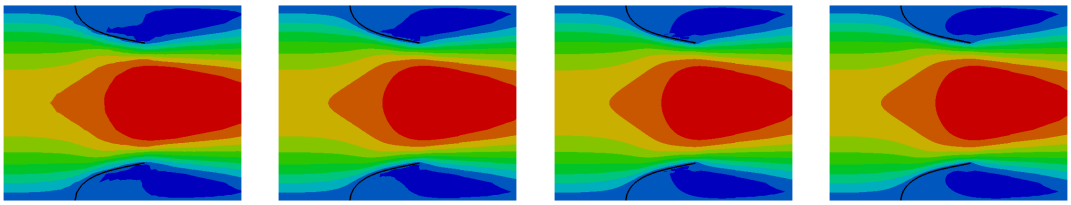
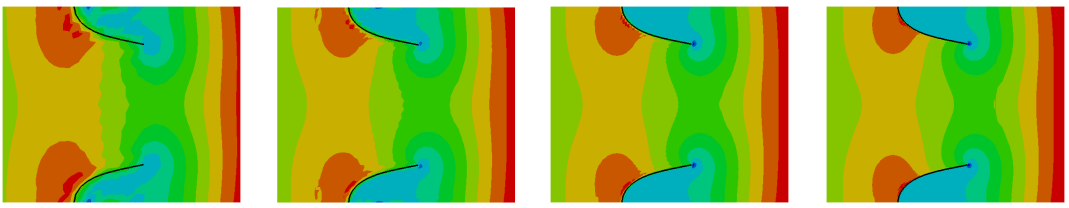


Figure 22: Two flapping leaves: tip displacements with different levels of hierarchical refinement with $\Delta t = 0.005$.



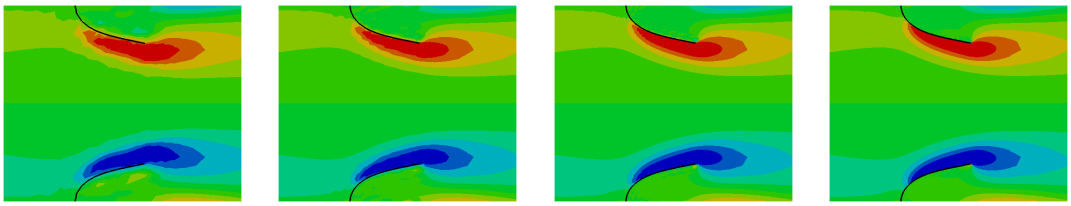
(a) Level-0 (b) Level-1 (c) Level-2 (d) Level-3

Figure 23: Two flapping leaves: X-velocity contour plots at $t = 0.5$ with $\Delta t = 0.005$.



(a) Level-0 (b) Level-1 (c) Level-2 (d) Level-3

Figure 24: Two flapping leaves: pressure contour plots at $t = 0.5$ with $\Delta t = 0.005$.



(a) Level-0 (b) Level-1 (c) Level-2 (d) Level-3

Figure 25: Two flapping leaves: vorticity contour plots at $t = 0.5$ with $\Delta t = 0.005$.

7.4. Vortex-induced vibrations of a flexible beam

This problem was introduced by [59] and is used as a benchmark to demonstrate the accuracy of a numerical scheme for fluid-flexible body interaction. This problem has been studied by [11, 13, 14, 27, 33] using various numerical schemes. The geometry and boundary conditions of the problem are as shown in Fig. 26. All the dimensions shown in the geometry description are in centimetres. Fluid density and viscosity are $\rho^f = 1.18 \times 10^{-3}$ g/cm^3 and

$\mu^f = 1.82 \times 10^{-4}$ g/cm s, respectively. The density of the beam material is $\rho^s = 0.1$ g/cm^3, its Young's modulus is $E = 2.5 \times 10^6$ g/cm s^2 and Poisson's ratio is $\nu = 0.35$. The inflow velocity in X-direction is $v_{in} = 51.3$ cm/s. For these properties and based on side of the square ($D = 1.0$ cm) Reynolds number is $Re = \rho^f D v_{in}/\mu^f \approx 333$. The beam, which is attached behind a fixed square body, starts to oscillate due to vortices shed by the corners of the square body.

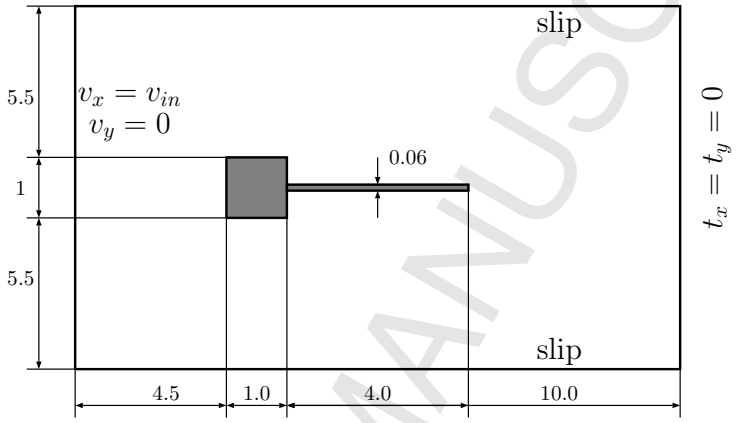


Figure 26: Vortex-induced vibrations of a flexible beam: geometry and boundary conditions.

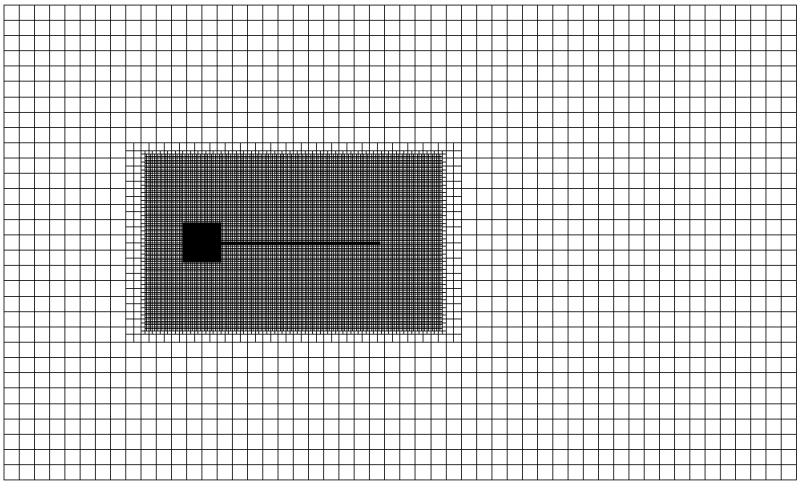


Figure 27: Vortex-induced vibrations of a flexible beam: hierarchical B-Spline mesh with three levels of refinement. DOFs for Level-2 = 15966+160+123 = 16249. DOFs for Level-3 = 47514+320+243 = 48077.

