FLUID-SOLID INTERACTION BY NON-CONFORMING MESH METHODS

JUAN GUILLERMO GARCÍA NAVARRO

UNIVERSIDAD PONTIFICIA BOLIVARIANA
ESCUELA DE INGENIERÍA
DOCTORADO EN INGENIERÍA
MEDELLIN
2020

FLUID-SOLID INTERACTION BY NON-CONFORMING MESH METHODS

JUAN GUILLERMO GARCIA NAVARRO

Thesis submitted for the degree of Doctor en Ingeniería

Supervisors:
CÉSAR NIETO LONDOÑO, PhD.
CAMILO ANDRÉS BAYONA ROA, PhD.

UNIVERSIDAD PONTIFICIA BOLIVARIANA
ESCUELA DE INGENIERÍA
DOCTORADO EN INGENIERÍA
MEDELLIN
2020

Declaratoria de originalidad

26 de Octubre de 2020

Juan Guillermo García Navarro

"Declaro que este trabajo de grado no ha sido presentado con anterioridad para optar a un título, ya sea en igual forma o con variaciones, en ésta o en cualquiera otra universidad". Art. 92, parágrafo, Régimen Estudiantil de Formación Avanzada.

Dedication

A mi esposa Natalia
y a mi hija Gabriela

Acknowledgements

Primero que todo quiero agradecer a Dios por este logro. Durante este tiempo he tenido que sobrepasar muchas pruebas, tal vez algunas de las más difíciles que he tenido, pero que también me ha regalado los momentos más bonitos al lado de mis seres queridos. Este tiempo me ha enseñado a valorar las cosas más importantes de la vida.

Quiero agradecer a mi asesor Cesar Nieto por todo lo que aprendí de él a través de los años. También a mi otro asesor Camilo Bayona, quien llegó en el momento más crítico de esta investigación, su aporte a este trabajo es invaluable, una persona que no solo me ayudó intelectualmente, sino que en todo momento me dio ánimos para seguir adelante. Ese complemento te hace una gran persona y por supuesto un gran investigador, y estoy seguro de que estas hecho para grandes cosas, felicitaciones. Quiero agradecer a Diego Francisco Hidalgo, un gran amigo y compañero, siempre las pláticas que tuvimos a cerca de este trabajo fueron enriquecedoras y me ayudaron a tomar decisiones importantes dentro de esta investigación. Tambien agradezco a Joan Baiges por su guía durante la pasantía en la Universidad Politécnica de Cataluña.

Por supuesto a mis padres, su apoyo ha sido incondicional en cada momento de mi vida y sobre todo durante esta etapa de mi vida profesional. Por utlimo y con gran alegría quiero agradecer a mi esposa, la mujer que ha estado conmigo en todo momento, apoyándome, escuchándome, y sufriendo conmigo. Eres la gran mujer que Dios puso en mi camino y estoy feliz de tenerte como compañera.

Contents

List of Figures

List of Tables

Abstract

This thesis aims to better understand the numerical approximation of the fluid-solid interaction that involves large deformations and displacements of an immersed solid within the flow field. It can be applied as an engineering tool for improving mechanical component design that typically leads to various numerical and algorithmic challenges.

The numerical strategy in this thesis uses fixed and non-conforming mesh methods due to their ability to solve the fluid flow in a static mesh even with considerable displacements of internal boundaries. But several other numerical ingredients are developed in this work to overcome instability issues that may appear in the numerical solution and to make affordable such computation. One is the application of the Variational Multi-Scale (VMS) framework to construct a stable discrete formulation of the incompressible flow equations. Others have to do with non-conforming mesh methods.

The sharp interface methods implemented in this work can accurately define the boundaries of the immersed solid, however, the moving boundary randomly cuts the cells of the fluid mesh triggering the following two difficulties. The first is the prescription of no-slip boundary conditions over the solid surface in the case of moving cell edges that do not coincide with the nodes of the fluid mesh. Yet, these conditions are imposed weakly by adding new terms to the discrete fluid flow formulation, and specifically, by using a modified version of the *Nitsche's* method to avoid user-defined parameters. The second is the instabilities that may appear at the fluid-solid interface due to the lack of information and the ill-conditioning of the discrete linear system of equations associated with highly-covered fluid cells. Our strategy is to apply the Ghost Penalty method by adding some extra terms to the Variational Multi-Scale formulation.

Also, an implicit approach is adopted to deal with the strong coupling between the fluid and the solid solutions. With this method in hand, some numerical examples are demonstrated, including the solution of well-known benchmark problems and some proposed problem settings. One original application is the study of an auto-rotating seed from a two-dimensional perspective.

Keywords – NCMM, FSI, Variational Multi-Scale, Ghost Penalty method, Nitsche method, Triparis seed

1 Introduction

1.1 Prologue

The study of Fluid-Solid Interaction (FSI) arises from the need of analyzing the coupled response that can occur between a fluid flow and a deforming solid. The simultaneous analysis becomes essential when those phenomena feedback each other, making it unsuitable to split the analysis into isolated solid or fluid solutions. Some interesting applications such as flapping foils or self-rotation seeds are interesting in the context of FSI problems because the solid immersed in the flow field can experience large deformations and displacements, which leads to several scientific challenges.

Numerical simulations have become the most efficient methodology to study these types of FSI problems. This has been explained since computations can be easily addressed for a wide range of problem definitions [2]. Interest in achieving accurate FSI simulations has grown in the last few decades, mainly driven by computational fluid dynamics-assisted design, but also to gain a better understanding of the core of the physical phenomena. This thesis aims to better understand this phenomenon since it can be applied as an engineering tool for improved mechanical component design, for instance in the sense of wind turbine development. Hence, it presents a numerical methodology for solving FSI problems that can deal with Newtonian and incompressible fluid flows interacting with solids subjected to large displacements and/or deformations. The direct consequence of this work has been the possibility to simulate some original and challenging engineering FSI applications, such as the aerodynamic applications involving large trajectories of objects immersed in fluids.

One of the scientific issues is how to ensure a stable and accurate coupled solution in both physical domains: the fluid flow domain and the deforming solid domain. The answer depends on the topological treatment and information sharing between the domains. Two strategies can be chosen in this regard. Since the type of problems studied in this thesis involve a solid domain occupying a region of the fluid domain, the first strategy completely separates the domains by subtracting overlapped regions, while the overlapped region is accounted in the second approach, also producing redundancy [3]. The communication between the fluid and solid also differs for the two approaches: while it is achieved solely through their interface for the first approach, in the case of the second it must be performed through the whole covered region. The second strategy is the so-called Non-Conforming Mesh Methods (NCMM) in Fixed-Mesh Methods. It has been preferred for the immersed solid problems because there is no need of deforming the fluid domain. However, defining the overlapped region of the immersed solid is not trivial: the way to describe the immersed interface demands enhanced algorithmics. Another problem has to do with the numerical approximation of the incompressible Navier-Stokes equations, which represents an active research topic itself in computational mechanics, and the interaction with a deforming solid's structure makes it even harder.

Stabilized based FSI formulations have been recently developed to overcome these issues, which include a great variety of numerical ingredients working in line. The Variational Multi-Scale (VMS) [4] method has been proved to stabilize the convection-dominated fluid flow problems or when the interpolation spaces used for pressure and velocity variables do not satisfy the *inf-sup* condition. This specific framework is adopted and extended in the present thesis to achieve stable fluid flows in Fixed-Mesh Methods for FSI problems. Another ingredient is the weak imposition of no-slip conditions at the fluid-solid interface through the inclusion of some penalization terms to the stabilized formulation. The cut elements are also stabilized by using the Ghost Penalty method with a recent approach that accounts for the orthogonal projection to the finite element space.

The developed numerical methodology has proven to be successful in evaluating engineering problems, some of which are addressed in this document. The first set of these problems deal with the transient analysis of a throttled flow inside a channel by the constraining effect of different types of valves. In those cases, the fluid flow is only affected by the displacements of the solid obstacles, which are not solved but their movement is predefined. The second set of addressed problems involves the partial solution of the fluid-structure coupling in the sense that the fluid effects only displace the solid, being considered as a rigid solid. In these practical applications the aerodynamic forces are considered to displace the immersed obstacle, which are created by the interaction of the viscous flow past the geometric profile. Finally, the last set of problems includes the evaluation of the *Triplaris Americana* auto-rotating seed by fully coupled two-dimensional FSI analysis of different cut sections of the seed.

1.2 Literature Review

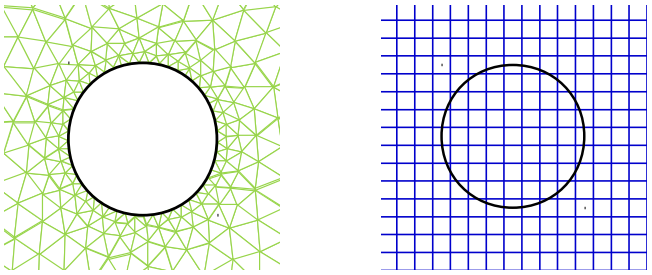
The FSI phenomena have been studied since the 1930s [5], but the interest has grown in recent decades due to the need of understanding high-performance structures that interact with fluid flows. Some industries such as aerospace, marine and bioengineering require a great deal of FSI analysis in their tasks. In the aerospace field, for instance, FSI analysis is used to test critical aircraft components and to ensure their proper performance during flight operation [6]. FSI studies are also frequently applied to wing, helicopter blades [7], wind turbine blades [1], and turbomachinery component design [5]. Several engineering efforts have been committed by the maritime industry in improving the ship structures, especially regarding the geometry and material selection of the hull and propeller. It has also been applied to predict the performance of newly developed materials in aerospace applications [8]. Additionally, high-speed turbomachinery exhibits high strains and stresses in their structure [9]. Bioengineering is another area of extensive FSI research. One of the prevailing topics in physiology is the analysis of the blood and human body tissue interaction, and the prediction of a certain tissue responds to natural and artificial conditions of the blood flow [10]. In other engineering fields, there has grown a recent interest in flexible structures that interact with fluids in a non-conventional way, such as wrapped structures like parachutes or airbags, which experience high deformation

rates [11, 12].

FSI analysis is typically addressed rather by numerical simulation or experimental testing [2]. The former is mostly used during the early design stages, while the latter is implemented for assuring the final prototypes. But numerical simulations have intrinsic advantages over experimental testing, including their relatively low cost, rapid evaluation of different configurations, and easy data collection [13]. Unfortunately, there is not a general FSI numerical methodology for all FSI problems [1]. Several applications may require high accuracy, regardless of the computational cost, but in most industrial applications a low accuracy of the numerical description is accepted [6]. Hence, the construction of the numerical method may depend on its final application and requirements.

1.2.1 Numerical methods

Numerical methods for FSI are classified as Moving Mesh Methods and Fixed Mesh Methods. The classification corresponds well to the information transfer strategy between the solid and fluid domains. In this regard, Conforming Mesh Methods (CMM) and Non-Conforming Mesh Methods (NCMM) are related directly to these definitions. In the first approach, there is a defined interface that separates the fluid domain from the solid domain (see Figure 1.1(a)), whereas in the second approach the interface is immersed in the fluid mesh (see Figure 1.1(b)).



(a) **(b)**

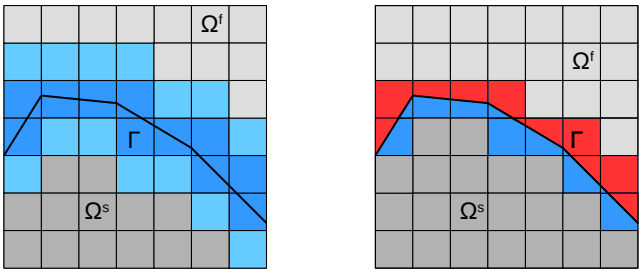
Figure 1.1: Conforming mesh (a) and non-conforming mesh (b). Taken from [1]

CMM are favored in FSI numerical methods due to their easy-to-deploy and stability properties. They have been applied in a variety of industries such as in the aerospace industry to simulate parachute deployment [14, 15] or insect flight aerodynamics [16], as well as in bioengineering to assess the movement of blood within deforming valves [17]. The Arbitrary Lagrangian-Eulerian method (ALE) is the departing point for most of the CMM [18]. The main idea of ALE is to solve the fluid domain using a Eulerian reference frame, while the solid is addressed by a Lagrangian approach. Therefore, the fluid domain is deformed to track the solid motion. But the ALE method is limited to moderate boundary deformations [19], and authors have proposed variations and improvements in order to extend the possible applications. Among these improvements, the re-meshing [20], overset meshes [21, 22] the Extended ALE method [23], and the Fixed ALE method [24], stand above the rest. The motivation behind these methods is to avoid highly deformed

meshes by means of regeneration and control strategies which imply a constant transfer of information between old and new meshes. This particular feature increases the simulation time and produces numerical error [1].

On the other hand, the solid interface does not coincide with boundary of the fluid domain for NCMM, but it is defined in its interior. In this case, there is no need to re-mesh when there are large displacements of the internal solid boundaries inside the fluid domain [25]. Unlike CMM, NCMM are suitable for analyzing problems that involve large deformation and great changes in the structure topology [26]. One drawback is the need for a finer fluid mesh resolution near the interface, such as in boundary layer problems [21] or high gradient zones that require a detailed numerical description of the phenomena. Additionally, a stabilized formulation for the numerical approximation of the Navier-Stokes equations in the fluid domain is mandatory in both CMM and NCMM approaches: the *Galerkin* approximation is enough to solve the elastic solid deformation problem using stantard approaches, however, instabilities like the *inf-sup* compatibility condition between the interpolation spaces of the different variables of the problem may appear for the fluid flow problem. Therefore, the Variational Multi-Scale (VMS) framework has been recently used to stabilize these problems [27], mainly defining the subscales as Algebraic Sub-Grid Scales (ASGS) or Orthogonal Sub-Grid Scales (OSGS) [28].

The way the fluid-solid interface is solved generates another division in NCMM: diffuse and sharp interface methods [3, 29]. In the first type of methods, the interface is defined by a wide layer of grid cells using discrete delta functions. In the second, the interface is tracked by setting a very thin edge. This thin edge can be defined, among several techniques, by a certain function, by modifying the shape of the cells belonging to the interface, or by incorporating jump conditions [30]. One brief depiction of the difference between diffuse and sharp interface methods is presented in Figure 1.2. The immediate consequence is the higher accuracy near the interface obtained by sharp interface methods in contrast to the diffuse interface methods.



(a) **(b)**

Figure 1.2: Diffuse interface (a) and sharp interface (b).

A widely used NCMM and diffuse interface method is the Immersed Boundary Method (IBM) proposed in [31]. In that approach, the interface is defined using elastic constitutive laws that are added to the fluid equations. The IBM has been applied to problems where the mass of the solid object is not relevant, for example, flexible flapping wings, vocal fold vibration, [32], flow patterns around the heart valves [33], movement of biological

swimmers [34], and others. That method has demonstrated to be suitable for FSI problems with elastic solid boundaries. However, it is difficult to apply in FSI problems involving rigid solid bodies due to the increased stiffness of the discrete system. In this sense, the Direct Forcing IBM has overcome these type of problems [35]. Also, the penalty IBM allows the inclusion of the solid mass in the FSI system [36], and the Fictitious Domain Method (also called Penalization Method) represents the rigid solid obstacles as porous media with zero porosity [3]. Although diffuse interface methods have a lower accuracy at the interface, they have successfully been used to model compressible and incompressible fluid-structure interaction problems [37, 38, 39, 36], multiphase fluids [40], and others.

Sharp interface methods, on the other hand, portray an internal classification due to the wide variety of approaches: Cut-Cell methods, Immersed Interface Methods (IIM), eXtended Finite Element Method (XFEM), Finite Cell Methods (FCM), CutFEM, among others. In the Cut-Cell method [41], the interface cells are cut locally and re-meshed so that the faces (or edges) of the new cells match the interface. Its drawbacks are the handling of cut cell topology for complex geometries and the change of Degrees of Freedom (DOF) as a result of re-meshing. Similar to the Cut-Cell method, the XFEM can be used to simulate discontinuities and cracks in materials [42]. This is achieved by improving the interpolation space of the unknowns where the interface intersects with the cells. The XFEM has also been used to model multi-phase fluid problems [43] or FSI problems [44]. Another approach is the inclusion of jump conditions at the solid-fluid interface: for instance, the IIM [45]. In contrast, the CutFEM method [46] includes penalty terms in the fluid equations and modifies the integration scheme over the cut elements. It also includes a stabilization method to avoid instabilities due to bad cuts. This method has been successfully applied in incompressible fluid problems [47], representation of complex geometries [46], multi-phase fluid problems [48], FSI problems [49, 50], and others. A similar approach to the CutFEM method is implemented in the present work, but the cut element stabilization is improved by the VMS framework.