In this paper, we present numerical results obtained with two and three levels of hierarchical refinement. The hierarchically refined B-Spline mesh is shown in Fig. 27. For Level-2 mesh the square is represented by 40 points along its boundary and the beam is modelled with 40 linear geometrically exact beam elements. These quantities are doubled for the Level-3 mesh. This discretisation corresponds to approximately one point for each B-Spline element. Time increments of $\Delta t = 0.005$ and $\Delta t = 0.004$, respectively, are used for the Level-2 and Level-3 meshes. Evolution of Y-displacement of the beam-tip with respect to time is presented in Fig. 28 and the maximum tip displacement and frequency of oscillations are compared with the values from literature in Table. 4. The results obtained with the present scheme match well the values from literature. Fig. 29 shows the contour plots of magnitude of velocity, pressure and vorticity at two time instants for the Level-3 mesh.

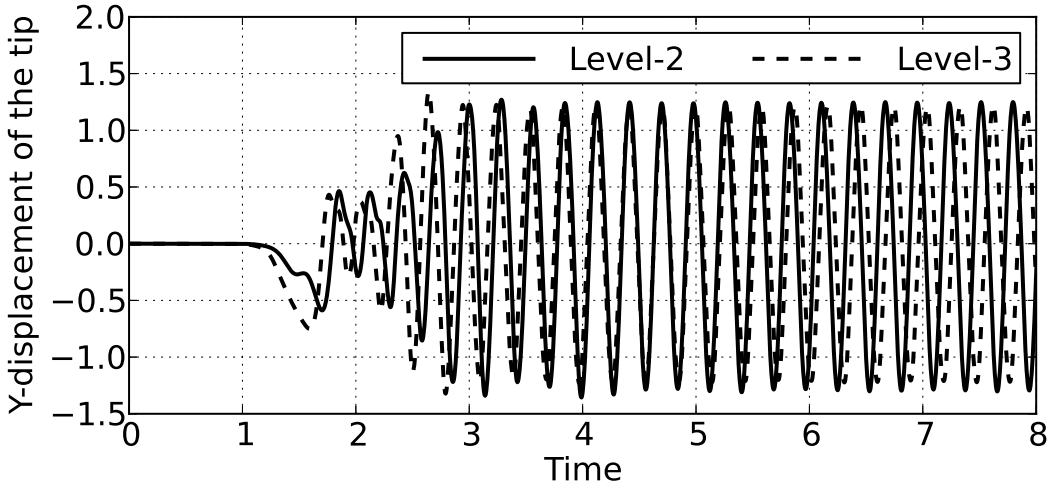


Figure 28: Vortex-induced vibrations of a flexible beam: evolution of vertical displacement of beam tip with respect to time.

Author	Max tip-displacement	Frequency (Hz)
Wall [59]	1.12 - 1.32	2.78 - 3.22
Dettmer and Perić [11]	1.1 - 1.4	2.96 - 3.31
Present (Level-2 mesh)	1.27	3.41
Present (Level-3 mesh)	1.26	3.22

Table 4: Vortex-induced vibrations of a flexible beam: maximum vertical displacement of the beam-tip and frequency of oscillations obtained with Level-2 and Level-3 meshes.

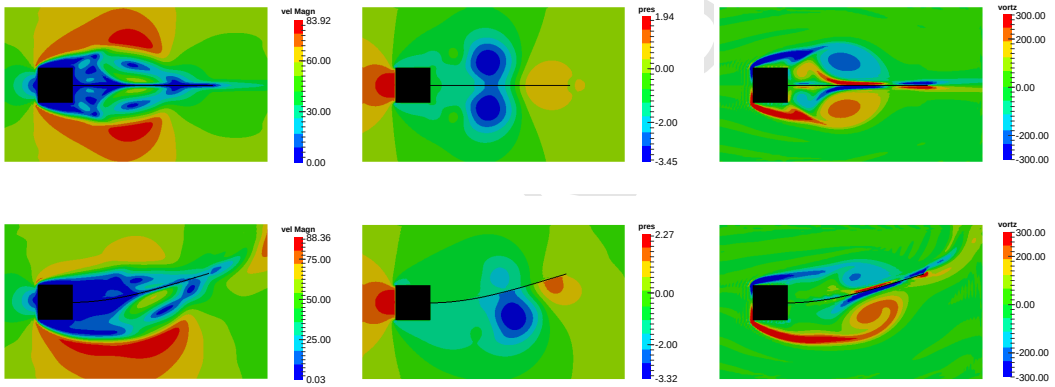


Figure 29: Vortex-induced vibrations of a flexible beam: contour plots of velocity-magnitude, pressure and vorticity at two different time instants for Level-3 mesh.

7.5. Single leaf in cross-flow

This problem is similar to that of heart-valve example but involves only one flexible leaf. This example is chosen to demonstrate the robustness of the proposed method to simulate the motion of flexible leaf undergoing extreme deformations. This problem was introduced by Baaijens [3] and was also studied by Yu [62]. While [3, 62] modelled the leaf using solid elements we model it using 1D beam elements. Unlike in Baaijens [3] we model the dynamics of the beam. The geometry and boundary conditions are as depicted in Fig. 30. The leaf is attached to the bottom wall of the channel where no-slip boundary condition is applied. The density and viscosity of the fluid and solid are: $\rho^f = 100.0$, $\mu^f = 1.0$,

$\rho^s = 100.0$ and $\mu^s = 1.0 \times 10^5$. The thickness of the leaf is 0.0212. A pulsating velocity field, $v_{in} = 1.5y(2-y)\sin(2\pi t/10)$, is applied at the inlet.