In all these methods, the moving boundary randomly cuts the cells of the fluid mesh triggering numerical difficulties. The first is the tracking scheme of the immersed and moving solid domain. The second is the treatment of the fluid cells belonging to the interface: the strategy on how to account for in the fluid flow equations the partially covered cells at the interface. The third is the prescription of no-slip boundary conditions over the solid surface in the case of moving cell edges that do not coincide with the nodes of the fluid mesh. The fourth are the instabilities that may appear at the fluid-solid interface due to the lack of information and the ill-conditioning of the discrete linear system of equations associated with highly-covered fluid cells. Hence, other numerical ingredients are necessary in the construction of a successful sharp interface scheme.

To track the location of the interface, both the Level-Set method and overlapping meshes can be used. The Level-Set method is used for simple geometries: this method defines a distance function to classify the elements that belong to the fluid or the immersed body. Instead, overlapping meshes are used for more complex geometries: it considers two meshes, one at the background that is precisely the fixed fluid mesh, and another one

at the foreground that represents the immersed solid body.

Once the interface is located, the fluid elements cut by the interface must be carefully treated with the corresponding contribution of those elements to the global assembly of the discrete linear system of equations. In this sense, integrating only the region of cut elements that belong to the fluid domain becomes crucial. One robust strategy for integrating the elements cut by the interface is the sub-triangulation (tessellation) technique that divides the elements and redefines quadrature points to only integrate the part that belongs to the fluid domain [42, 51]. This numerical sub-integration keeps the same DOF of the original mesh, and therefore, preserves the original linear system size [46]. The use of local refinement schemes near the interface has been also used to improve the accuracy of the method, which can be hierarchical [52, 53] or spacetree-based [54] mesh size refinement, but also interpolation order refinement [50]. However, some drawbacks are related to the difficulties to use the mesh refinement technology or the linear system update at each refinement step. Other methods are based on the exclusion of the basis functions of the cut elements from the approximation space, or the recombination of these parts with nearby functions [55]. Besides, the use of Legendre, Lagrange, B-splines, or Nurbs interpolation polynomials have been identified to affect the accuracy of the covered fluid element integration [56, 57, 50].

In the case of the Dirichlet's boundary conditions prescription over the solid interface, those can be imposed weakly by adding new terms to the discrete fluid flow formulation. This method can for example set no-slip conditions for the viscous fluid flow over the immersed and moving solid boundary directly in the fluid cells discrete equations. The main idea of the weak imposition is to penalize the numerical solution, making it equivalent to the Dirichlet value. The method of *Lagrange* multipliers is one classic technique to weakly impose boundary conditions at the immersed interface. However, it increases the DOF of the discrete linear system [50, 58], and restricts the interpolation spaces of the Lagrange multipliers to the fulfillment of an *inf-sup* compatibility condition. More recently, the *Nitsche's* method has been widely used for weakly imposing boundary conditions. It has been implemented along with the CutFEM method [46, 49], the XFEM [59], and the FCM [56] to solve the Poisson, Stokes, Navier-Stokes, and Fluid Structure Interaction (FSI) problems [60, 50]. Several Nitsche-based techniques have been proposed to avoid user-defined parameters. Those methods have been suitable for problems with complex linear systems because those reduce the number of parameters to be included. One of these methods is the penalty-free Nitsche method, which is a non-symmetric modification of the original Nitsche method [61, 56, 47]. Another is the Linked Lagrange method in [62], which arises from the three-field fluid flow equations and demonstrates optimal convergence error for diffusive problems.

The fourth ingredient is the stabilization technique for bad cut elements at the interface. Indeed, lack of information associated with highly-covered elements affects the assembled linear system making it ill-conditioned and triggering instabilities in the discrete solution. That numerical issue can be explained by the small support given by the fluid part of the cut element. One strategy to overcome this issue is the Fictitious Domain Stiffness

method, for which a penalization is included in the formulation only for elements in the interface. It is based on the idea of increasing the contribution of the basis functions of bad cut elements. Preconditioners have also been used for stabilizing the discrete formulation. Those are designed to act over the assembled system matrix, such that the method works independently of the mesh size or the shape of the cut elements [55, 63]. Similarly, other methods exclude the contribution of bad-cut elements to the linear system assembly or replace the local support by using composed functions. A recent approach is the Ghost Penalty (GP) method [64, 46], which increases the weight of the basis functions of bad cut elements by adding extra terms to the variational formulation. Currently, the Ghost Penalty with Orthogonal Projections to the finite space (GPOP) proposed in [65] for Stokes problems has demonstrated improved performance for prescribed Neumann conditions [24] at the boundaries of the fluid domain. The GPOP is an alternative to the classical GP method that is used in the CutFEM method [46], which uses the orthogonal projection to the finite element space of the unknowns [65, 24]. Finally, it has also been demonstrated in [61] that the penalty parameters of the Nitsche method have stabilization properties for the bad cut elements.

Another important ingredient, not necessarily related to NCMM but to FSI, is the coupling method between the fluid and solid domains, which refers to the bidirectional transfer of information between the domains. These strategies can be classified as monolithic or partitioned [1]. In the first approach, fluid and solid problems are solved simultaneously in the same matrix, while in the second approach each problem is solved in separate matrices linked by boundary conditions or source terms. Monolithic approaches are more robust and efficient, but programming is more demanding and the solution is more difficult to converge on the non-linearities of the problems. Partitioned approaches, or so-called segregated approaches, are more stable and allow greater flexibility to select the methods used in each domain, allowing each field to be resolved using robust and efficient solvers that have already been evaluated and validated [66, 67]. In partitioned methods, the choice of the boundary conditions pair at the interface directly affects the stability and convergence of the numerical problem. The most commonly used in the literature is the Dirichlet–Neumann (DN), in which solid displacements are prescribed on the fluid, and tractions are imposed on the solid boundary. If Dirichlet boundary conditions at the immersed fluid interface are weakly imposed, the DN pairs could be seen as a Neumann-Neumann (NN) coupling. Many other strategies have been analyzed in literature and have shown better results than the DN, i.e. the Robin-Neumann (RN) algorithm has shown superior results over other approaches. In [23] it was shown that RN is better than DN for problems affected by the added mass effect: a numerical issue associated with the density value of the fluid and solid. A similar conclusion about RN was stated in [68], where the authors showed that RN is more accurate than Robin–Dirichlet (RD) or DN, but also that RN always converges and that it is insensitive to the added mass effect. Despite the above, NN is the preferred approach in this work, and therefore, the density between domains is always considered to vary in magnitude to avoid stability problems.

Furthermore, partitioned techniques can be classified as explicit or implicit [69] depending

on the temporal integration scheme. In any case, the preferred approach must ensure that the fluid and solid solutions at the interface are balanced at each consecutive time step. Explicit (or loosely coupled) techniques resolve fluid and solid domains once per time step using only the information of the previous time steps, so these techniques do not assure the balance of velocities and stresses at the interface. Small-time steps are typically used in those methods to guarantee the solution stability, but also to avoid the loss of mass and momentum balance. Hence, these methods never are exactly conservative, however, the accumulated temporal error can be improved by implementing predictor-corrector schemes. On the other hand, implicit (or strongly coupled) techniques enforce the balance of velocities and stresses at each time step through an iterative process of resolving both domains until a convergence tolerance is reached; this also implies a high computational cost but a rise in the temporal accuracy.

Indeed, FSI problems for incompressible flows coupled with hyperelastic solids are highly non-linear problems, explained by the inner non-linear condition of the separated problems and to their interaction at the interface. Newton method can be used to solve the coupling non-linearity, however, it is expensive and hard to calculate. In this regard, a fixed point method —like the well known Picard's scheme— is a fair trade-off between robustness and computational cost: at each Picard's iteration, the fluid flow field is solved through the tracking of the dynamic interface and the weak imposition of the no-slip boundary conditions. Then, the fluid tractions are prescribed to deform the solid domain and this process is repeated until a certain convergence criterion is satisfied. This algorithmic strategy has been widely used in literature to deal with FSI problems.

1.2.2 Fluid-Solid Interaction applications

As mentioned before, FSI applications are countless. However, in this work, we are especially interested in two topics: flexible flapping foils, and numerical studies of auto-rotating seeds.

Flapping foil mechanisms are a relatively new field of research that was first explored in the 1980s [70], and have been an active field of study ever since. Flapping foils are systems that are used to extract (harvest) or deliver (propel) energy between a fluid flow and a mechanism of moving solid components through oscillating heaving and pitching movements. When foils are used to extract energy, they work as turbines: they leverage the flutter phenomena for the energy harvesting [71], which is an unwanted aeroelastic effect generated by the transfer of energy from the fluid to a solid structure that is transformed into oscillations that are increasingly larger. The efficiency of these devices is relatively low compared to traditional wind turbines and therefore, experimental and numerical studies have been conducted to understand their operation and improve their efficiency.

The survey of the state of the art classifies related research into oscillating trajectories exploration and airfoil performance examination. The first group of studies focuses on understanding and optimizing the pitching (rotation) and heaving (translation) movements of the flapping foil. The second branch seeks to optimize the airfoil shape to improve the

efficiency of the system. In this last group, some surveys are dedicated to improving the energy harvesting efficiency of the system by modifying the trailing edge of the flapping foil. For instance, a numerical study of a flapping foil with adaptive Gurney flap using the finite volume method with sliding meshes has been reported in [72]. The authors of that work found that the adaptive Gurney flap airfoil has greater efficiency than the fixed Gurney flap airfoil or a conventional trailing edge airfoil. Another study conducted inside a wind tunnel aimed to test a flapping wing with a rigid, passive, and spring-jointed trailing edge [73]. The findings were related to the efficiency increase when the natural frequency of the trailing edge is close to the oscillation frequency. Also that the high rigidity of the spring affects the mean output power. In the research published in [74], IBM was used to study the effect of a flapping foil with a flexible tail. The efficiency was higher for the flexible tail than for the rigid one. This flexible configuration is somewhat idealized since IBM can not consider thick flexible objects. Similarly, increased efficiency of a flapping airfoil with a spring-connected tail has been reported in [75] using the IBM methodology. In general, all these trailing edge mechanisms raise the instantaneous camber of the airfoil and therefore, generate a greater pressure difference between the upper and lower surfaces of the profile allowing it to move easily and producing an increased lift. Other studies aim to investigate the aerodynamic response by completely deforming the flapping foil shape. In one of these evaluations, an oscillating foil with a prescribed deformation was solved using a volume finite method with moving and conforming meshes [76]. They reported an increased power efficiency with respect to the original and non-deformed oscillating foil. A similar analysis has been achieved in [77] and [78] using an IBM. In [79], IBM was used to analyze a combined active control of the heaving and pitching movements with a flapping rigid plate, which reached a significant power coefficient. Worth to mention that in this work we show the capacity of the proposed technology to solve this type of problems without entering into developments or improvements that allow increasing the efficiency of the flapping foil systems.

Regarding auto-rotating seeds, there have been recent studies of their flight due to the interesting physics related to their aerodynamics. Most studies are devoted to the *Maple* seed: a single-leafed seed that involves a special aerodynamic effect generated at very low *Reynolds* number called Leading Edge Vortices (LEVs). This effect increases lift, allowing the seed to remain in the air longer and thus, improving its dispersion. The *Maple* seed has been used as an input to design bio-inspired flight vehicles and wind turbines [80, 81]. Also, many studies have characterized its aerodynamic, from experimental analysis [82, 83, 84], analytical methods [85], and numerical simulations [86, 87, 36, 88, 89]. Some numerical studies, like the ones carried out in [36, 87] used NCMM, and specifically the IBM to describe the flight of the seed. Another auto-rotating seed is the one from the *Triplaris Samara* tree, which is a three-leaf seed that has also aroused a lot of interest and has bio-inspired wind turbines design [90, 91]. This seed also has been studied by means of experimental tests [92, 93], or analytical analysis [94], however no one has made numerical analysis of its geometry.

1.3 Objectives and outline

A numerical methodology based on NCMM and the VMS-FEM stabilized formulation for FSI problems is presented in this thesis. In particular, this methodology can solve Newtonian and incompressible fluid flows at low *Reynolds* number interacting with rigid and hyperelastic solids subjected to large displacements and/or deformations.

The specific content of this work is divided into several topics, which are progressively developed, and will be presented in the document as follows:

- Chapter 2 presents the VMS stabilized formulation of the incompressible Navier-Stokes equations, including the weak imposition of essential boundary conditions, and the stabilized formulation for the cut elements. In this chapter, a dedicated study of the GPOP along the sharp interface scheme is presented, emphasising the inclusion of additional penalty parameters based on the fraction of the element belonging to the fluid. Some applications are presented, most of which use the Level Set method [95] to track the interface. In the case of complex applications, overlapping meshes of the solid and fluid domains are implemented [19, 49, 96]. For example, for the three different cases of valves closing a channel flow.
- Chapter 3 presents the FSI coupling numerical strategy. A partitioned FSI scheme is adopted in this thesis, where each problem is solved in separate matrices that are communicated after each obtained solution. In this sense, an implicit (or strongly coupled) technique is used to enforce the balance of velocities and stresses at each time step. This is achieved through an iterative Picard's scheme with a coupling convergence tolerance. The applied case a flapping foil is presented along with some benchmark problems.
- Chapter 4 presents the aerodynamic simulation and two-dimensional characterization of the auto-rotating *Triplaris* seed that uses the numerical methodology presented in the previous chapters.
- Chapter 5 closes the thesis with some conclusions and the summary of further possible research lines.

1.4 Research dissemination

The research work contained in each chapter is quite self-contained, although this implies the need to repeat some information. The notation is introduced gradually as required and may vary (slightly) from chapter to chapter. This is because each chapter of this thesis is intended to be disseminated in the form of articles in peer-reviewed scientific journals, as indicated below:

1. Chapter 2:
   "Stabilized formulation for fluid flows with immersed moving solids using overlapping and non-conforming meshes", In preparation.

2. Chapter 3:
   "FSI simulations using non-conforming mesh methods by the Ghost Penalty stabilization technique with orthogonal projections defining the fluid as incompressible and the solid as non-linear", In preparation.

3. Chapter 4:
   "Finite Element Simulation of the *Triplaris Americana* seed flight using a Non-Conforming and Overlapping Meshes approach for the fluid-structure interaction", In preparation.

2 Stabilized formulation for fluid flows with immersed moving solids using overlapping and non-conforming meshes

Abstract

The fluid flow can be solved using a fixed domain, as long as it is not required to simulate the effect of an immersed solid having displacements. This chapter investigates a computational methodology that can solve this fluid and structure problem through the use of two separated and overlapping domains. It is based on the Non-Conforming Meshes Method, that leaves fixed the mesh for the fluid on the background but allows high displacements of an overlapped mesh for the solid on the foreground. In the case of the fluid flow solver, a stabilized Finite Element formulation of the incompressible Navier-Stokes equations is used, which comes from the Variational Multi-Scale paradigm. Since the moving solid boundaries do not necessarily coincide with the edges of the background elements, Dirichlet boundary conditions over the fluid velocity are weakly prescribed by employing the Nitsche's method. Local instabilities on the sharp interface are controlled with a Ghost Penalty method that includes the orthogonal projections to the Finite Element space of the interface jumps. The main contribution of the present study is the definition of the algorithmic parameters involved in the cut element stabilization. For the proposed numerical schemes optimal convergence results are obtained for the velocity and pressure variables. The method is applied in two-dimensional dynamic problems involving stationary and moving immersed solids, demonstrating the ability of the proposed computational strategy to simulate real engineering problems.

Highlights

- An overlapping domain method for solving FSI problems is applied.
- The Variational Multi-Scale method is used to solve the incompressible flow.
- The weak imposition of no-slip conditions is applied over the solid interface.
- A Ghost Penalty method with Orthogonal Projections controls cutting instabilities.
- Closing movements for different types of gates are successfully tested.