Background fluid grid with 81×11 mesh at level-0 is enriched with three levels of hierarchical refinement as shown in Fig. 31. The leaf is modelled with 80 beam elements. A time increment of $\Delta t = 0.02$ is used in order to resolve the motion of the leaf accurately in time. The oscillatory motion of the leaf is tracked and time histories of X- and Y-displacements of the free end of the leaf are presented in Fig. 32. These graphs show that the symmetry of oscillatory motion of the leaf on either side of vertical center is captured quite accurately.

Contour plots of velocity magnitude and pressure at four time instants during one cycle of the oscillatory motion of the leaf, along with its deformed configurations, are presented in Figs. 33 and 34, respectively. Baaijens [3] observed that use of Taylor-Hood family of elements produced unsatisfactory results when the kinematic constraint is enforced with Lagrange multipliers and argued that a discontinuous interpolation of the pressure appears to be mandatory. Therefore, the Crouzeix-Raviart family elements are used in [3]. We believe that when Taylor-Hood family (bi-quadratic interpolation for velocity and linear continuous interpolation pressure) is used, the pressure space within an element is not sufficient to accommodate the pressure-jumps across the immersed boundaries and hence it yields unsatisfactory velocity fields. Where as, the use of same basis for velocity and pressure (Q_2 B-Splines) in the present work allows pressure-jumps even within an element and does not pollute the velocity field.

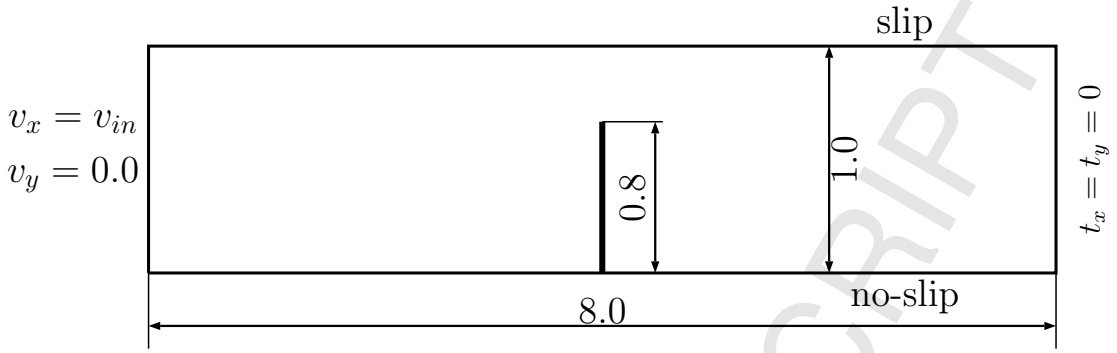


Figure 30: Single-leaf in cross-flow: geometry and boundary conditions.

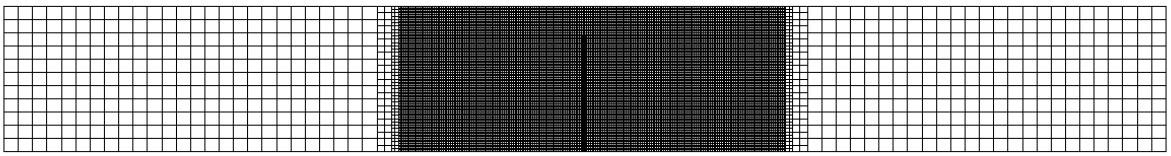


Figure 31: Single-leaf in cross-flow: hierarchical B-Spline mesh. DOF = 60765 + 162 + 243 = 61170.

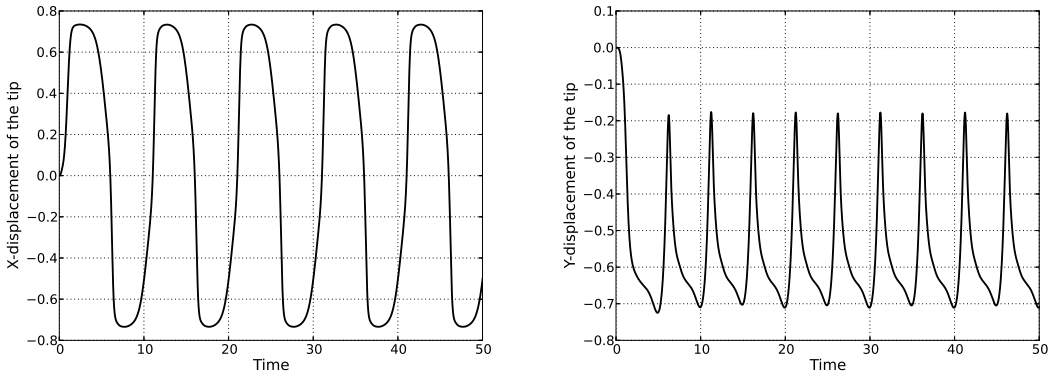


Figure 32: Single-leaf in cross-flow: X- and Y-displacement of the free end of the leaf.

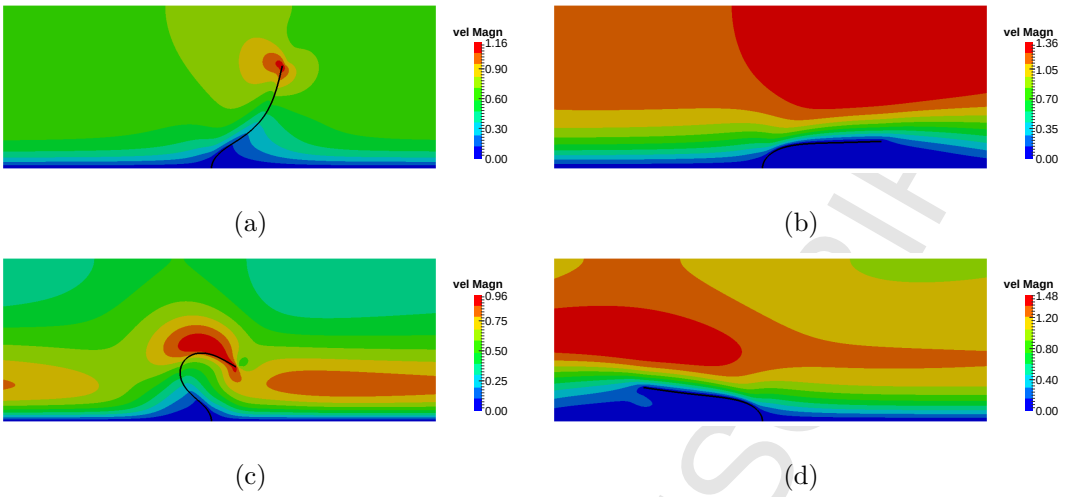


Figure 33: Single-leaf in cross-flow: contour plots of velocity magnitude.

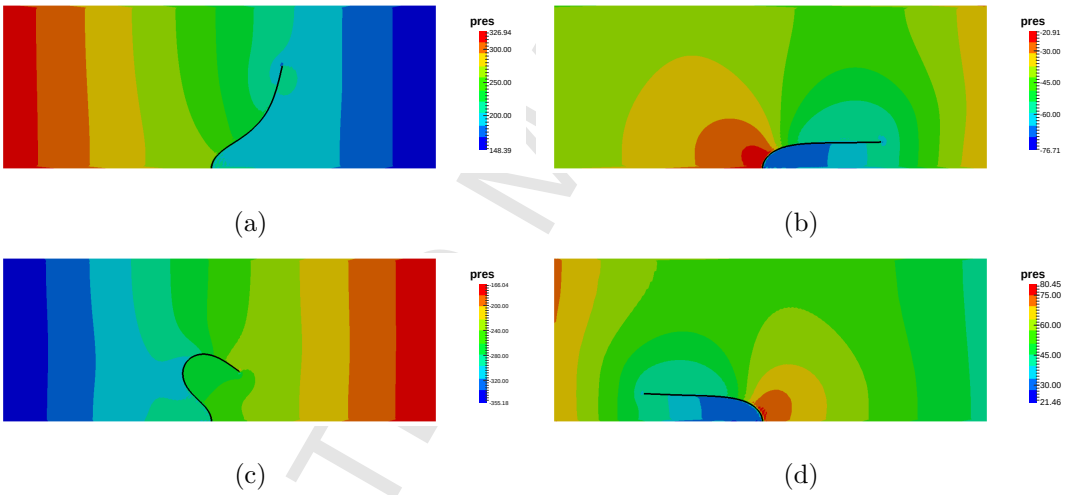


Figure 34: Single-leaf in cross-flow: contour plots of pressure.

7.6. Self-sustained oscillations of a flexible filament

The geometry and boundary conditions of the problem are as depicted in Fig. 35. This problem serves to model the phenomenon of flag-flapping, to understand the locomotion of aquatic animals and micro-organisms, and also study of motion of filament in a soap-film. This example has been studied in [16, 62, 63, 66]. In the present work we do not consider the air-resistance of the fluid and acceleration due to gravity. We would also like to point

out that the task of adding these terms to the present scheme is trivial.

Depending upon the parameters chosen the filament can either settle in a stretched-straight state or sustained oscillating state. A three-level hierarchical B-Spline mesh as shown in Fig. 36 is used for the purpose of analysis. The filament is modelled with 100 geometrically-exact beam elements. One end of the beam, denoted as A in Fig. 35, is fixed. The perturbation is introduced into the system by placing the filament at an angle $arctan(\delta_0)$ to the horizontal direction, at time $t = 0$.

In this example we assess the oscillatory behaviour of the filament under different parameter combinations, same as in [62]. The fixed parameters are: fluid density, $\rho^f = 1.0$, length of the filament $L = 1.0$ and thickness of the filament $d = 0.025$. Simulations are carried out for different values of ρ^s, μ^f, μ^s and δ_0 and the variation of the vertical displacement of the free end of the filament is presented in Figs. 37, 38, 39 and 40, respectively. A timestep of $\Delta t = 0.02$ is chosen for all the simulations. The following observations can be drawn from these simulations.

- Different ρ^s

  As shown in Fig. 37, the filament reaches stretched-straight state for $(\mu^f, \mu^s, \delta_0) =$ (0.01, 100, 0.1) and $\rho^s = 2$ and for the other values it undergoes sustained oscillatory motion. The amplitude of oscillations increases with increasing the values of ρ^s. The frequency of oscillations decreases with increasing the value of ρ^s, as expected.

- Different μ^f

  Fig. 38 shows the filament response for $(\rho^s, \mu^s, \delta_0) =$ (8, 100, 0.1) and different μ^f. The filament settles in stretched-straight state for $\mu^f = 0.01$ and undergoes sustained oscillatory motion for lower values of μ^f. The amplitude of oscillations as well as the frequency increases with decreasing the value of μ^f.

- Different μ^s

  The response of the filament for $(\mu^f, \rho^s, \delta_0) =$ (0.01, 8, 0.1) and different μ^s is shown in Fig. 39. The filament undergoes sustained oscillatory motion for all the values of

μ^s considered in the present work. However, it takes longer time for the oscillations to develop as the value of μ^s is increased.

- Different δ_0

  The effect of initial perturbation δ_0 for $(\mu^f, \rho^s, \mu^s) = (0.01, 8, 100)$ is shown in Fig. 40. In this graph, Y-coordinate of the filament free end is plotted instead of its displacement for the purpose of clean visualisation. The amplitude and frequency of the oscillations remain the same but the amount of time required to establish the sustained oscillations increases with decreasing δ_0.

All the above observations made with respect to the stability of the filament and the amplitude of oscillations obtained from these simulations performed with the proposed scheme match well with those reported in [62].

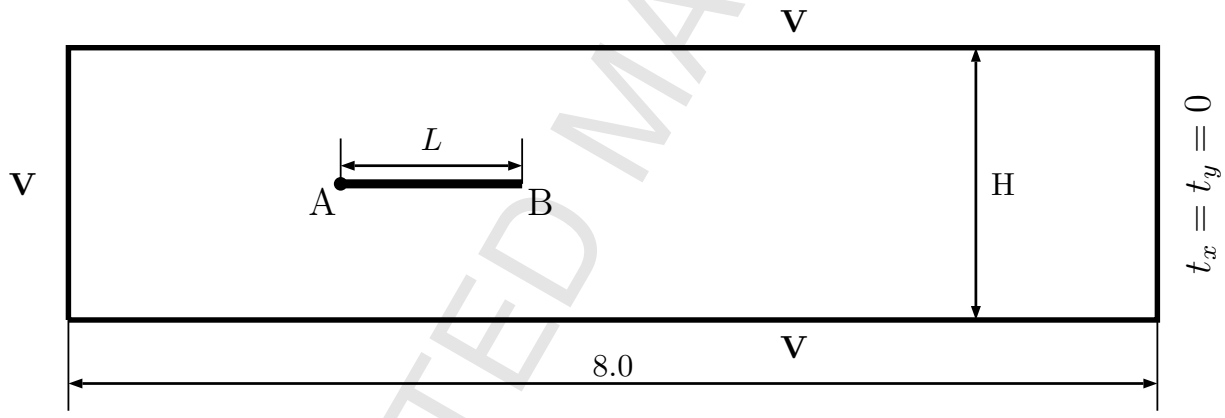


Figure 35: Self-sustained oscillations of a flexible filament: geometry and boundary conditions.