2.1 Introduction

The vast majority of fluid flow simulations are performed using a fixed computational domain. Solving fluid flows inside a varying domain is also feasible, but the complexity of the computational methods increases. For instance, flow simulations including moving obstacles (rigid or flexible) imply some additional internal moving boundaries of the fluid domain. If the obstacle's rigidity is high, the deformation of the obstacle can be neglected and the problem reduces entirely to reproducing the fluid flow with a predefined motion of some internal boundaries. Lots of practical applications can be represented in that way:

for example, the opening or closing of a valve in a channel [97], a bi-leaflet mechanical heart valve [98, 99], the aerodynamic phenomena of a car overtaking another car [100, 101], or the rotation of propellers [102]. Another is the case of the flow affecting the motion of the rigid obstacle —like in the free-falling of a sphere [103]—, for which the coupling between the tractions of the fluid and the solid must be accounted for in the displacement calculation of the obstacle. Yet, the mechanical flow problem with moving boundaries is hard to be solved and requires the development of additional numerical methods than the approximation of the incompressible fluid flow equations.

In any case, several issues arise when trying to numerically approximate the incompressible *Navier-Stokes* equations inside a changing fluid domain. The Finite Element Method (FEM) has the capacity to solve both fluid and solid problems. One of the classic formulations is the *Galerkin* approximation, which accurately approximates solutions of the heat conduction and elasticity (or generalized *Poisson*) problems: analytic solutions even exist and can be used to evaluate the accuracy of the numerical method. However, for complex problems, like the fluid flow equations, the stability of the Galerkin approximation may be affected. For example, for convection-dominated problems, instabilities may appear. The $inf-sup$ compatibility condition between the interpolation spaces of the different variables of the problem also triggers instabilities for the Galerkin approximation. To overcome these issues, stabilization methods were proposed at the end of the last century: those include, the Streamline-Upwind/Petrov-Galerkin (SUPG) method [104], the Galerkin Least Squares (GLS) method [105], and the Variational Multi-Scale (VMS) method [27]. Since the VMS constitutes a mathematical framework itself, several ways can be chosen to construct the stabilized formulation. Essentially, by including the Orthogonal Sub-Grid Scales (OSGS) [28], the Dynamic Sub-Grid Scales (DSGS) [106], and the Non-linear Sub-Grid Scales (NSGS) [107, 108].

In the discrete version of the fluid domain, one can use a mesh for the fluid that deforms gradually as the boundaries displace, but which can originate ill-shaped elements for large boundary displacements. It is the so-called Conforming Mesh Method (CMM), which typically implies to regenerate the deformed mesh with the consequent increase of the simulation cost and the need of transferring the information from the old to the new mesh. Also, inducing diffusion errors during the transfer operation [1]. Another approach is to use two separated and overlapping meshes for the fluid and solid domains: a static background mesh for the fluid and a dynamic at the foreground for the solid obstacle. That is the so-called Non-Conforming Mesh Method (NCMM) [1], for which the internal boundary can be described as a cut interface of the elements at the background mesh. In the NCMM approach, there is no need to deform any mesh and, therefore, there is no re-meshing procedure, not even for large displacements of the immersed solid obstacle [25]. Hence, NCMM is suitable for solving problems that involve large relative displacements between the solid and the bulk fluid flow, but also when great changes in the fluid domain topology are present, such as closing throats in a flow. Nevertheless, the tracking of the cut interface is not easy and great effort has to be made to numerically represent the moving boundary.

Numerical techniques for defining the cut interface are usually divided into diffused [34, 36, 109, 35] and sharp interface methods [3, 29]. In the diffused interface methods, the interface is represented by adding source terms to the Navier-Stokes formulation. In contrast, the main idea of the sharp interface methods is to cut the fluid elements at the interface by using very thin edges which can be tracked, for example, by the Volume of Fluid (VOF), Level-Set function [95], or an overlapping mesh [49, 96, 19]. Other approaches, like the Cut-Cell Method (CCM) [41, 110], Immersed Interface Method (IIM) [45], eXtended Finite Element Method (XFEM) [42, 43, 44], Finite Cell Method (FCM) [56, 111], and CutFEM method [46, 47, 50, 48], have also been developed to represent the continuity between the fluid and the solid. Some important numerical ingredients are necessary to make sharp interface methods viable. One is the strategy for imposing Dirichlet boundary conditions at the moving interface, like the non-slip condition over the obstacle. The second is the algorithmic tracking of the elements belonging to the interface. The third is the cut cell stabilization that overcomes the lack of numerical information inside bad-cut elements.

Certainly, Dirichlet boundary conditions can not be imposed over the element nodes since the moving interface does not coincide exactly with the elements of the fluid mesh. Yet, these conditions can be imposed *weakly* by adding new terms to the discrete fluid flow formulation [3]. The main idea of the weak imposition is to penalize the numerical solution, making it identical to the Dirichlet value. The method of *Lagrange* multipliers is one classic technique to weakly impose boundary conditions at the immersed interface. However, it increases the Degrees Of Freedom (DOF) of the discrete linear system [50, 58] and restricts the interpolation spaces of the Lagrange multipliers to the fulfillment of an *inf-sup* compatibility condition. More recently, the *Nitsche's* method has been widely used for weakly imposing boundary conditions. It has been implemented along with the CutFEM method [46, 49], the XFEM [59], and the FCM [56] to solve the Poisson, Stokes, Navier-Stokes, and Fluid Structure Interaction (FSI) problems [60, 50]. Several Nitsche-based techniques have been proposed to avoid user-defined parameters. Those methods have been suitable for problems with complex linear systems because those reduce the number of parameters to be included. One of these methods is the penalty-free Nitsche method, which is a non-symmetric modification of the original Nitsche method. That method has suboptimal convergence in L^2-norm error and optimal convergence in H^1-norm error [61, 56, 47]. Another is the Linked Lagrange method in [62], which arises from the three-field fluid flow equations and demonstrates optimal convergence in L^2-norm error for diffusive problems using linear elements.

The second ingredient is the tracking of the elements at the interface and the contribution of those elements to the assembly of the discrete linear system of equations. A common strategy for calculating the numerical integration of the elements cut by the interface is the sub-triangulation (tessellation) technique that divides the elements and redefines quadrature points to only integrate the part that belongs to the fluid domain [42, 51]. This numerical sub-integration keeps the same DOF of the original mesh, and therefore, preserves the original linear system size [46]. The use of local refinement schemes near the interface has been used in [112] as an alternative to the sub-integration schemes. In

the refinement strategy, only those elements in the area of influence of the interface are subdivided recursively until a spatial tolerance is reached: the elements belonging to the fluid domain are accounted for, while those covered by the solid are neglected. The refinement algorithms make use either of hierarchical [56, 53] or tree-based [54] methods, which add a few new DOF to the original linear system. However, some drawbacks are related to the refinement technology development and the linear system update at each refinement step. Besides the interpolation order, the use of Legendre, Lagrange, B-splines, or Nurbs interpolation polynomials have been identified to affect the accuracy of the fluid element integration [56, 57, 50].

The third ingredient is the stabilization technique for bad cut elements at the interface. Indeed, lack of information associated with highly-covered elements affects the assembled linear system making it ill-conditioned and triggering instabilities in the discrete solution. That numerical issue can be explained by the small support given by the fluid part of the cut element, which increases the stiffness matrix condition. One strategy to overcome this issue is Fictitious Domain Stiffness method, for which a penalization is included in the formulation only for elements in the interface. It is based on the idea of increasing the contribution of the basis functions of bad cut elements. Similarly, other methods exclude the contribution of bad-cut elements to the linear system assembly or replace the local support by using composed functions. Preconditioners have also been used for stabilizing the discrete formulation. Those are designed to act over the assembled system matrix, such that the method works independently of the mesh size or the shape of the cut elements [55, 63]. On the other hand, it has also been demonstrated in [61] that the penalty parameters of the Nitsche method have stabilization properties for the bad cut elements. A recent approach is the Ghost Penalty (GP) method [64, 46], which increases the weight of the basis functions of bad cut elements by adding extra terms to the variational formulation. Currently, the Ghost Penalty with Orthogonal Projections to the finite space (GPOP) proposed in [65] for Stokes problems has demonstrated improved performance for prescribed Neumann conditions [24] at the boundaries of the fluid domain.

In the present study, some applied engineering problems are simulated for which non-conforming mesh methods become mandatory. In this sense, two separated domains are used: one at the background for the FEM fluid-flow solver, and another on the foreground that represents the moving solid obstacle. Additionally, a stabilized formulation for the FEM approximation of the Navier-Stokes equations in the fluid domain is achieved by means of the VMS framework. One fundamental contribution of the present methodology is the evaluation of the Ghost penalty stabilization technique of cut elements including the Orthogonal Projection to the finite element space for Navier-Stokes problems. Specifically, the evaluation of that technique when the Dirichlet conditions are imposed weakly through the Nitsche's method with a definition of the algorithmic parameters based on the mesh size.

The remaining parts of this chapter are organized as follows. In Section 2.2, we recall the variational formulation of the fluid flow problem. In that section, we detail the VMS stabilized formulation and the numerical techniques that we implement to deal with

the moving interface. Next, some numerical examples are demonstrated in Section 3.4, including the solution of well-known benchmark problems and their extensions to real engineering problems of valves with high displacements. Finally, some conclusions regarding the achievements of the proposed schemes close the chapter at Section 2.4.

2.2 Finite Element Method approximation of the flow problem

In this section, we recall the complete numerical approximation for solving the fluid flow problem with the moving immersed solid obstacles. First, the Galerkin approximation and the Variational Multi-Scale stabilized formulation of the incompressible Navier-Stokes equations are presented. Then, the immersed interface tracking strategy using both a Level-Set equation and overlapping meshes is described. Also, we detail the sub-triangulation technique that is adopted to integrate the cut elements at the interface. Next, the weak imposition of boundary conditions given by the Penalty and Nitsche methods and the stabilization methods for the cuts based on the Ghost Penalty are described. Finally, the implicit time integration scheme and the linearization scheme for the non-linear flow equations are defined.

2.2.1 Strong form of the flow problem

The fluid and solid domains are denoted, respectively, as $\Omega^f \subset \mathbb{R}^d$ and $\Omega^s \subset \mathbb{R}^d$, being $d = 2$ or 3 the number of space dimensions and $(0, T)$ the time interval. The overall domain Ω is composed of $\Omega = \Omega^s \cup \Omega^f$. The flow problem is defined in the fluid domain Ω^f, with its boundary denoted as $\partial\Omega^f = \Gamma$ and the unit outward normal to the boundary Γ denoted as $\boldsymbol{n}$. Here Γ is defined as $\Gamma = \Gamma_d \bigcup \Gamma_n$, where Γ_d is the boundary where Dirichlet boundary conditions are prescribed and Γ_n is the boundary where Neumann boundary are applied.

The strong form of the incompressible Navier-Stokes equation is such that, given the prescribed velocity $\boldsymbol{u}_d$ in the Dirichlet boundary Γ_d and the prescribed traction $\boldsymbol{t}_n$ in the Neumann boundary Γ_n, find the velocity $\boldsymbol{u} : \Omega^f \to \mathbb{R}^d$ and the pressure $p : \Omega^f \to \mathbb{R}^d$ satisfying

$$\begin{aligned} \rho\left(\partial_t \boldsymbol{u} + \boldsymbol{u} \cdot \nabla \boldsymbol{u}\right) - \mu \Delta \boldsymbol{u} + \nabla p = \boldsymbol{f} \quad & \text{in } \Omega^f, t \in (0,T), \\ \nabla \cdot \boldsymbol{u} = 0 \quad & \text{in } \Omega^f, t \in (0,T), \\ \boldsymbol{u} = \boldsymbol{u}_d \quad & \text{on } \Gamma_d, t \in (0,T), \\ \boldsymbol{\sigma} \cdot \boldsymbol{n} = \boldsymbol{t}_n \quad & \text{on } \Gamma_n, t \in (0,T), \\ \boldsymbol{u} = \boldsymbol{u}^0 \quad & \text{in } \Omega^f, t = 0, \end{aligned} \tag{2.1}$$

where, ρ is the density of the fluid, μ is the dynamic viscosity, and $\boldsymbol{f}$ is the body force. In the case of the viscous fluid, the Cauchy stress tensor is calculated as $\boldsymbol{\sigma} = 2\mu\nabla^s \boldsymbol{u} - p\boldsymbol{I}$ and

there exists a non-slip boundary condition $\boldsymbol{u}_d = 0$ over the immersed solid $\partial\Omega^s$ contour that corresponds, in part, to Γ_d.

For the sake of conciseness, the first two equations in (4.1) can be written in vector form as

$$\boldsymbol{M}\partial_t \boldsymbol{U} + \boldsymbol{\mathcal{L}}\left(\boldsymbol{a};\boldsymbol{U}\right) = \boldsymbol{F} \qquad \text{in } \Omega^f,\, t \in (0,T)\,, \tag{2.2}$$

where $\boldsymbol{U} = [\boldsymbol{u}, p]^\top$ is the vector of unknowns, $\boldsymbol{F} = [\boldsymbol{f}, 0]^\top$ is the vector of forces, and $\boldsymbol{\mathcal{L}}\left(\boldsymbol{a};\boldsymbol{U}\right)$ is the linearized vector form of the differential operator that is defined as

$$\boldsymbol{\mathcal{L}}\left(\boldsymbol{a};\boldsymbol{U}\right) := \begin{bmatrix} \rho\left(\boldsymbol{a}\cdot\nabla\boldsymbol{u}\right) - \mu\triangle\boldsymbol{u} + \nabla p \\ \nabla\cdot\boldsymbol{u} \end{bmatrix}, \tag{2.3}$$

for which the first argument $\boldsymbol{a}$ is the advection velocity field. $\boldsymbol{M}$ is the mass matrix defined as:

$$\boldsymbol{M} = \begin{bmatrix} \rho\boldsymbol{I} & 0 \\ 0 & 0 \end{bmatrix} \tag{2.4}$$

2.2.2 Weak form of the flow problem

Let $L^2(\Omega^f)$ be the space of square-integrable functions in the domain Ω^f, and $H^m(\Omega^f)$ the space of functions whose derivatives are of order up to $m \geq 0$ and that belong to $L^2(\Omega^f)$. The space $H_0^1(\Omega^f)$ is the space of functions in $H^1(\Omega^f)$ that vanish on Γ. The L^2-inner product in Ω^f is denoted by $(\cdot,\cdot)$ and its norm by $\|\cdot\|$.

Let us consider the space $\boldsymbol{\mathcal{W}}_0 = \left(H_0^1(\Omega^f)\right)^d$ as the space where we may seek the velocity for each fixed time t and $\mathcal{Q} \subset L^2(\Omega^f)/\mathbb{R}$ as the space where we seek the pressure for each fixed time t. The weak form of (2.2) is to find $\boldsymbol{U} = [\boldsymbol{u}, p]^\top \in \boldsymbol{\mathcal{W}}_0 \times \mathcal{Q}$, such that appropriate initial and boundary conditions are satisfied, and

$$(\boldsymbol{M}\partial_t\boldsymbol{U}, \boldsymbol{V}) + B(\boldsymbol{U};\boldsymbol{U},\boldsymbol{V}) = L(\boldsymbol{V}), \tag{2.5}$$

for all weighting functions $\boldsymbol{V} = [\boldsymbol{v}, q]^\top \in \boldsymbol{\mathcal{W}}_0 \times \mathcal{Q}$.

In the previous relation, $B(\boldsymbol{U};\boldsymbol{U},\boldsymbol{V})$ and $L(\boldsymbol{V})$ are bilinear and linear forms, respectively, that for the incompressible Navier-Stokes equations are defined as

$$\begin{aligned} B\left(\hat{\boldsymbol{U}};\boldsymbol{U},\boldsymbol{V}\right) =& \rho\int_{\Omega^f}(\hat{\boldsymbol{u}}\cdot\nabla\boldsymbol{u})\cdot\boldsymbol{v}\,d\Omega^f + 2\mu\int_{\Omega^f} D(\boldsymbol{u}) : D(\boldsymbol{v})\,d\Omega^f \\ &- \int_{\Omega^f} p\nabla\cdot\boldsymbol{v}\,d\Omega^f + \int_{\Omega^f} q\nabla\cdot\boldsymbol{u}\,d\Omega^f, \end{aligned}$$

and

$$L\left(\boldsymbol{V}\right) = \int_{\Omega^f}\boldsymbol{f}\cdot\boldsymbol{v}\,d\Omega^f + \int_{\Gamma}\boldsymbol{t}_n\cdot\boldsymbol{v}\,d\Gamma,$$

where $D(\cdot) := \frac{1}{2}\left(\nabla(\cdot) + (\nabla(\cdot)^\top\right)$ is the symmetric gradient operator, and $\widehat{\boldsymbol{U}} = (\widehat{\boldsymbol{u}}, p)^\top$ are the non-linear components in the equations.