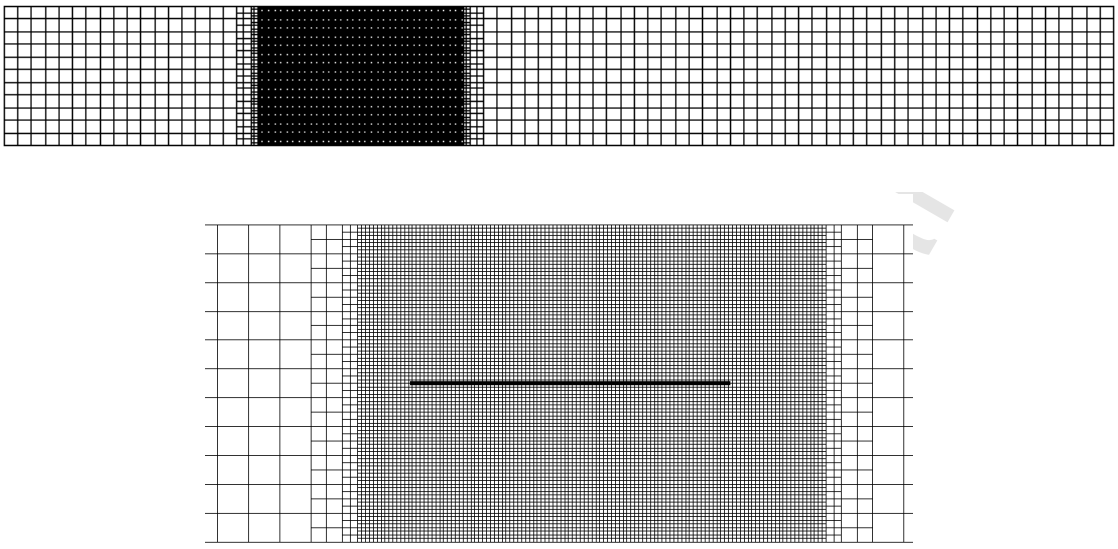


Figure 36: Self-sustained oscillations of a flexible filament: hierarchical B-Spline mesh used for the analysis. DOFs = 42465 + 202 + 303 = 42970.

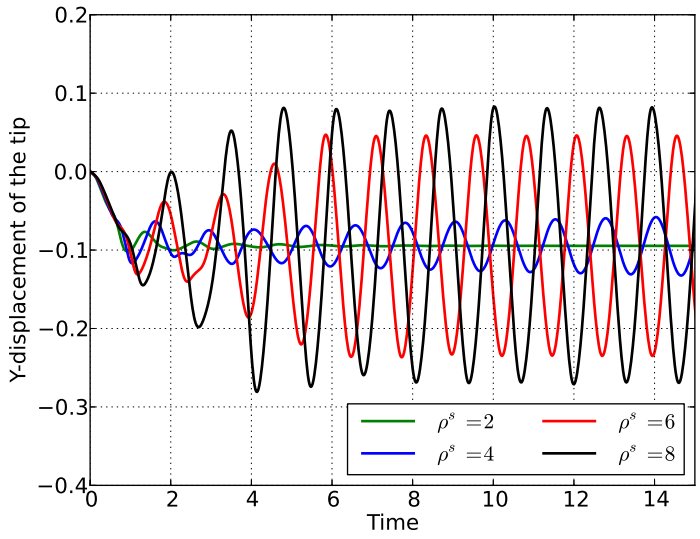


Figure 37: Self-sustained oscillations of a flexible filament: evolution of tip displacement for $(\mu^f, \mu^s, \delta_0) = (0.01, 100, 0.1)$ and different ρ^s.

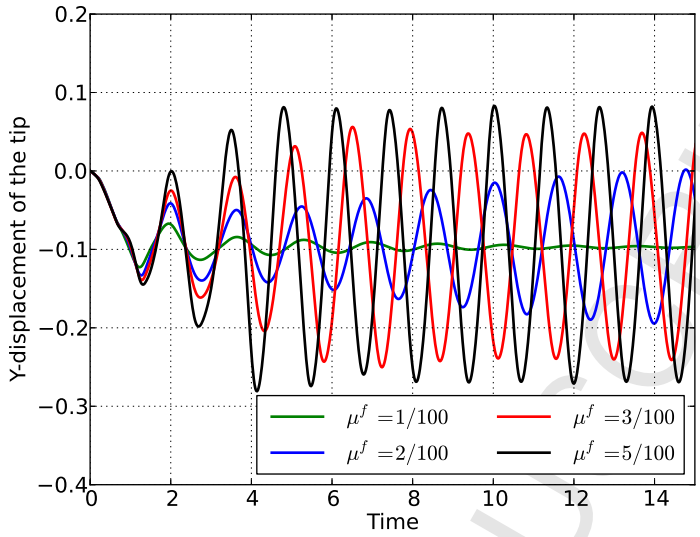


Figure 38: Self-sustained oscillations of a flexible filament: evolution of tip displacement for $(\rho^s, \mu^s, \delta_0) = (8, 100, 0.1)$ and different μ^f.

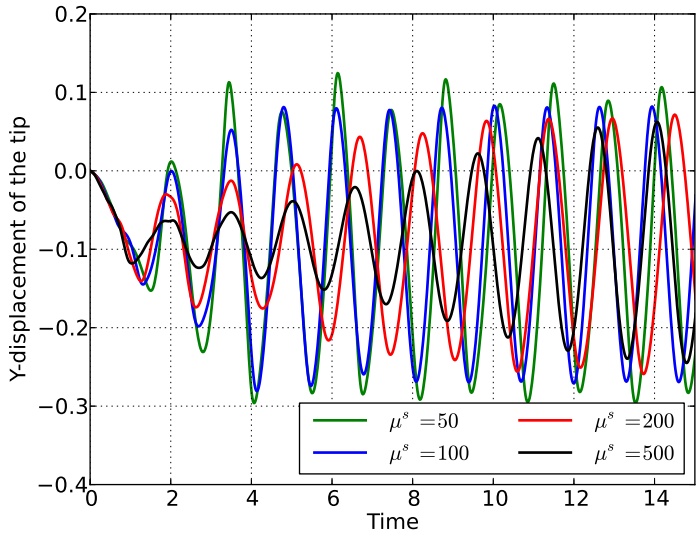


Figure 39: Self-sustained oscillations of a flexible filament: evolution of tip displacement for $(\mu^f, \rho^s, \delta_0) = (0.01, 8, 0.1)$ and different μ^s.

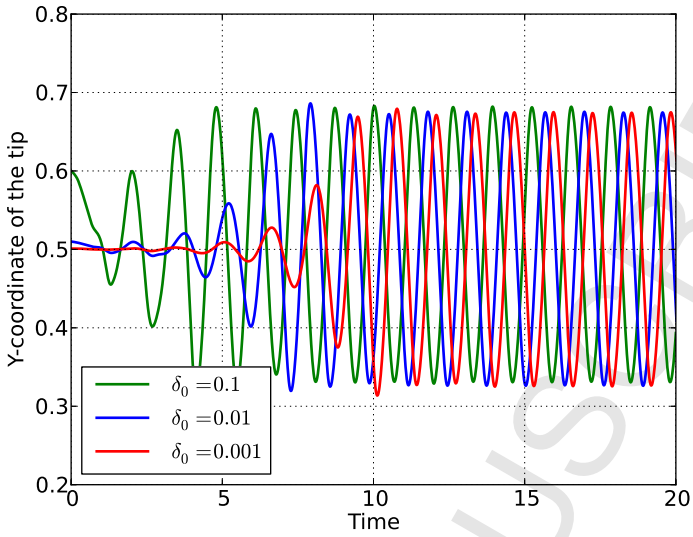


Figure 40: Self-sustained oscillations of a flexible filament: evolution of tip displacement for $(\mu^f, \rho^s, \mu^s) = (0.01, 8, 100)$ and different δ_0.

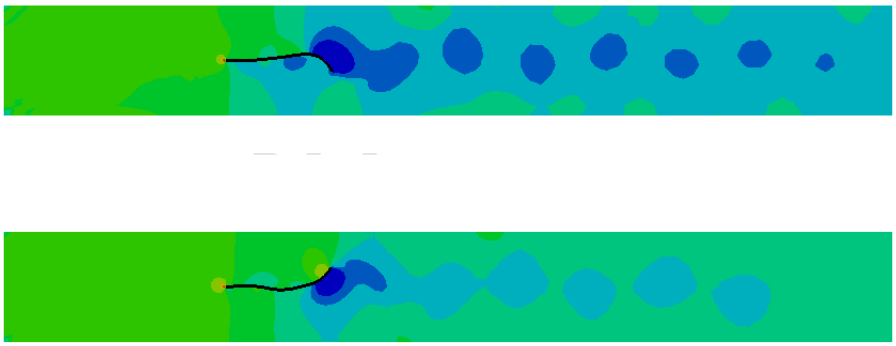


Figure 41: Self-sustained oscillations of a flexible filament: contour plots of pressure at two different time instants, for $(\mu^f, \mu^s, \delta_0) = (0.01, 100, 0.1)$ and $\rho^s = 8$.

8. Summary and conclusions

In this paper we presented a robust numerical scheme for fluid-flexible body interaction based on hierarchical B-Spline Cartesian grids and fictitious-domain/distributed-Lagrange

multipliers. The convergence of the proposed method with respect to spatial and temporal discretisations is assessed and its ability to accurately simulate fluid-flexible body interactions is demonstrated by studying several benchmark examples. The essential features of the proposed scheme are summarised as follows:

- ease of generating Cartesian grid for the fluid and the ability to enrich it with local refinement around immersed bodies using hierarchical B-Splines. This local refinement strategy, which can be easily extended into three-dimensions, results in significant savings in computational time.

- use of equal-order basis functions for velocity and pressure avoids the need to maintain two different grids and/or need to compute two sets of basis functions at each Gauss point while performing matrix computations.

- even though the velocity-pressure combination (equal-order quadratic B-Splines) used in the present work is *inf-sup* unstable, the pressure obtained is sufficiently smooth and the results obtained in FSI simulations match well with the reference values.

- the presented scheme yields accurate results even with large timesteps as the time-integration is second-order unconditionally stable generalized-α method. This feature is essential for simulating real-world problems in 3D.

We conclude this paper by pointing out some of the limitations of the presented scheme and identifying the directions for future developments:

- Use of equal-order quadratic (Q_2) B-Splines for both velocity and pressure, without any pressure-stabilisation, used in this work give sufficiently smooth pressure even though the combination is *inf-sup* unstable. Work should be carried out towards understanding the reasons behind this peculiar behaviour of quadratic B-Splines.

- The boundary of the immersed objects is represented using a set of Lagrange points. This can be improved by representing the boundary using edges in 2D (triangles in 3D)

or using NURBS based discretisation so that the geometry of the immersed boundaries can be represented exactly.

- In this work we have used a direct solver PARDISO [2] to solve the global matrix system of equations. However, this may be prohibitively expensive in 3D which justifies the research work on efficient iterative solvers for this class of problems.

- A fully-coupled monolithic scheme has been used in the present work. Though, this can be used without worrying about added-mass issues, it would be worth exploring staggered or partitioned schemes for the cases where the added-mass is not significant.

Acknowledgements

This research project is funded by Schaeffler Technologies AG & Co. KG, Germany. This support is gratefully acknowledged.

References

References

[1] *Encyclopedia of Computational Mechanics*, volume 1. Wiley.

[2] PARDISO solver project. (http://www.pardiso-project.org).

[3] F. P. T. Baaijens. A fictitious domain/mortar element method for fluid-structure interaction. *International Journal for Numerical Methods in Fluids*, 35:743–761, 2001.

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

[5] P. B. Bornemann and F. Cirak. A subdivision-based implementation of the hierarchical b-spline finite element method. *Computational Methods in Applied Mechanics and Engineering*, 253:584–598, 2013.