2.2.3 Galerkin approximation

Let us define a finite element partition $\boldsymbol{T}_h^f = \{K\}$ of the domain Ω^f composed by non-overlapping elements K with diameter h_K. The diameter of the element partition defined by $h = \max\{h_K | K \in \boldsymbol{T}_h^f\}$. We construct conforming Finite Element spaces $\boldsymbol{\mathcal{W}}_{h,0} \subset \boldsymbol{\mathcal{W}}_0$ and $\mathcal{Q}_h \subset \mathcal{Q}$ for the velocity and the pressure, respectively. These finite spaces are constructed using piece-wise polynomials of order p (not to be confused with pressure) such that those spaces tend to the continuous spaces in the case of $h \to 0$ or $p \to \infty$.

The Galerkin approximation of the weak problem (2.5) can be stated as: find $\boldsymbol{U}_h = [\boldsymbol{u}_h, p_h]^\top \in \boldsymbol{\mathcal{W}}_{h,0} \times \mathcal{Q}_h$, satisfying the initial and boundary conditions, such that $(\boldsymbol{M}\partial_t \boldsymbol{U}_h, \boldsymbol{V}_h) + B(\boldsymbol{U}_h; \boldsymbol{U}_h, \boldsymbol{V}_h) = L(\boldsymbol{V}_h)$, for all weighting functions $\boldsymbol{V}_h = [\boldsymbol{v}_h, q_h]^\top \in \boldsymbol{\mathcal{W}}_{h,0} \times \mathcal{Q}_h$. The previous approximation of the Navier-Stokes equations suffers from instability problems for convection-dominant flows or when the interpolation spaces used for pressure and velocity variables do not satisfy the *inf-sup* condition.

2.2.4 Variational Multi-Scale stabilized element formulation

Some numerical techniques —called stabilized formulations— are used to overcome the instability problems described before [113]. The general idea of stabilization methods is to add some mesh dependent terms into the Galerkin approximation that are intended not to affect convergence and to stabilize the problem. In the sense of

$$\begin{aligned}(\boldsymbol{M}\partial_t \boldsymbol{U}_h, \boldsymbol{V}_h) + B(\boldsymbol{U}_h; \boldsymbol{U}_h, \boldsymbol{V}_h) + S(\boldsymbol{U}_h; \boldsymbol{U}_h, \boldsymbol{V}_h) =& L(\boldsymbol{V}_h), \\ & \forall \quad \boldsymbol{V}_h \in \boldsymbol{\mathcal{W}}_{h,0} \times \mathcal{Q}_h,\end{aligned} \tag{2.6}$$

where $S(\boldsymbol{U}_h; \boldsymbol{U}_h, \boldsymbol{V}_h)$ is the added stabilization term.

A general approach to derive stabilized formulations is the so-called *Variational Multi-Scale* (VMS) framework in [4], which consists of dividing the space of the continuous solution into a finite space and an infinite one (called sub-scales): $\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{X}}_h \oplus \tilde{\boldsymbol{\mathcal{X}}}$, where $\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{W}} \times \mathcal{Q}$ is the continuous space, $\boldsymbol{\mathcal{X}}_h = \boldsymbol{\mathcal{W}}_h \times \mathcal{Q}_h$ is the finite element space, and $\tilde{\boldsymbol{\mathcal{X}}} = \tilde{\boldsymbol{\mathcal{W}}} \times \tilde{\mathcal{Q}}$ is any space to complete $\boldsymbol{\mathcal{X}}_h$ in $\boldsymbol{\mathcal{X}}$. Indeed, $\boldsymbol{\mathcal{X}}_0 = \boldsymbol{\mathcal{X}}_{h,0} \oplus \tilde{\boldsymbol{\mathcal{X}}}_0$ with $\tilde{\boldsymbol{\mathcal{X}}}_0$ any complement of $\boldsymbol{\mathcal{X}}_{h,0}$ in $\boldsymbol{\mathcal{X}}_0$.

When the split of the spaces is applied to the weak form of the flow problem (2.5), and accordingly $\boldsymbol{U} = \boldsymbol{U}_h + \tilde{\boldsymbol{U}}$, being $\boldsymbol{U}_h \in \boldsymbol{\mathcal{X}}_{h,0}$ and $\tilde{\boldsymbol{U}} \in \tilde{\boldsymbol{\mathcal{X}}}_0$, one obtain two equivalent

sub-problems:

$$(\boldsymbol{M}\partial_t \boldsymbol{U}_h, \boldsymbol{V}_h) + \left(\boldsymbol{M}\partial_t \widetilde{\boldsymbol{U}}, \boldsymbol{V}_h\right) + B(\boldsymbol{U}; \boldsymbol{U}_h, \boldsymbol{V}_h) + B(\boldsymbol{U}; \widetilde{\boldsymbol{U}}, \boldsymbol{V}_h) = L(\boldsymbol{V}_h) \quad \forall \quad \boldsymbol{V}_h \in \boldsymbol{\mathcal{X}}_{h,0}, \tag{2.7}$$

$$\left(\boldsymbol{M}\partial_t \boldsymbol{U}_h, \widetilde{\boldsymbol{V}}\right) + \left(\boldsymbol{M}\partial_t \widetilde{\boldsymbol{U}}, \widetilde{\boldsymbol{V}}\right) + B(\boldsymbol{U}; \boldsymbol{U}_h, \widetilde{\boldsymbol{V}}) + B(\boldsymbol{U}; \widetilde{\boldsymbol{U}}, \widetilde{\boldsymbol{V}}) = L(\widetilde{\boldsymbol{V}}) \quad \forall \quad \widetilde{\boldsymbol{V}} \in \widetilde{\boldsymbol{\mathcal{X}}}_0. \tag{2.8}$$

2.2.4.1 Finite element scale

Integrating by parts equation (2.7), and defining subscales zero on element boundaries one obtains

$$(\boldsymbol{M}\partial_t \boldsymbol{U}_h, \boldsymbol{V}_h) + \left(\boldsymbol{M}\partial_t \widetilde{\boldsymbol{U}}, \boldsymbol{V}_h\right) + B(\boldsymbol{U}; \boldsymbol{U}_h, \boldsymbol{V}_h) + \sum_{K \in \boldsymbol{\mathcal{T}}_h^f} \left(\boldsymbol{\mathcal{L}}^*(\boldsymbol{a}; \boldsymbol{V}_h), \widetilde{\boldsymbol{U}}\right)_K = L(\boldsymbol{V}_h) \quad \forall \quad \boldsymbol{V}_h \in \boldsymbol{\mathcal{X}}_{h,0}, \tag{2.9}$$

where $\boldsymbol{\mathcal{L}}^*(\cdot)$ is the *adjoint* operator of $\boldsymbol{\mathcal{L}}(\cdot)$, that in the case of the incompressible Navier-Stokes equations is given by

$$\boldsymbol{\mathcal{L}}^*(\boldsymbol{a}; \boldsymbol{V}_h) = \begin{bmatrix} -\rho\left(\boldsymbol{a} \cdot \nabla \boldsymbol{v}_h\right) - \mu \triangle \boldsymbol{v}_h - \nabla q_h \\ \nabla \cdot \boldsymbol{v_h} \end{bmatrix}.$$

2.2.4.2 Sub-scale

We can also subvert the integration by parts for the diffusive terms, such that (2.8) can be written as

$$\left(\boldsymbol{M}\partial_t \widetilde{\boldsymbol{U}} + \boldsymbol{\mathcal{L}}\left(\boldsymbol{U}; \widetilde{\boldsymbol{U}}\right), \widetilde{\boldsymbol{V}}\right) = \left(\boldsymbol{F} - \boldsymbol{M}\partial_t \boldsymbol{U}_h - \boldsymbol{\mathcal{L}}\left(\boldsymbol{a}, \boldsymbol{U}_h\right), \widetilde{\boldsymbol{V}}\right), \quad \forall \quad \widetilde{\boldsymbol{V}} \in \widetilde{\boldsymbol{\mathcal{X}}}_0, \tag{2.10}$$

At this point, it becomes necessary to approximate the application of the non-linear differential operator into the subscales (at the left hand side of the previous relation) in the sense of

$$\boldsymbol{\mathcal{L}}\left(\boldsymbol{U}, \widetilde{\boldsymbol{U}}\right) \approx \boldsymbol{\tau}^{-1}\left(\boldsymbol{U}\right) \widetilde{\boldsymbol{U}},$$

such that the application is modeled by a matrix of stabilization parameters $\boldsymbol{\tau}^{-1}\left(\boldsymbol{U}\right)$ multiplying the subscales. Also, expression (2.10) can be written in terms of the finite residual $\boldsymbol{R}_h = \boldsymbol{F} - \boldsymbol{M}\partial_t \boldsymbol{U}_h - \boldsymbol{\mathcal{L}}(\boldsymbol{a}; \boldsymbol{U}_h)$ as

$$\left(\boldsymbol{M}\partial_t \widetilde{\boldsymbol{U}} + \boldsymbol{\tau}^{-1}\left(\boldsymbol{U}\right) \widetilde{\boldsymbol{U}}, \widetilde{\boldsymbol{V}}\right) = \left(\boldsymbol{R}_h, \widetilde{\boldsymbol{V}}\right), \quad \forall \quad \widetilde{\boldsymbol{V}} \in \widetilde{\boldsymbol{\mathcal{X}}}_0.$$

Note that all the Left Hand Side (LHS) terms of the previous variational form already belong to the space of subscales $\widetilde{\boldsymbol{\mathcal{X}}}_0$. Hence, the equation for the subscales can be formally reduced to the following ordinary differential equation:

$$\boldsymbol{M}\partial_t\widetilde{\boldsymbol{U}} + \boldsymbol{\tau}^{-1}\left(\boldsymbol{U}\right)\widetilde{\boldsymbol{U}} = \widetilde{\boldsymbol{P}}\left(\boldsymbol{R}_h\right), \tag{2.11}$$

where $\widetilde{\boldsymbol{P}}(\boldsymbol{R}_h)$ stands for the L^2-projection of the finite residual into the space of sub-scales. We call the subscales *quasistatic* because the temporal derivatives of subscales in (2.9) and (2.11) are neglected, and *linear*, as the subscales are not included in the non-linear terms of both the finite scale and sub-scale equations. This means that at all non-linear instances $\boldsymbol{U}$ is replaced by $\boldsymbol{U}_h$.

In the present chapter we use the definitions from [28] to construct the matrix of algorithmic parameters. This is, we use a diagonal matrix with each term calculated as

$$\boldsymbol{\tau}^{-1}\left(\boldsymbol{U}_h\right) = \text{diag}\left(\tau_1^{-1}\left(\boldsymbol{u}_h\right)\boldsymbol{I}, \tau_2^{-1}\left(\boldsymbol{u}_h\right)\right)$$

where τ_1 is the algorithmic parameter associated with the momentum equation and τ_2 is the one for the conservation equation. These can be defined within each element as:

$$\begin{aligned} \tau_1\left(\boldsymbol{u}_h\right) &= \left(\frac{C_1\mu}{h^2} + \frac{C_2\rho|\boldsymbol{u}_h|_\infty}{h}\right)^{-1}, \\ \tau_2\left(\boldsymbol{u}_h\right) &= C_3\mu + C_4\rho|\boldsymbol{u}_h|_\infty h, \end{aligned} \tag{2.12}$$

where $C_1 = 4.0$, $C_2 = 2.0$, $C_3 = 1.0$ and $C_4 = 1.0$ are algorithmic parameters independent of the element size h and the velocity, and $|\boldsymbol{u}_h|_\infty$ is the $L^\infty-$norm of $\boldsymbol{u}_h$ within each element.

At this point, we can use several different approaches that will depend on the definition of the space of the subscales. Some of those definitions include the transient description of the subscales [106], the inclusion of the subscales in the non-linear terms [108], and the term-by-term definition of the subscales [114] which leads to a non-residual approach. Nevertheless, we restrain this work to the application of the two most studied subscales definitions. The first one is the *Algebraic Sub-Grid Scales* (ASGS) approach that considers the space of the sub-scales as the space of the finite residual, and therefore $\widetilde{\boldsymbol{P}}(\boldsymbol{R}_h) = \boldsymbol{R}_h$. With this in hand, the term of the LHS on equation (2.9) involving the subscales becomes

$$S\left(\hat{\boldsymbol{U}}_h; \boldsymbol{U}_h, \boldsymbol{V}_h\right) = \sum_{K\in\boldsymbol{T}_h}\left(\boldsymbol{\mathcal{L}}^*(\hat{\boldsymbol{u}}_h; \boldsymbol{V}_h), \boldsymbol{\tau}(\hat{\boldsymbol{U}}_h)\left(\boldsymbol{F} - \boldsymbol{M}\partial_t\boldsymbol{U}_h - \boldsymbol{\mathcal{L}}(\hat{\boldsymbol{u}}_h; \boldsymbol{U}_h)\right)\right)_K \tag{2.13}$$

The second approach is the so-called *Orthogonal Sub-Grid Scales* (OSGS), which considers that the space of sub-scales is orthogonal to the finite space, and therefore the projection is $\widetilde{\boldsymbol{P}}(\boldsymbol{R}_h) = \boldsymbol{P}_h^\perp(\boldsymbol{R}_h) = \left(\boldsymbol{I} - \boldsymbol{P}_h\right)(\boldsymbol{R}_h)$, being $\boldsymbol{P}_h$ the L^2-projection onto the finite space. Hence, the projection of the residual can be expressed as $\widetilde{\boldsymbol{P}}(\boldsymbol{R}_h) = \boldsymbol{R}_h - \boldsymbol{P}_h(\boldsymbol{R}_h)$, and the term of the LHS of equation (2.9) involving the subscales becomes the following stabilization term:

$$S\left(\hat{\boldsymbol{U}}_h;\boldsymbol{U}_h,\boldsymbol{V}_h\right)=\sum_{K\in\boldsymbol{T}_h}\left(\boldsymbol{\mathcal{L}}^*(\hat{\boldsymbol{u}}_h;\boldsymbol{V}_h),\boldsymbol{\tau}(\hat{\boldsymbol{U}}_h)\left(\boldsymbol{F}-\boldsymbol{M}\partial_t\boldsymbol{U}_h-\boldsymbol{\mathcal{L}}(\hat{\boldsymbol{u}}_h;\boldsymbol{U}_h)\right)\right)_K - \sum_{K\in\boldsymbol{T}_h}\left(\boldsymbol{\mathcal{L}}^*(\hat{\boldsymbol{u}}_h;\boldsymbol{V}_h),\boldsymbol{\tau}(\hat{\boldsymbol{U}}_h)\boldsymbol{P}_h\left(\boldsymbol{R}_h\right)\right)_K. \tag{2.14}$$

Note that the first term of the Right Hand Side (RHS) of the previous equation is equal to the RHS of (2.13). In order to calculate the L^2-projection of the residual onto the finite space $\boldsymbol{P}_h(\boldsymbol{R}_h)$ in (2.14), one need to solve the following variational problem:

$$\int_\Omega \boldsymbol{V}_h\cdot\boldsymbol{P}_h(\boldsymbol{R}_h)\,d\Omega=\int_\Omega \boldsymbol{V}_h\cdot\left(\boldsymbol{F}-\boldsymbol{M}\partial_t\boldsymbol{U}_h-\boldsymbol{\mathcal{L}}(\boldsymbol{u}_h;\boldsymbol{U}_h)\right)\,d\Omega \qquad \forall \quad \boldsymbol{V}_h\in\boldsymbol{\mathcal{X}}_{h,0}.$$

It is worth mentioning that in this work equal linear interpolation spaces are used (for velocity and pressure) because the VMS approach guarantees the $inf-sup$ condition. However, high order definitions could be used, such as those of [115].