[6] M. Braza, P. Chassaing, and H. Ha Minh. Numerical study and physical analysis of the pressure and velocity fields in the near wake of a circular cylinder. *Journal of Fluid Mechanics*, 165:79–130, 1986.

[7] D. Calhoun. A cartesian grid method for solving the two-dimensional streamfunction-vorticity equations in irregular regions. *Journal of Computational Physics*, 176:231–275, 2002.

[8] C. L. Chang and J. J. Nelson. Least-squares finite element method for the Stokes problem with zero residual of mass conservation. *SIAM Journal on Numerical Analysis*, 34:480–489, 1997.

[9] J. Chung and G. M. Hulbert. A time integration algorithm for structural dynamics with improved numerical dissipation: the generalized-α method. *Journal of Applied Mechanics*, 60:353–371, 1993.

[10] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms*. MIT Press, 3rd edition, 2009.

[11] W. G. Dettmer. *Finite element modelling of fluid flow with moving free surfaces and interfaces including fluid-solid interaction*. PhD thesis, Swansea University, 2004.

[12] W. G. Dettmer and D. Perić. An analysis of the time integration algorithms for the finite element solutions of incompressible Navier-Stokes equations based on a stabilised formulation. *Computational Methods in Applied Mechanics and Engineering*, 192:1177–1226, 2003.

[13] W. G. Dettmer and D. Perić. A fully implicit computational strategy for strongly coupled fluid-solid interaction. *Archives of Computational Methods in Engineering*, 14:205–247, 2007.

[14] W. G. Dettmer and D. Perić. A new staggered scheme for fluid-structure interaction. *International Journal for Numerical Methods in Engineering*, 93:1–22, 2013.

[15] C. Diaz-Goano, P. D. Minev, and K. Nandakumar. A fictitious domain/finite element method for particulate flows. *Journal of Computational Physics*, 192:105–123, 2003.

[16] D. J. J. Farnell, T. David, and D. C. Barton. Numerical simulations of a filament in a flowing soap film. *International Journal for Numerical Methods in Fluids*, 44:313–330, 2004.

[17] A. J. Gil, A. A. Carreño, J. Bonet, and O. Hassan. The immersed structural potential method for haemodynamic applications. *Journal of Computational Physics*, 229:8613–8641, 2010.

[18] R. Glowinski, T. W. Pan, T. I. Hesla, and D. D. Joseph. A distributed Lagrange multiplier/fictitious domain method for particulate flows. *International Journal of Multiphase Flow*, 25:755–794, 1999.

[19] R. Glowinski, T. W. Pan, T. I. Hesla, D. D. Joseph, and J. J. Périaux. A fictitious domain approach to the direct numerical simulation of incompressible viscous flow past moving rigid bodies: application to particulate flow. *Journal of Computational Physics*, 169:363–426, 2001.

[20] R. Glowinski, T.-W. Pan, T. I. Hesla, D. D. Joseph, and J. Periaux. A fictitious domain method with distributed Lagrange multipliers for the numerical simulation of particulate flow. *Contemporary Mathematics*, 218:121–137, 1998.

[21] R. Glowinski, T. W. Pan, and J. Periaux. A Lagrange multiplier/fictitious domain method for the Dirichlet problem. Generalization to some flow problems. *Japan Journal of Industrial and Applied Mathematics*, 12:87–108, 1995.

[22] R. Glowinski, Pan T. W., and J. Périaux. A fictitious domain method for Dirichlet problems and applications. *Computational Methods in Applied Mechanics and Engineering*, 111:283–303, 1994.

[23] R. Glowinski, Pan T. W., and J. Periaux. A fictitious domain method for external incompressible viscous flow modeled by Navier-Stokes equations. *Computational Methods in Applied Mechanics and*

Engineering, 112:133–148, 1994.

[24] C. Hesch, A.J. Gil, A. A. Carreño, and J. Bonet. On continuum immersed strategies for fluid-structure interaction. *Computational Methods in Applied Mechanics and Engineering*, 247-248:51–64, 2012.

[25] K. Höllig. *Finite Element Methods with B-Splines*. SIAM, Philadelpia, 2003.

[26] K. Höllig, U. Reif, and J. Wipper. Weighted extended B-spline approximation of Dirichlet problems. *SIAM Journal on Numerical Analysis*, 39:442–462, 2001.

[27] B. Hübner, E. Walhorn, and D. Dinkler. A monolithic approach to fluid-structure interaction using space-time finite elements. *Computational Methods in Applied Mechanics and Engineering*, 193:2087–2014, 2004.

[28] K. E. Jansen, C. H. Whiting, and G. M. Hulbert. A generalized-α method for integrating filtered Navier-Stokes equations with a stabilized finite element method. *Computational Methods in Applied Mechanics and Engineering*, 190:305–319, 2000.

[29] A. A. Johnson and T. E. Tezduyar. Mesh update strategies in parallel finite element computations of flow problems with moving boundaries and interfaces. *Computational Methods in Applied Mechanics and Engineering*, 119:73–94, 1994.

[30] M. M. Joosten, W. G. Dettmer, and D. Perić. Analysis of the block Gauss-Seidel solution procedure for a strongly coupled model problem with reference to fluid-structure interaction. International Journal for Numerical Methods in Engineering, 78:757–778, 2009.

[31] M. M. Joosten, W. G. Dettmer, and D. Perić. On the temporal stability and accuracy of coupled problems with reference to fluid-structure interaction. *International Journal for Numerical Methods in Fluids*, 64:1363–1378, 2010.

[32] C. Kadapa, W. G. Dettmer, and D. Perić. NURBS based least-squares finite element methods for fluid and solid mechanics. International Journal for Numerical Methods in Engineering, 101:521–539, 2015.

[33] D. Kamensky, M. Hsu, D. Schillinger, J. A. Evans, A. Aggarwal, Y. Bazilevs, M. S. Sacks, and T. J. R. Hughes. An immersogeometric variational framework for fluid-structure interaction: application to bioprosthetic heart valves. *Computational Methods in Applied Mechanics and Engineering*, 284:1005–1053, 2015.

[34] D. V. Le, B. C. Khoo, and J. Peraire. An immersed interface method for viscous incompressible flows involving rigid and flexible boundaries. *Journal of Computational Physics*, 220:109–138, 2006.

[35] R. L. Leveque and Z. Li. The immersed interface method for elliptic equations with discontinuous coefficients and singular sources. *SIAM Journal on Numerical Analysis*, 31:1019–1044, 1994.

[36] R. L. Leveque and Z. Li. Immersed interface method for Stokes flow with elastic boundaries or surface tension. *SIAM Journal on Scientific Computing*, 18:709–735, 1997.

[37] Z. Li. The immersed interface method using a finite element formulation. *Applied Numerical Mathematics*, 27:253–267, 1998.

[38] Z. Li and M. Lai. The immersed interface method for the Navier-Stokes equations with singular forces. *Journal of Computational Physics*, 171:822–842, 2001.

[39] M. N. Linnick and H. F. Fasel. A high-order immersed interface method for simulating unsteady incompressible flows on irregular domains. *Journal of Computational Physics*, 204:157–192, 2005.

[40] C. Liu, X. Sheng, and C. H. Sung. Preconditioned multigrid methods for unsteady incompressible flows. *Journal of Computational Physics*, 139:35–57, 1998.

[41] R. Mittal and G. Iaccarino. Immersed boundary methods. *Annual Review of Fluid Mechanics*, 37:239–261, 2005.

[42] T. Muroi and M. Kawahara. A fictitious domain method with the distributed Lagrange multiplier for incompressible viscous flow around moving particle. *Journal of Algorithms and Computational Technology*, 3:75–93, 2009.

[43] M. Nagai and M. M. Kawahara. A fictitious domain method with distributed Lagrange multiplier for particulate flow. *International Journal for Computational Methods in Engineering Science and Mechanics*, 8:115–122, 2007.

[44] X. D. Niu, C. Shu, Y. T. Chew, and Y. Peng. A momentum exchange-based immersed boundary-lattice Boltzmann method for simulating incompressible viscous flows. *Physics Letters A*, 354:173–182, 2006.

[45] N. A. Patankar, P. Singh, D. D. Joseph, R. Glowinski, and T. W. Pan. A new formulation of the distributed Lagrange multiplier/fictitious domain method for particulate flows. *International Journal of Multiphase Flow*, 26:1509–1524, 2000.

[46] C. S. Peskin. The immersed boundary method. *Acta Numerica*, 11:479–517, 2002.

[47] L. Piegl and W. Tiller. *The NURBS Book (Monographs in Visual Communication)*. Springer-Verlag, New York, 1997.

[48] D. F. Rogers. *An Introduction to NURBS With Historical Perspective*. Academic Press, San Diego, CA, 2001.

[49] M. E. Rosar and C. S. Peskin. Fluid flow in collapsible elastic tubes: a three-dimensional numerical model. *New York Journal of Mathematics*, 7:281–302, 2001.

[50] T. Rüberg and F. Cirak. Subdivision-stabilised immersed b-spline finite elements for moving boundary flows. *Computer Methods in Applied Mechanics and Engineering*, 209-212:266–283, 2012.

[51] T. Rüberg and F. Cirak. A fixed-grid b-spline finite element technique for fluid-structure interaction. *International Journal for Numerical Methods in Fluids*, 74:623–660, 2014.

[52] D. Russell and Z. J. Wang. A cartesian grid method for modeling multiple moving objects in 2D

incompressible viscous flow. *Journal of Computational Physics*, 191:177–205, 2003.

[53] P. H. Saksono, W. G. Dettmer, and D. Peric. An adaptive remeshing strategy for flows with moving boundaries and fluid-structure interaction. *International Journal for Numerical Methods in Engineering*, 71:1009–1050, 2007.

[54] H. Samet. *Foundations of Multidimensional and Metric Data Structures*. Morgan Kaufmann Publishers, San Francisco, 2006.

[55] R. A. K. Sanches, P. B. Bornemann, and F. Cirak. Immersed b-spline (i-spline) finite element method for geometrically complex domains. *Computational Methods in Applied Mechanics and Engineering*, 200:1432–1445, 2011.

[56] D. Schillinger, L. Dedè, M. A. Scott, J. A. Evans, M. J. Borden, E. Rank, and T. J. R. Hughes. An isogeometric design-through-analysis methodology based on adaptive hierarchical refinement of NURBS, immersed boundary methods, and T-spline CAD surfaces. *Computer Methods in Applied Mechanics and Engineering*, 249-252:116–150, 2012.

[57] T. E. Tezduyar. Stabilized finite element formulations for incompressible flow computations. *Advances in Applied Mechanics*, 28:1–44, 1991.

[58] D. Vandevoorde and N. M. Josuttis. *C++ Templates: The Complete Guide*. Pearson Education, 1st edition, 2003.

[59] W. A. Wall. *Fluid-Struktur Interaktion mit stabilisierten Finiten Elementen*. PhD thesis, University of Stuttgart, 1999.

[60] T. Wick. Flapping and contact FSI computations with the fluid-solid interface-tracking/interface-capturing technique and mesh adaptivity. *Computational Mechanics*, 53:29–43, 2014.

[61] J. Yao, G. R. Liu, D. A. Narmoneva, R. B. Hinton, and Z. Zhang. Immersed smoothed finite element method for fluid-structure interaction simulation of aortic valves. *Computational Mechanics*, 50:789–804, 2012.

[62] Z. Yu. A DLM/FD method for fluid/flexible-body interactions. *Journal of Computational Physics*, 207:1–27, 2005.

[63] J. Zhang, S. Childress, A. Libchaber, and M. Shelley. Flexible filaments in a flowing soap film as a model for one-dimensional flags in a two-dimensional wind. *Nature*, 408:835–839, 2000.

[64] L. T. Zhang and M. Gay. Immersed finite element method for fluid-structure interactions. *Journal of Fluids and Structures*, 23:839–857, 2007.

[65] Z. Q. Zhang, J. Y. Yao, and G. R. Liu. An immersed smoothed finite elementmethod for fluid-structure interaction problems. *International Journal for Computational Methods*, 8:747–757, 2011.

[66] L. Zhu and C. S. Peskin. Simulation of a flapping flexible filament in a flowing soap film by the immersed boundary method. *Journal of Computational Physics*, 179:452–468, 2002.

[67] O. C. Zienkiewicz and R. L. Taylor. *The Finite Element Method for Solid and Structural Mechanics*. Elsevier Butterworth and Heinemann, Oxford, England, sixth edition, 2005.