2.2.5 Fluid-Structure tracking

An important ingredient that must be accounted for in the numerical formulation is the geometric tracking of the immersed boundary $\Gamma_d=\partial\Omega^s$. As mentioned before, we use the sharp interface approach to track the interface that defines the immersed boundary. To track the location of the interface, we use both the Level-Set method and overlapping meshes. The Level-Set method is used for simple geometries: this method defines a distance function to classify the elements that belong to the fluid or the immersed body. Denoting the Level-Set function as $\zeta(\boldsymbol{x},t)\in\mathbb{R}$, $\boldsymbol{x}\in\Omega$, $t\in(0,T)$, it can be defined as

$$\zeta\left(\boldsymbol{x},t\right)=\begin{cases}+\min_{\boldsymbol{x}_i\in\Gamma_d}\|\boldsymbol{x}-\boldsymbol{x}_i\|, & \boldsymbol{x}\in\Omega^f,\\ -\min_{\boldsymbol{x}_i\in\Gamma_d}\|\boldsymbol{x}-\boldsymbol{x}_i\|, & \boldsymbol{x}\in\Omega^s.\end{cases}$$

Hence, one can trace the interface as the location where it occurs the isovalue $\zeta(\boldsymbol{x},t)=0$. That is, the immersed solid surface can be located from $\Gamma_d=\{\boldsymbol{x}:\zeta\left(\boldsymbol{x},t\right)=0\}$. Also, the mesh of active fluid elements can be calculated using the Level-Set function as $\boldsymbol{T}_h^f=\{K|K\in\zeta\left(\boldsymbol{x},t\right)<0\}$.

Instead, overlapping meshes are used for more complex geometries: it considers two meshes, one at the background that is precisely $\boldsymbol{T}_h^b$, and another one at the foreground $\boldsymbol{T}_h^s$ that represents the immersed solid body. The mesh of active fluid elements can be calculated as $\boldsymbol{T}_h^f=\{K|K\in\boldsymbol{T}_h^b\cap\Omega^f\}$.

Note that, with both methods some elements of the fluid mesh may be partially covered by the fluid. The patch that extends over the elements cut by Γ_d and the first fluid neighbor elements is $\boldsymbol{T}_h^c\subset\boldsymbol{T}_h^f$. Finally, there is also the set of elements that must be neglected in the analysis, which are the fully covered elements by the immersed solid $\boldsymbol{T}_h^o=\boldsymbol{T}_h^b\setminus\boldsymbol{T}_h^f$.

In those special cases, the fluid flow needs not to be solved, but the inactive DOF must be carefully neglected in the assembly process.

Once the interface is located, the elements cut by the interface must be carefully treated. We care for integrating only the region of cut elements that belong to the fluid domain. One robust strategy for integrating the elements cut by the interface is the sub-triangulation technique that divides the cut element. We use this technique to perform the numerical integration of the part of the cut element that belongs to the fluid domain. As it is shown in Figure 2.1, this method redefines the original integration points of the element by relocating their positions. In this figure, the black dashed line represents the interface and the circles represent the new integration points, where only the points that fall within the fluid domain will be considered. The numerical sub-integration keeps the same number of DOF's of the original mesh [51], and therefore, the assembled matrix size is also preserved.

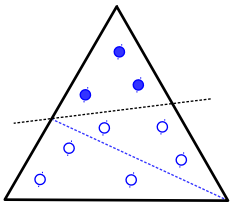


Figure 2.1: Integration splitting of the cut elements

Although the sub-triangulation approach is reliable, some limitations inherent to the method are the following. First, the cut in the covered element is geometrically a straight line between the intersecting points at the edges. The definition of the cut is also restricted to at least two points located in two different edges of the element, such that an interface can not enter or exit through the same edge. One example of this exceptional case is an incoming corner through only one edge: the case of a rectangle moving upwards through a triangular mesh is illustrated in Figure 2.2. This also means that a detail definition of the shape of the interface inside an element is neglected (with the corresponding loss of information and accuracy).

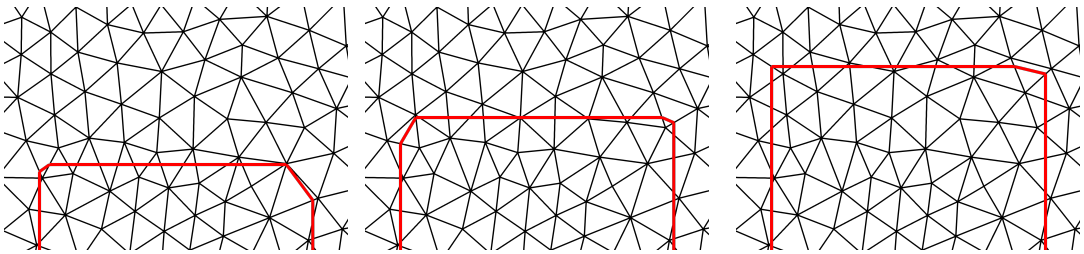


Figure 2.2: Advance of the interface at different times

As mentioned before, there are other integration techniques [116] that overcome these issues by recursively subdividing the elements and using higher order element functions to increase the accuracy of the interface. However, we believe that those methods can be implemented in the proposed methodology as a future work. In the numerical examples of Section 3.4, the interface dispersion can be mitigated by using small time steps and mesh sizes. Also, by using soft interfaces to avoid sharp angles.

2.2.6 Weak imposition of boundary conditions

Since the motivation of the present work is to represent the action of immersed objects in the fluid domain, some specialized numerical methods for imposing boundary conditions in the NCMM strategy are used. These are motivated since the no-slip condition of the immersed solid in the fluid flow can not be prescribed strongly at the nodes of the fluid element. Hence, the Dirichlet type boundary conditions are accounted in a weak way similarly as a Neumann condition at the interface boundary. Specifically, in this subsection we detail two methods for the weak imposition of Dirichlet boundary conditions into the flow equations: one is the Penalty method [117], which is one of the simplest approximations, and the other is the Nitsche method [118], that has been widely used and gives better results than the Penalty method.

Regarding the Penalty method, it seeks to weakly impose the velocity boundary conditions $\boldsymbol{u}_d$ by adding the term $W(\boldsymbol{u}_d; \boldsymbol{U}_h, \boldsymbol{V}_h)$ to the LHS of (2.6). The classical definition of this method is

$$W(\boldsymbol{u}_d; \boldsymbol{U}_h, \boldsymbol{V}_h) = \frac{\alpha}{h} \int_{\Gamma_d} (\boldsymbol{u}_h - \boldsymbol{u}_d) \cdot \boldsymbol{v}_h \, d\Gamma - \int_{\Gamma_d} (2\mu \boldsymbol{n} \cdot \nabla^s \boldsymbol{u}_h - p_h \boldsymbol{n}) \cdot \boldsymbol{v}_h \, d\Gamma, \tag{2.15}$$

where α is a penalty parameter defined to be $\alpha = s_1 \mu + s_2 \rho h \left\| \mathbf{u}_h \right\|_{L^\infty(K)}$, with s_1 and s_2 algorithmic parameters and where α depends on the geometry of the mesh, especially in the geometry of the cut elements. It is precisely this mesh dependent parameters that affect the stability of the method [62]. However, the value of the parameters are not known *a priori*, and they are user-defined values to decide the degree of enforcement of the boundary condition, and also to ensure stability. The natural choice of s_1 and s_2 will be to take them as high as possible to be on the safe side of the stability; however, there could be a compromise between accuracy at the boundary and ill-conditioning of the system.

The first term at the RHS of the previous equation is the penalization term for the weak imposition of the Dirichlet boundary condition that vanishes for the true solution (when $\boldsymbol{u} = \boldsymbol{u}_d$ at the boundary), while the second comes form the integration by parts of the weak form of the fluid equation.

The second approach is the Nitsche method, which includes a term that guarantees the symmetry of equation (2.15) but also vanishes at the boundary. Assuming that $w(\boldsymbol{u}, \boldsymbol{v}) = \int_\Gamma \boldsymbol{u} \cdot \boldsymbol{v} \, d\Gamma$, the symmetrization consist in defining a new term that meets $w(\boldsymbol{v}, \boldsymbol{u}) - w(\boldsymbol{u}, \boldsymbol{v}) = 0$. Certainly, the Nitsche weak imposition leads to the form

$$\begin{aligned} W(\boldsymbol{u}_d; \boldsymbol{U}_h, \boldsymbol{V}_h) = & \frac{\alpha}{h} \int_{\Gamma_d} (\boldsymbol{u}_h - \boldsymbol{u}_d) \cdot \boldsymbol{v}_h \, d\Gamma - \int_{\Gamma_d} (2\mu \boldsymbol{n} \cdot \nabla^s \boldsymbol{u}_h - p_h \boldsymbol{n}) \cdot \boldsymbol{v}_h \, d\Gamma \\ & - \int_{\Gamma_d} (2\mu \boldsymbol{n} \cdot \nabla^s \boldsymbol{v}_h - q_h \boldsymbol{n}) \cdot (\boldsymbol{u}_h - \boldsymbol{u}_d) \, d\Gamma, \end{aligned} \tag{2.16}$$

by adding these terms to the LHS of equation (2.6). Indeed, the last term at the RHS of the previous expression is a symmetry preserving term exclusive to the Nitsche method.

2.2.7 Ghost Penalty stabilization using orthogonal projections to the FE space

In highly convective flows, badly cut elements at the interface need to be treated to avoid instabilities. These issues are caused by the small support given by the convection terms in the stiffness matrix, and therefore, the lack of control may lead to oscillations in the solution near the interface. To this end, stabilization methods for the cut cells are related to the correct transfer of physical information between the fluid and the boundary.

One stabilization method for the cut cell is the so-called Ghost Penalty (GP) method [64], which has been designed to improve the condition number of the stiffness matrix by adding convection to the bad cut elements. In the case of the incompressible Navier-Stokes equations, the GP method penalizes the jumps of the unknowns over inter-element faces by adding a term of the form $S_G\left(\boldsymbol{U}_h;\boldsymbol{U}_h,\boldsymbol{V}_h\right)$ to the LHS of equation (2.6). Let us denote F_g the set of faces (or edges) cut by the interface and those belonging to the first internal layer of fluid connected to them. We use $[\![\cdot]\!]$ to denote the jump operator over the edge f. Let us also use the short-hand notation $\partial_{\boldsymbol{n}}$ for the normal derivative $(\boldsymbol{n}\cdot\nabla)$. The classical GP method [119] introduces the following terms to the fluid flow FE formulation,

$$\begin{aligned} S_G\left(\hat{\boldsymbol{U}}_h;\boldsymbol{U}_h;\boldsymbol{V}_h\right) =& \sum_{f\in F_g}\alpha_\beta\left\langle[\![(\hat{\boldsymbol{u}}_h\cdot\nabla)\partial_{\boldsymbol{n}}\boldsymbol{u}_h]\!],[\![(\hat{\boldsymbol{u}}_h\cdot\nabla)\partial_{\boldsymbol{n}}\boldsymbol{v}_h]\!]\right\rangle_f \\ &+\sum_{f\in F_g}\alpha_{\boldsymbol{u}}\left\langle[\![\nabla\cdot\partial_{\boldsymbol{n}}\boldsymbol{u}_h]\!],[\![\nabla\cdot\partial_{\boldsymbol{n}}\boldsymbol{v}_h]\!]\right\rangle_f \\ &+\sum_{f\in F_g}\alpha_p\left\langle[\![\partial_{\boldsymbol{n}}p_h]\!],[\![\partial_{\boldsymbol{n}}q_h]\!]\right\rangle_f \\ &+\sum_{f\in F_g}\alpha_\nu\left\langle[\![\partial_{\boldsymbol{n}}\boldsymbol{u}_h]\!],[\![\partial_{\boldsymbol{n}}\boldsymbol{v}_h]\!]\right\rangle_f, \end{aligned} \tag{2.17}$$

which are GP operators for the streamline derivative, fluid tractions, pressure and velocity, respectively. Also, $\alpha_\beta,\alpha_{\boldsymbol{u}},\alpha_p$ and α_ν are penalty parameters which are functions of h_K and the physical problem. The main idea of the GP method in equation (2.17) is to reduce the jump of gradients between elements at the interface of the immersed boundary.

More recently, a method based on the orthogonal projection to the finite element space of the unknowns effectively controls the jumps (or instabilities). This method, proposed originally in [65], can be related directly to the OSGS Variational Multiscale stabilized formulation. Hence, we apply the Ghost Penalty with Orthogonal Projection (GPOP) stabilization method into the LHS of the fluid flow finite formulation (2.6):

$$\begin{aligned} S_G\left(\hat{\boldsymbol{U}}_h;\boldsymbol{U}_h,\boldsymbol{V}_h\right) =& \sum\nolimits_{K\in\boldsymbol{T}_h^c}\sigma_G\left[\gamma_1\left(\hat{\boldsymbol{u}}_h\right)\left\langle\nabla\boldsymbol{v}_h,\boldsymbol{P}_h^{\perp}(\nabla\boldsymbol{u}_h)\right\rangle_K\right] \\ &+\sum\nolimits_{K\in\boldsymbol{T}_h^c}\sigma_G\left[\gamma_2\left(\hat{\boldsymbol{u}}_h\right)\left\langle\nabla q_h,\boldsymbol{P}_h^{\perp}(\nabla p_h-\rho\boldsymbol{f})\right\rangle_K\right], \end{aligned} \tag{2.18}$$

where σ_G is a penalty parameter defined by the user, and $\gamma_1\left(\boldsymbol{u}_h\right)$, $\gamma_2\left(\boldsymbol{u}_h\right)$ are algorithmic

constants that depend on the matrix of stabilization parameters $\boldsymbol{\tau}(\boldsymbol{u}_h)$:

$$\gamma_1(\boldsymbol{u}_h) = s_u h^2 \tau_1^{-1}(\boldsymbol{u}_h),$$
$$\gamma_2(\boldsymbol{u}_h) = s_p \tau_1(\boldsymbol{u}_h).$$

Here s_u and s_p are algorithmic parameters independent of the element size h. The orthogonal projections $\boldsymbol{P}_h^{\perp}$ in the cut elements are defined as:

$$\boldsymbol{P}_h^{\perp}(\nabla \boldsymbol{u}_h) = \nabla \boldsymbol{u}_h - \boldsymbol{P}_h(\nabla \boldsymbol{u}_h),$$
$$\boldsymbol{P}_h^{\perp}(\nabla p_h - \rho \boldsymbol{f}) = \nabla p_h - \rho \boldsymbol{f} - \boldsymbol{P}_h(\nabla p_h - \rho \boldsymbol{f}).$$

Following the work in [119], we further penalize the weak imposition and ghost penalty terms using $\alpha\beta^{-1}$ and $\gamma_1\beta^{-1}$ instead of α and γ_1 alone, where β is calculated as the fluid's fraction at each element. This fraction can be calculated as the ratio between the fluid's area and the total element area in the two-dimensional case, or by using the volumes in the three-dimensional case. The modified parameters have demonstrated to improve the control over the boundary cut elements, as it will be presented in the numerical experiments.

2.2.8 Time integration

Let us now comment on how to implement the time discretization for the fluid problem. The time interval $(0, T)$ is divided in a sequence of discrete time steps $0 = t^0 < t^1 < ... < t^N = T$, with a constant time-step, $\delta t > 0$, defining $t^{n+1} = t^n + \delta t$ for $n = 0, 1, 2, ..., N$.

In this work, an implicit time integration scheme to integrate the time derivative of Eq. (2.6) is used. In the case of the first-order temporal derivative of the fluid flow problem, the Backward Differentiation Formula (BDF) scheme is applied. For the time dependent function $y(t)$, the BDF approximation of order $k = 1, 2, ..$, is given by $\delta_k y^{n+1}/\delta t$, with

$$\frac{\delta_k y^{n+1}}{\delta t} = \frac{1}{\delta t}\frac{1}{\gamma_k}\left(y^{n+1} - \sum_{i=0}^{k-1} \phi_k^i y^{n-1}\right),$$

where γ_k and ϕ_k^i are numerical parameters.

Hence, the fully discretized fluid flow problem given the initial conditions $\boldsymbol{U}^0$, and supposing that the subscales at the initial time step are identically zero, for $n = 1, 2, ..$, is to find $\boldsymbol{U}_h^{n+1} \in \boldsymbol{\mathcal{X}}_{h,0}$, such that:

$$\left(\frac{\delta_k \boldsymbol{U}_h^{n+1}}{\delta t}, \boldsymbol{V}_h\right) + B\left(\boldsymbol{U}_h^{n+1}; \boldsymbol{U}_h^{n+1}, \boldsymbol{V}_h\right) + S\left(\boldsymbol{U}_h^{n+1}; \boldsymbol{U}_h^{n+1}, \boldsymbol{V}_h\right)$$
$$+W(\boldsymbol{u}_d; \boldsymbol{U}_h^{n+1}, \boldsymbol{V}_h) + S_G\left(\boldsymbol{U}_h^{n+1}; \boldsymbol{U}_h^{n+1}, \boldsymbol{V}_h\right) = L\left(\boldsymbol{V}_h\right), \qquad (2.19)$$

for all $\boldsymbol{V}_h \in \boldsymbol{\mathcal{X}}_{h,0}$. The previous equation defines the spatial and temporal VMS

approximation of the fluid flow equations. A second-order BDF integration scheme is implemented, which is:

$$\frac{\delta_2 \boldsymbol{U}_h^{n+1}}{\delta t} = \frac{1}{\delta t}\frac{3}{2}\left(\boldsymbol{U}_h^{n+1} - \frac{4}{3}\boldsymbol{U}_h^{n} + \frac{1}{3}\boldsymbol{U}_h^{n-1}\right).$$

For this case, two successive starting values are required when $n = 1$ to solve $\boldsymbol{U}^{n+1}$: the initial condition $\boldsymbol{U}^0$ and $\boldsymbol{U}_h^1$. Hence, at the first time step we calculate $\boldsymbol{U}_h^1$ with the first order BDF approximation, given by

$$\frac{\delta_1 \boldsymbol{U}_h^{1}}{\delta t} = \frac{\boldsymbol{U}_h^{1} - \boldsymbol{U}_h^{0}}{\delta t}.$$

2.2.9 Linearization strategy

The implicit scheme brings the difficulty of solving the non-linearities of the discrete fluid problem. To treat this issue, a linearization strategy that is based on *Picard's* method is implemented. At each step $n+1$, given $\boldsymbol{U}_h^{n+1,i}$ and $\widetilde{\boldsymbol{U}}^{n+1,i}$, the finite element unknowns is used to evaluate $\boldsymbol{U}_h^{n+1,i+1}$ by considering $\boldsymbol{U}_h^{n+1,i}$ in all non-linear terms of Eq. (3.21), being i an iteration counter. The loop is iterated until the L^2-norm of the difference between consecutive finite element solutions is below a given convergence criteria, $|\phi_h^{n+1,i+1} - \phi_h^{n+1,i}| < \varepsilon|\phi_h^{n+1,i+1}|$, where ϕ_h stands for any finite element unknown and ε is the tolerance. When the OSGS or the GPOP methods are accounted for, the orthogonal projection of the finite quantity $\boldsymbol{\phi}_h^{n+1}$ is approximated at each *Picard's* iteration $i+1$ as

$$\boldsymbol{P}^{\perp}\left(\boldsymbol{\phi}_h^{n+1,i+1}\right) \approx \boldsymbol{\phi}_h^{n+1,i+1} - \boldsymbol{P}_h\left(\boldsymbol{\phi}_h^{n+1,i}\right).$$

This is, the L^2-projection onto the finite element space $\boldsymbol{P}_h$ is evaluated with the finite quantity at the end of the previous iteration. In our implementation, the finite element projection is computed using a consistent mass matrix.

2.3 Numerical examples

In this section, numerical results for several flow problems are presented, where the weak imposition of Dirichlet no-slip conditions on the boundary of an immersed object in a fluid's domain is mandatory. The first set of problems are intended to test the Nitsche and Ghost Penalty methods in the VMS stabilized finite element formulation of the incompressible Navier-Stokes equations. In this regard, the manufactured solution inside a two-dimensional squared domain is used to test the numerical error of the methods and to evaluate the definition of the algorithmic parameters in the formulation. The second case is the well-known flow past an infinitely long cylinder at low *Reynolds* number. This two-dimensional case is intended to test the weak imposition of Dirichlet conditions over a dynamic flow problem.

Next, we aim to solve more complex problems that exhibit a strong interaction between the immersed solid with a predefined motion and the fluid flow. In that final set of problems, we evaluate a rising gate in a pipe flow, a closing throat in a pipe flow (inlet hole, or Venturi tube), and the rotating flat body in pipe flow. These last three examples are planned to demonstrate the ability of the proposed methodology in the simulation of applied engineering problems. In all cases, we use finite element unstructured meshes composed by linear triangles.

Some preliminary numerical examples are solved using the OSGS global stabilization method and the Penalty method (2.15) for the weak imposition of essential conditions. In the case of the OSGS stabilization method, it demonstrates a slightly better accuracy than the ASGS. However its non-linear convergence is reduced and the computational effort is larger. Also, some tests with the Penalty method for the weak imposition of boundary conditions demonstrate similar results to the Nitsche method but its converge worsens when large penalty parameters are used. Thus, we use the ASGS and Nitsche methods in the following numerical examples.

2.3.1 Manufactured solution

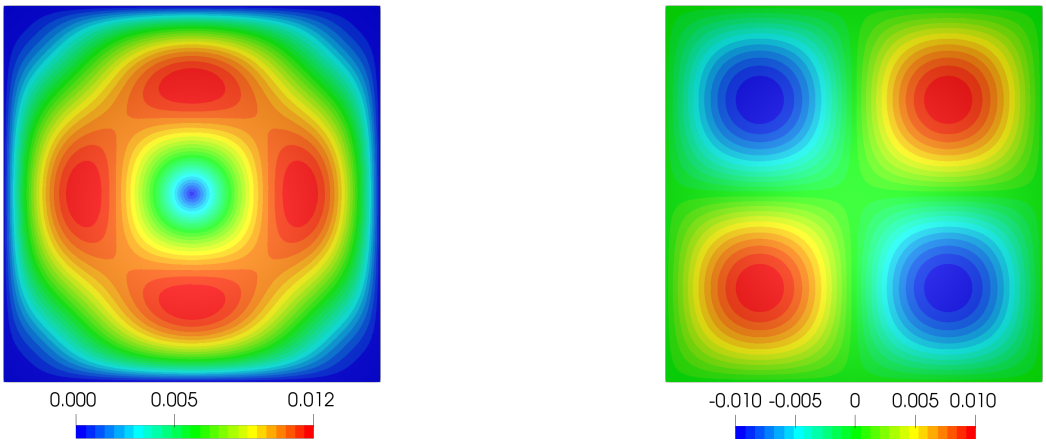
Manufactured solutions [120] are used in this work in order to test the accuracy of the discrete formulations. The main idea of the method of manufactured solutions is to use the differential operator (2.3) applied to an analytical (known) function, and to use this result as a discrete source term. The discrete problem aims to solve the forced setting that implies using the calculated source term. Hence, the obtained numerical solution can be compared against the analytic manufactured solution, and the accuracy of the proposed numerical scheme can be quantified.

In this first numerical example, we solve a manufactured solution flow inside a square domain $\Omega = [0,1]$ m $\times [0,1]$ m. The analytical functions chosen as the exact solution for velocity $\boldsymbol{u}_{ex}$ and pressure p_{ex} are, respectively,

$$\begin{aligned} \boldsymbol{u}_{ex}(\boldsymbol{x}) &= \begin{bmatrix} 2x_1^2 x_2 (x_1-1)^2 (x_2-1)(2x_2-1) \\ -2x_2^2 x_1 (x_1-1)(x_2-1)^2 (2x_1-1) \end{bmatrix} \quad \text{m/s}, & \boldsymbol{x} \in \Omega \\ p(\boldsymbol{x}) &= \mu \sin(2\pi x_1) \sin(2\pi x_2) \quad \text{kg/(m}\cdot s^2), & \boldsymbol{x} \in \Omega \end{aligned}$$

which satisfy the divergence free condition $\nabla \cdot \boldsymbol{u}_{ex} = 0$. A dynamic viscosity $\mu = 0.01$ kg/(m·s) and a density $\rho = 1.0$ kg/m^3 are set in all cases. The contours of the exact velocity magnitude and the exact pressure are shown in Figure 2.3.

The case of a submerged circle of radius $r = 0.25$ m inside the square domain Ω is evaluated first. The external boundaries conditions of the fluid's domain $\Gamma_{1,2,3,4}$ are prescribed in a strong form by calculating the exact velocity at those locations. Instead, the no-slip condition over the interface boundary Γ_5 is imposed weakly. Hence, the



(a) (b)

Figure 2.3: Contour plot of the analytic manufactured solution of (a) velocity magnitude and (b) pressure.

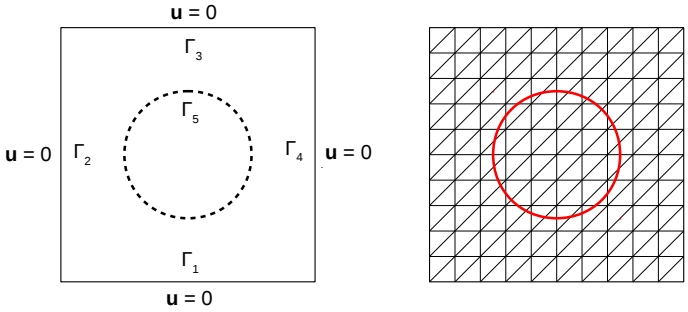


Figure 2.4: Manufactured solution problem setting. An scheme of the continuous problem is shown at the left and the discrete problem is shown at the right.

boundary conditions are defined as

$$\begin{aligned} \boldsymbol{u}(\boldsymbol{x}) &= 0, && && \boldsymbol{x} \in \Gamma_{1,2,3,4}, \\ \boldsymbol{u}(\boldsymbol{x}) &= \begin{bmatrix} 2x_1^2 x_2 (x_1-1)^2 (x_2-1)(2x_2-1) \\ -2x_2^2 x_1 (x_1-1)(x_2-1)^2 (2x_1-1) \end{bmatrix} && \text{m/s}, && \boldsymbol{x} \in \Gamma_5. \end{aligned}$$

The complete problem setting —including the boundary conditions— is shown in Figure 2.4.

Multiple parameters of the weak imposition of the boundary conditions and the stabilization of the cut cells are evaluated as follows. Specifically, the Nitsche method is tested using α and $\alpha\beta^{-1}$. Also, the Ghost Penalty with Orthogonal Projections is applied by using the γ_1 and the $\gamma_1\beta^{-1}$ stabilization parameters. To analyse the accuracy and convergence of the proposed scheme, the L^2-relative error norm of the numerical solution $[\boldsymbol{u}_h, p_h]$ is calculated with respect to the exact solution $[\boldsymbol{u}_{ex}, p_{ex}]$. Two type of relative error norms are calculated: the first is the L^2-relative error norm over the immersed

boundary (Γ_5), which is given by

$$e_{\boldsymbol{u},L^2(\Gamma)} = \frac{\|\boldsymbol{u}_{ex} - \boldsymbol{u}_h\|_{L^2(\Gamma)}}{\|\boldsymbol{u}_{ex}\|_{L^2(\Gamma)}}, \quad e_{p,L^2(\Gamma)} = \frac{\|p_{ex} - p_h\|_{L^2(\Gamma)}}{\|p_{ex}\|_{L^2(\Gamma)}},$$

$$e_{\nabla\boldsymbol{u},L^2(\Gamma)} = \frac{\|\nabla(\boldsymbol{u}_{ex} - \boldsymbol{u}_h)\cdot\boldsymbol{n}\|_{L^2(\Gamma)}}{\|\nabla\boldsymbol{u}_{ex}\|_{L^2(\Gamma)}}.$$

Here $\|\nabla\boldsymbol{u}_{ex}\|_{L^2(\Gamma)}$ is the Frobenius norm of the velocity gradient at the boundary. The relative error norm of the velocity gradient $e_{\nabla\boldsymbol{u},L^2(\Gamma)}$ is the most sensitive quantity to identify the instabilities at the cut elements, i.e., a great value is directly related to an instability occurrence at $\boldsymbol{T}_h^c$, even if other measured errors are low. The second type of error norm is the L^2-relative error norm of the numerical solution over the domain Ω, which for each variable is computed as

$$e_{\boldsymbol{u},L^2(\Omega)} = \frac{\|\boldsymbol{u}_{ex} - \boldsymbol{u}_h\|_{L^2(\Omega)}}{\|\boldsymbol{u}_{ex}\|_{L^2(\Omega)}}, \quad e_{p,L^2(\Omega)} = \frac{\|p_{ex} - p_h\|_{L^2(\Omega)}}{\|p_{ex}\|_{L^2(\Omega)}},$$

$$e_{\nabla\boldsymbol{u},L^2(\Omega)} = \frac{\|\nabla(\boldsymbol{u}_{ex} - \boldsymbol{u}_h)\|_{L^2(\Omega)}}{\|\nabla\boldsymbol{u}_{ex}\|_{L^2(\Omega)}}.$$

2.3.1.1 Mesh convergence

To measure the impact of the stabilization methods on the solution, the following scenarios are considered:

- The Nitsche method.
- The Nitsche method using $\alpha\beta^{-1}$.
- The Nitsche method and the Ghost Penalty with Orthogonal Projections.
- The Nitsche method and the Ghost Penalty with Orthogonal Projections using $\alpha\beta^{-1}$ and $\gamma_1\beta^{-1}$ respectively.

In all cases, the Nitsche parameters are set to $s_1 = 50$ and $s_2 = 1.0$, and the Ghost Penalty parameters to $s_G = 0.01$ and $s_u = 0.1$ and $s_p = 1.0$. These parameters have been defined in a similar way to those of [119] and have been corroborated with preliminary tests that have demonstrated numerical convergences.

The relative error norms are computed for the numerical results using different meshes with element sizes varying from $0.01 < h < 0.1$ m. The instabilities at the cut elements are present when at least one element of the interface has $\beta < 10^{-6}$. This is explained since instabilities are triggered by small ratios of β. In order to obtain reference solutions, values of $\beta > 10^{-3}$ are also tested using meshes with a similar number of elements as the

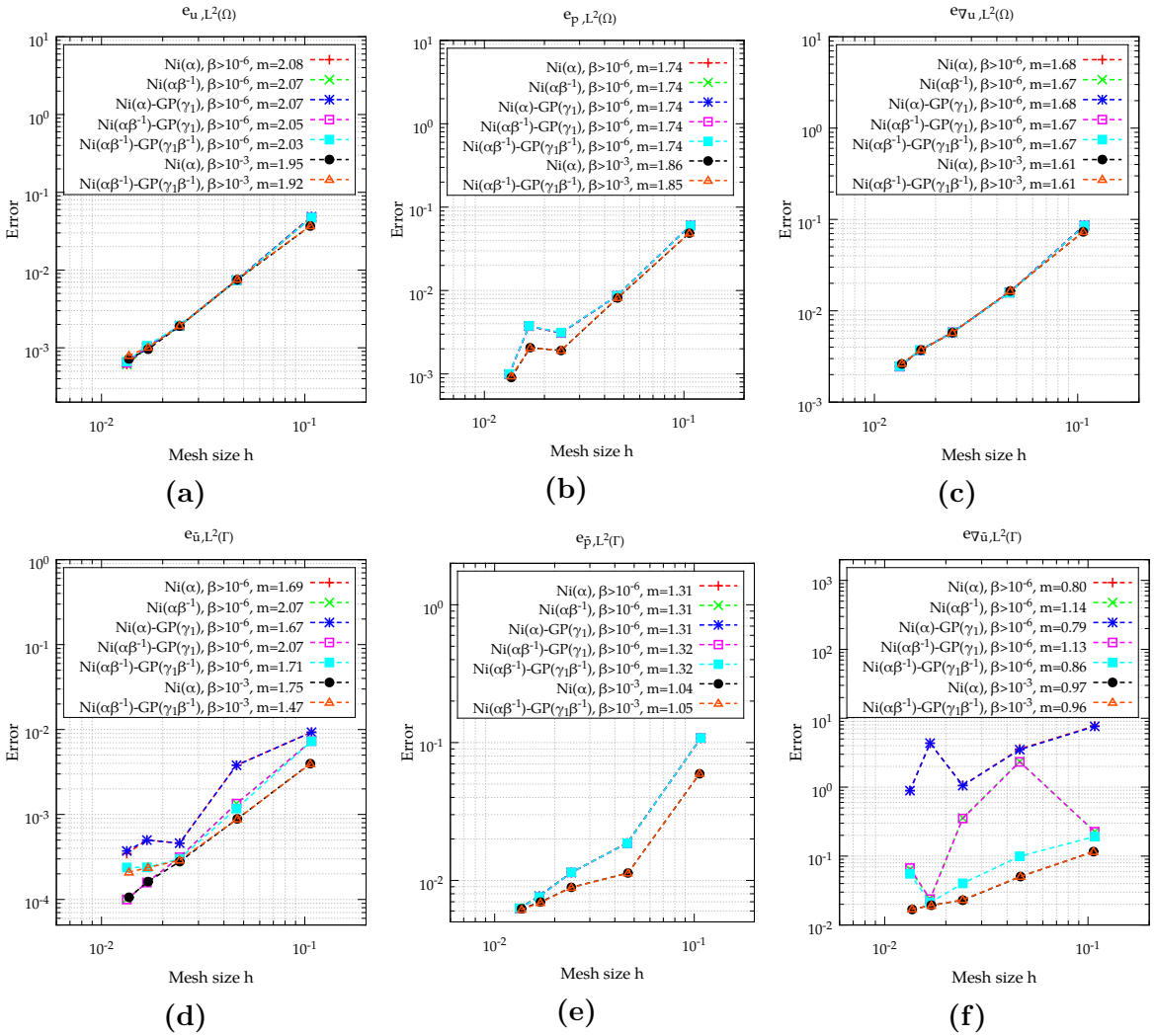


Figure 2.5: Mesh convergence results. Domain relative errors for: (a) velocity, (b) pressure, and (c) velocity gradient. Boundary relative errors for: (d) velocity, (e) pressure, and (f) velocity gradient.

$\beta < 10^{-6}$ meshes, but which guarantee that instabilities are controlled or negligible. To this end, the size of the elements is modified so that the desired β is reached in all meshes.

The mesh convergence results for the different formulations are presented in Figure 2.5. Both the accuracy and the slope m of each stabilized formulation are shown in this figure. In the case of the domain relative errors, it is observed a homogeneous behavior for all formulations: there is an optimal convergence of $e_{\boldsymbol{u},L^2(\Omega)}$ with a slope close to $m \approx 2$. Similarly, $e_{\nabla p,L^2(\Omega)}$ and $e_{\nabla \boldsymbol{u},L^2(\Omega)}$ demonstrate accurate results in all scenarios, with a sub-optimal convergence. Reference solutions with $\beta > 10^{-3}$ meshes show slightly better pressure accuracy, but the slopes are similar in all cases.

On the contrary, diverse results are depicted from the boundary error measure. It can be observed that the Nitsche method alone give inaccurate results: the poorest $e_{\boldsymbol{u},L^2(\Gamma)}$ accuracy is obtained for the Nitsche method with the Ghost Penalty. An asymptotic behavior of the error convergence can also be appreciated when activating the GP using $\gamma_1\beta^{-1}$ parameter. This can be explained because of the increase in bad cuts in the interface

zone when refining the mesh. Remarkably, a great improve in the accuracy of the method is obtained when the $\alpha\beta^{-1}$ parameter is used in the Nitsche scheme.

In the case of the $e_{p,L^2(\Gamma)}$ pressure results, those are similar for all methods. This can be explained since the β parameter does not affects most terms of (2.16) or (2.18). Special attention must be drawn to the $e_{\nabla \boldsymbol{u},L^2(\Gamma)}$ results, for which is evident the positive effect of the $\alpha\beta^{-1}$ and $\gamma_1\beta^{-1}$ parameters: the proposed stabilization scheme show accurate results for all mesh sizes. It also exhibits an asymptotic behavior at fine meshes, where the error can be mostly associated to the Nitsche method. Hence, one conclusion is that the proposed method can affect the accuracy of the solution at the interface, but it keeps control over $e_{\nabla \boldsymbol{u},L^2(\Gamma)}$ for all cut elements. However, since β can be close to zero, the $\alpha\beta^{-}1$ penalty parameter can go to infinity. This will create a severe ill conditioning of the discrete problem. Therefore, if this occur the solution will diverge. In other words, β can generate a problem of indetermination, but when this does not happen, the use of this ratio can improve the behavior of the method.

In order to further investigate this result near the interface, the velocity field and the velocity gradient magnitude are presented in Figure 2.6. Two different meshes are used to obtain these results: one with an average element size of $h = 1/22$ m, and another with $h = 1/22.01$ m. The results at the top of the figure are obtained using the $h = 1/22.01$ m mesh, while the ones at bottom are obtained by using the $h = 1/22$ m size mesh. Both meshes have $\beta < 10^{-6}$ elements at the interface and the boundary condition is imposed weakly using the Nitsche technique.

The results at the left of Figure 2.6 are obtained with the Nitsche method and the $\alpha\beta^{-1}$ parameter. It is easy to observe sources of instability in the results using the $h = 1/22.01$ m mesh (top-left). There is an atypical velocity vector at the bottom of the interface for which the magnitude of velocity is approximately four times the average, and the error is around 216% if compared against the analytic velocity. In addition, the velocity gradient magnitude is about 25 times the average. The elements where this instability appears have the lowest β ratio of the interface elements. Some few elements near the top region of the interface also show large velocity gradients (almost 15 times larger than the average). On the contrary, the $h = 1/22$ m mesh results (bottom-left) are stable using the Nitsche method and $\alpha\beta^{-1}$ alone.

The results at the right side of Figure 2.6 are obtained using the Nitsche and Ghost Penalty methods with the $\alpha\beta^{-1}$ and $\gamma_1\beta^{-1}$ parameters, respectively. When the Ghost Penalty using $\gamma_1\beta^{-1}$ is activated, is possible to see that these instabilities are reduced: both the magnitude and direction of the velocity are controlled. In the $h = 1/22.01$ m mesh (top-right) case, the relative velocity error of the maximum velocity is less than 7.5%. Also, the obtained velocity solution for the $h = 1/22$ m mesh (bottom-right) is kept under control but the accuracy is slightly affected in contrast to the Nitsche $\alpha\beta^{-1}$ method alone (bottom-left).

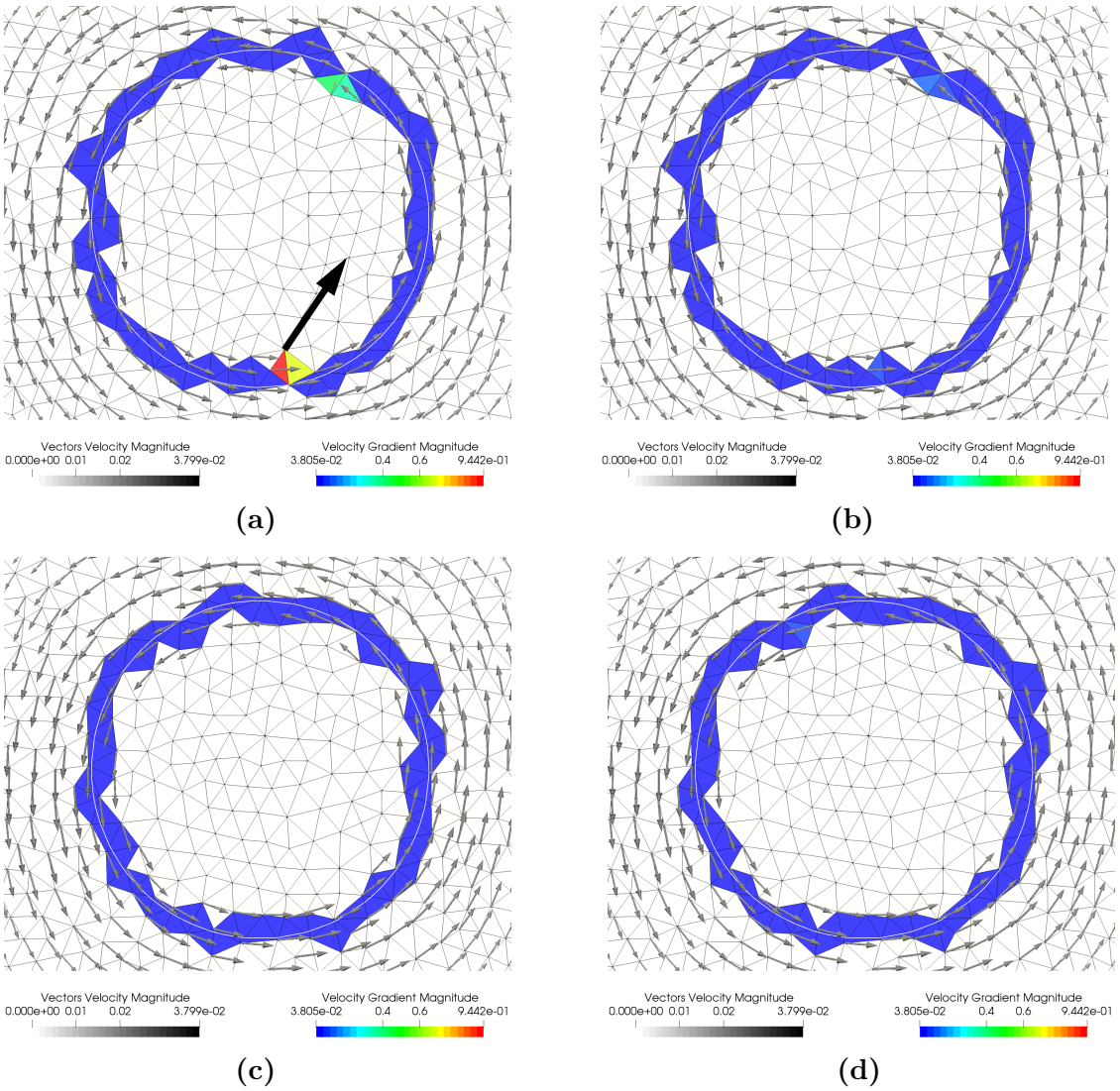


Figure 2.6: Velocity field and velocity gradient norm results. (Top) $h = 1/22.01$ m size mesh and (Bottom) $h = 1/22$ m size mesh. (Left) Nitsche method with the $\alpha\beta^{-1}$ parameter, (right) Nitsche and Ghost Penalty methods with the $\alpha\beta^{-1}$ and $\gamma_1\beta^{-1}$ parameters.

2.3.1.2 Algorithmic parameters

Next, the values of the algorithmic parameters are investigated. This analysis is done to identify the ranges where the method achieves best performance. The global ASGS stabilization is set including the Nitsche weak imposition method including $\alpha\beta^{-1}$ and the Ghost Penalty stabilization with $\gamma_1\beta^{-1}$. Four different meshes are tested ensuring that at least one element in the interface accomplishes $\beta < 10^{-6}$. These three meshes have average element sizes of $h_1 = 1/10$ m, $h_2 = 1/22$ m, and $h_3 = 1/60$ m. The kinematic viscosity is set to $\nu = 0.01$ m^2/s.

The first analysis is achieved for the Nitsche parameter s_1 in a range between 5 and 1000, and fixing $s_2 = 1.0$, $\sigma_G = 0.01$, and $s_u = 0.1$. The relative error results are shown in Figure 2.7 for the different values of the Nitsche parameter and the mesh sizes.

The $L^2(\Omega)$ relative error norm is almost constant for all values of s_1. However, at the interface is observed a reduction of the velocity error $e_{\boldsymbol{u},L^2(\Gamma)}$ with the increase of s_1. This can be explained since the weight of the first term of equation (2.15) is directly related to this parameter, imposing the $\boldsymbol{u}_d$ value at the interface. Yet, this reduction of the velocity error at the interface is compensated with the raise in the velocity gradient error $e_{\nabla\boldsymbol{u},L^2(\Gamma)}$ for which a high value of s_1 increases the error. Also, small values of s_1 affects the $e_{\nabla\boldsymbol{u},L^2(\Gamma)}$ accuracy, and therefore, optimal values for this Nitsche parameter are located inside a narrow range. The Nitsche method has therefore the disadvantage of defining *a-priori* the algorithmic parameters. In the manufactured solution problem, the optimal s_1 value is found to be close to 50.

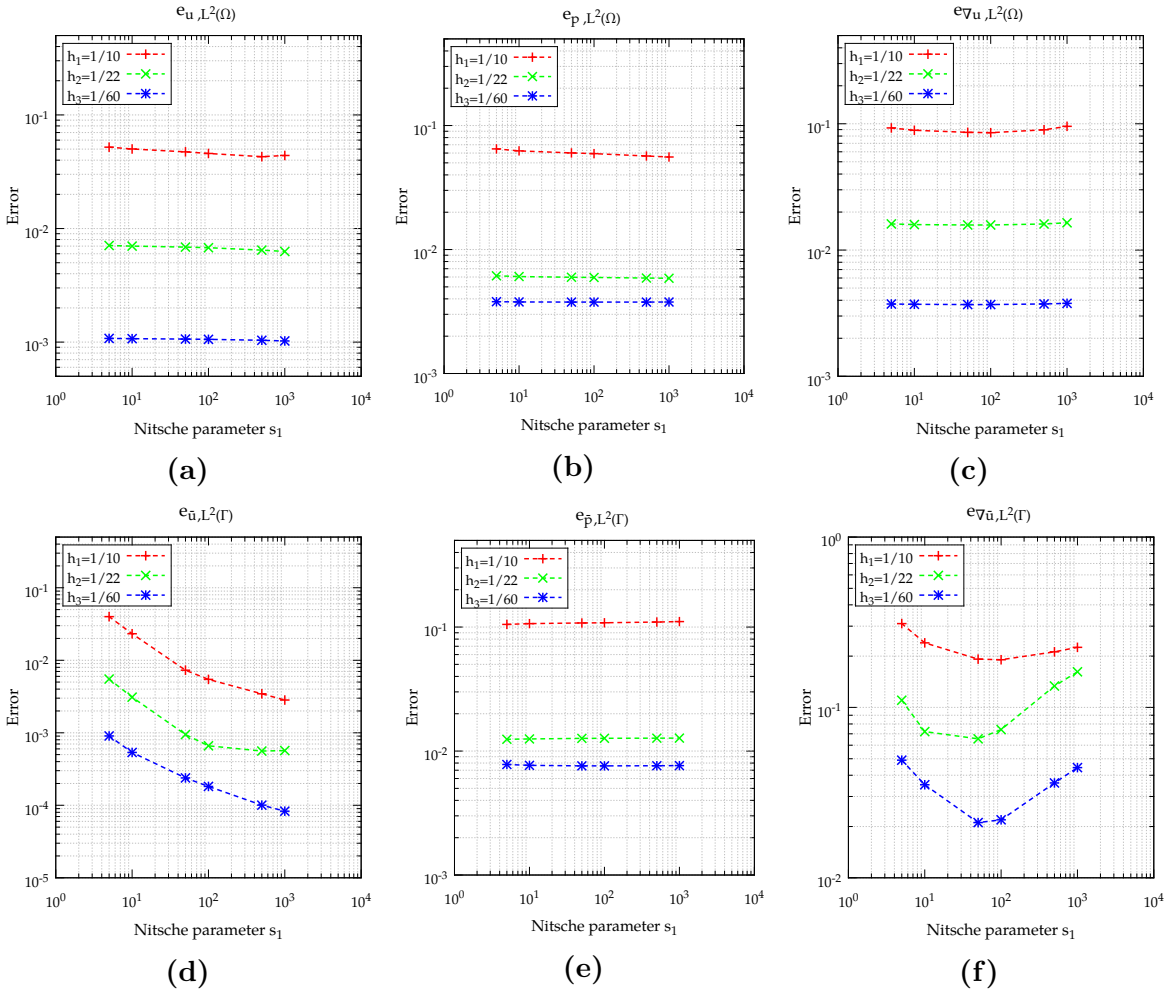


Figure 2.7: Relative error norms for different s_1 values and mesh sizes. The results are obtained by fixing $s_2 = 1.0$, $\sigma_G = 0.01$, and $s_u = 0.1$.

Finally, the Ghost Penalty parameters are evaluated by shifting s_u between 0.001 and 1.0, and setting $s_2 = 50$ and $\sigma_G = 0.01$. The error results of varying the s_u parameter are shown in Figure 2.8 for the different meshes. Unlike the previous results, both L^2 relative errors measured in the domain and over the interface seem to be affected by the definition of the s_u value. Regarding the velocity gradient error $e_{\nabla\boldsymbol{u},L^2(\Gamma)}$, it significantly depends over the mesh size. It is observed that result for the two meshes with similar element sizes ($h_2 = 1/22$ m and $h_3 = 1/22.01$ m) and $\beta < 10^{-6}$, behave oppositely. It is

possible to see that $e_{\nabla \boldsymbol{u}, L^2(\Gamma)}$ of $h_3 = 1/22.01$ m increases for small values of s_u. Hence, $s_u > 0.1$ is necessary to obtain accurate results, but also $s_u > 0.5$ so that the overall accuracy is not penalized when using fine meshes. These results confirm the relevance of the Ghost Penalty in the velocity control. However, the small range of values of s_u that can be chosen leads to a careful selection.

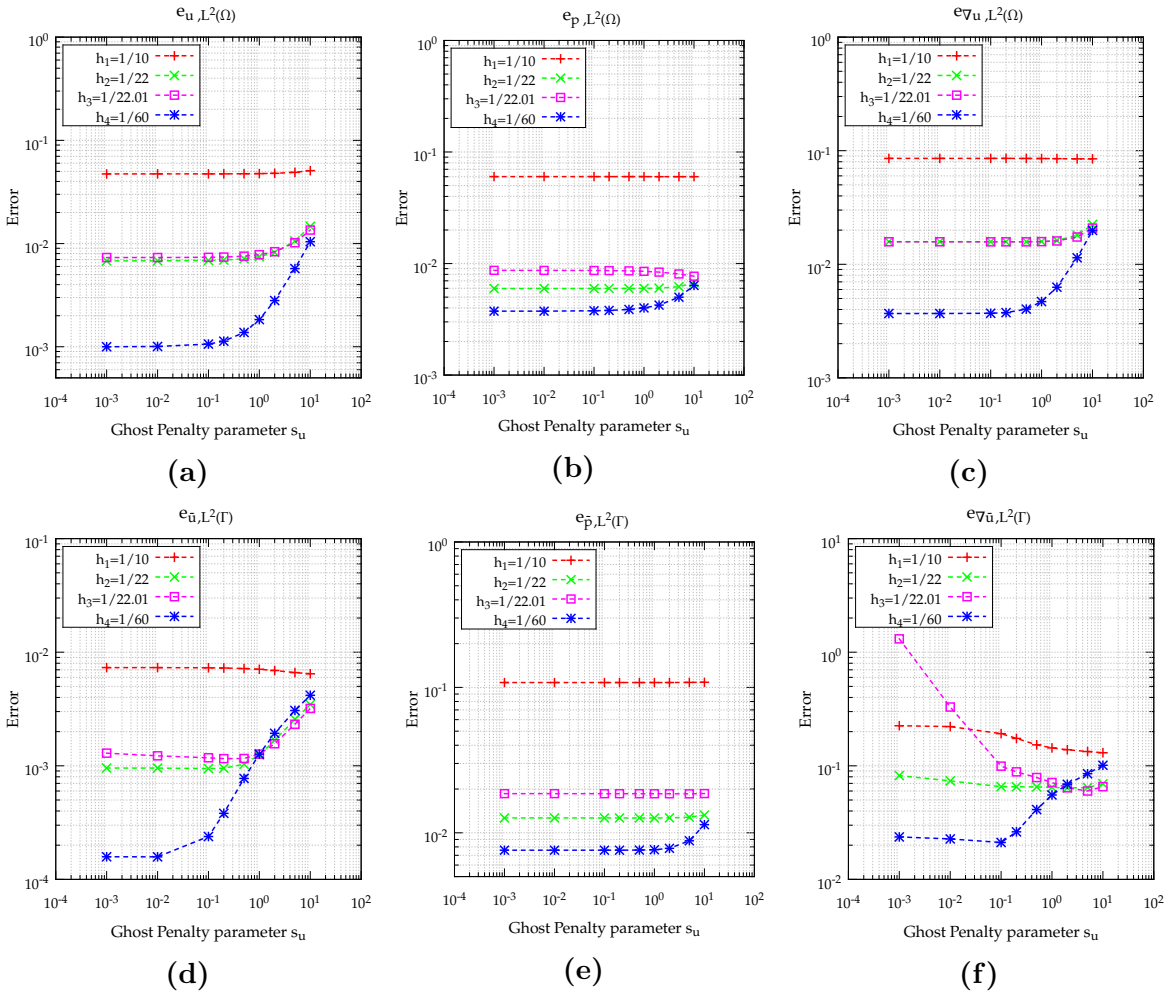


Figure 2.8: Relative error norms for different s_u values and mesh sizes. The results are obtained by fixing $s_1 = 1.0$, $s_2 = 1.0$, and $s_G = 0.01$.

2.3.2 Flow past a cylinder

Now, the flow past a cylinder at low *Reynolds* numbers is analyzed, which is a commonly known benchmark for computational fluid dynamics solvers. This problem models the flow past an infinitely long cylinder that can be represented as a two-dimensional flow. The computational domain is defined as a rectangular domain $\Omega = [0, L] \times [-H/2, H/2] = [0, 2.2]$ m $\times [0, 0.41]$ m with the definition of the cylinder as a circle cut of diameter $d = 0.1$ m located at $[0.2, 0.2]$ m, as schematically shown in Figure 2.9. The fluid density and kinematic viscosity are set to be $\rho = 1.0$ kg/m^3 and $\nu = 0.001$ m^2/s, respectively. As the boundary conditions of the problem, one sets no-slip boundary conditions for the upper

and lower walls, as well as for the cylinder surface. In the case of the inlet wall (left-most wall), we impose a parabolic profile for the horizontal component of velocity:

$$u_1(0, x_2) = 4U_{max}\left(\frac{x_2(0.41 - x_2)}{0.41^2}\right),$$

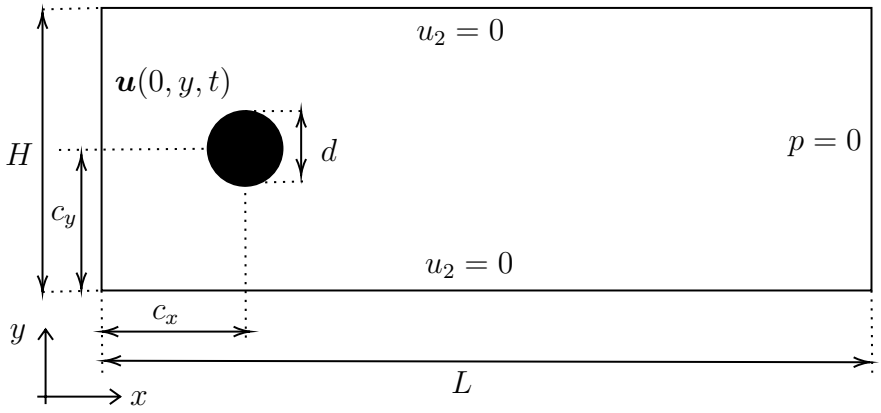


Figure 2.9: Flow past a cylinder problem scheme.

where, U_{max} is the maximum velocity of the inlet. Additionally, at the inlet face we impose a null value for the vertical component of velocity. In the case of the outlet boundary, tractions are prescribed as $\boldsymbol{\sigma} \cdot \boldsymbol{n} = 0$, which is the free boundary condition. The *Reynolds* number is calculated based on the cylinder diameter, the kinematic viscosity, and the mean inlet velocity $U_{mean} = U_{max}/1.5$. The problem is tested for both $Re = 20$ and $Re = 100$. The former leads to a steady flow, while the later case is periodically unsteady. The numerical results can be compared against reported results in the literature [121].

Indeed, to make a quantitative comparison of the results, some integral values of the flow field are calculated: some of the most comparable quantities are the integral value of the lift and drag coefficients over the cylinder surface, which is calculated using the solution of the velocity field, and compared against benchmarked results. Firstly, forces that the fluid exerts over the cylinder surface Γ in each spatial direction are computed. To do that, the force $\boldsymbol{F}$ over the immersed boundary Γ by the surface integral is calculated as follows:

$$\boldsymbol{F} = \int_{\Gamma} \boldsymbol{\sigma} \cdot \boldsymbol{n}\, d\Gamma. \tag{2.20}$$

To calculate the lift (C_L) and drag (C_D) coefficients, the vertical and horizontal components of the force are split and scaled by the dynamic pressure $\frac{1}{2}\rho {U_{mean}}^2 D$.

2.3.2.1 Laminar and steady *Re=20* case

To evaluate the flow around cylinder for a *Reynolds* number equal to 20, the max velocity is set to $U_{max} = 0.3$ m/s. To further test this assessment of the numerical methods for stabilizing the cut elements, the flow around a cylinder is solved in a mesh that has at

least one element with a cut ratio $\beta < 10^{-6}$, such that instabilities at the interface are induced.

In that order, four different cases are studied: the first one is the conforming mesh solution of the problem. This case is intended to obtain a reference result, for which the cylinder boundary exists as a real boundary of the fluid mesh and the Dirichlet conditions are strongly imposed. The other cases are solved using the NCMM approach developed in this work, defining the immersed cylinder by the interface and weakly imposing the boundary conditions by the Nitsche method. In this sense, the second case is a baseline case for the NCMM approach since it is obtained with a mesh in which all elements at the interface fulfill $\beta < 10^{-3}$, seeking to avoid the instabilities of bad cuts. Any cut stabilization methods in the baseline case is neglected, such that the reference solution for the weak imposition is obtained. The third and fourth cases use a similar mesh to the second case (same number of elements), but with a $\beta < 10^{-6}$. The third case is solved with Nitsche $\alpha\beta^{-1}$, and the fourth combines Nitsche $\alpha\beta^{-1}$ and Ghost Penalty γ_1/β. For all the NCMM cases the Nitsche parameters are set to $s_1 = 110$ and $s_2 = 1$. In the fourth case, the Ghost Penalty parameters are set to $s_G = 0.01$ and $s_u = 0.1$.

The results for the drag and lift coefficients are presented in Table 2.1. Also, the pressure difference $\Delta p = p(\boldsymbol{x}_1) - p(\boldsymbol{x}_2)$ between the points $\boldsymbol{x}_1 = (0.15, 0.2)$ m and $\boldsymbol{x}_2 = (0.25, 0.2)$ m on the front and rear side of the cylinder is given in this table. The conforming mesh results agree to the reported benchmark results. However, the C_L results between those deviate. This deviation is explained from the numerical precision of the method, which is restricted to the use of linear triangle elements in the mesh. For the NCMM cases, it is observed a lower accuracy of the methods in contrast to the CMM, but the results are still close to and consistent with the benchmark values. However, the accuracy of the NCMM results are limited in the refinement by the asymptotic behavior that we described before in Section 2.3.1. As a remark, the solution diverges when the Nitsche method without cut stabilization is attempted using the $\beta < 10^{-6}$ mesh.

Case	**DOF**	C_D	C_L	Δp
Reference [121]	51159	5.5567	0.0106	0.1172
Conformed mesh	49731	5.5118	0.0150	0.1192
NCM - Nitsche (α)	55326	5.4625	0.0136	0.1148
NCM - Nitsche ($\alpha\beta^{-1}$)	52433	5.4618	0.0155	0.1147
NCM - Nitsche ($\alpha\beta^{-1}$) - GP ($\gamma_1\beta^{-1}$)	52433	5.4570	0.0156	0.1142

Table 2.1: Results for the flow past an infinite cylinder at $Re = 20$.

2.3.2.2 Laminar and transient *Re=100* case

The flow past a cylinder for a $Re = 100$ generates swirling vortices due to the unsteady flow separation of the fluid around the cylinder. The same fluid properties and boundary conditions than the $Re = 20$ case are applied, except for the $U_{max} = 1.5$ m/s velocity inlet. To capture this dynamic phenomena, a transient simulation using the second order

backward difference (BDF2) method for the time integration is performed. Simulations are run with a time step equal to $\delta t = 0.01$ s, for a total simulation time of $T = 10$ s, so that the flow develops completely. The C_D, C_L, and Δp integral quantities are measured at the run. Additionally, the f frequency for C_L is computed, such that the dimensionless *Strouhal* number can be calculated as $St = fD/U_{mean}$. The numerical results are compared against the reported benchmarks in [121], which were obtained using conforming meshes of different sizes, and a time step equal to $\delta t = 0.01$ s.

For this transient case, a mesh with fewer elements than the $Re = 20$ case is used. Nevertheless, the mesh fulfills a $\beta < 10^{-6}$ for at least one element at the interface. The Nitsche $\alpha\beta^{-1}$ method with Ghost Penalty $\gamma_1\beta^{-1}$ are tested. The Nitsche parameters are set to $s_1 = 50$ and $s_2 = 1$, and the Ghost Penalty parameters are applied as $s_G = 0.01$ and $s_u = 0.01$, that have been defined after previous tests for this case. The decrease of the Nitsche and GP parameters is explained as the gradients and instabilities of the numerical velocity increase with Re. Hence, the Nitsche parameters need to be decreased with respect to the $Re = 20$ case to improve the non-linear convergence of the method.

In Table 2.2 the maximum, minimum, and mean values for the C_D and C_L are presented. The frequency $1/f$ and the *Strouhal* number St values are displayed in that table. Moreover, the time-dependent results for C_D, C_L, and Δp are shown in Figure 2.10. As mentioned above, the NCMM results obtained in this work are consistent with the CMM references cases. Despite the coarse mesh used, the post-processed integral quantities agree well with the coarse reference case. Also, the pressure contour and velocity streamlines are shown in Figure 2.11, which demonstrate a consistent generation of *von Kármán* vortex over a complete *Strouhal* period T_S. However, it is noted that Δp improves with finer meshes. Likewise, a residual instability can be observed in badly cut elements at the interface, even though the Ghost Penalty method controls the velocity solution at the interface. This effect is not captured in the calculation of the forces.

		C_D			C_L				
Case	**DOF**	**min**	**max**	**mean**	**min**	**max**	**mean**	$1/f$	St
CMM [121]	42016	3.1338	3.1964	3.1651	-1.0155	0.9838	-0.0158	0.3400	0.2941
CMM [121]	167232	3.1497	3.2124	3.1811	-1.0169	0.9859	-0.0155	0.3300	0.3030
NCMM	21426	3.1335	3.1940	3.1649	-0.9845	0.9158	-0.0677	0.3400	0.2941

Table 2.2: Integral results for the flow past an infinite cylinder at $Re = 100$.

2.3.3 Flow past rigid obstacles with predefined motion

The final three cases that are simulated with methods developed in the present work are: a rising wall obstacle in a pipe flow, a closing throat in a pipe, and a rotating flat body in a pipe. These cases comprise interactions between the fluid flow and a solid obstacle with a predefined motion. Most are original simulations of engineering applications to be first published here. In a strict sense, these problems cannot be considered as 2-way

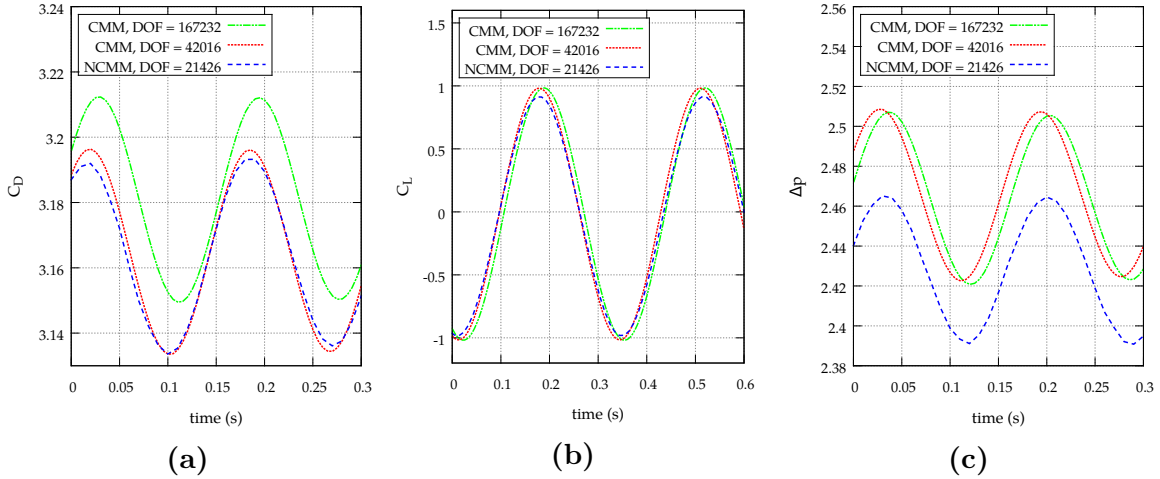


Figure 2.10: Time-dependent results of the flow past an infinite cylinder at *Re=100*. (a) Drag coefficient, (b) lift coefficient, and (c) pressure difference.

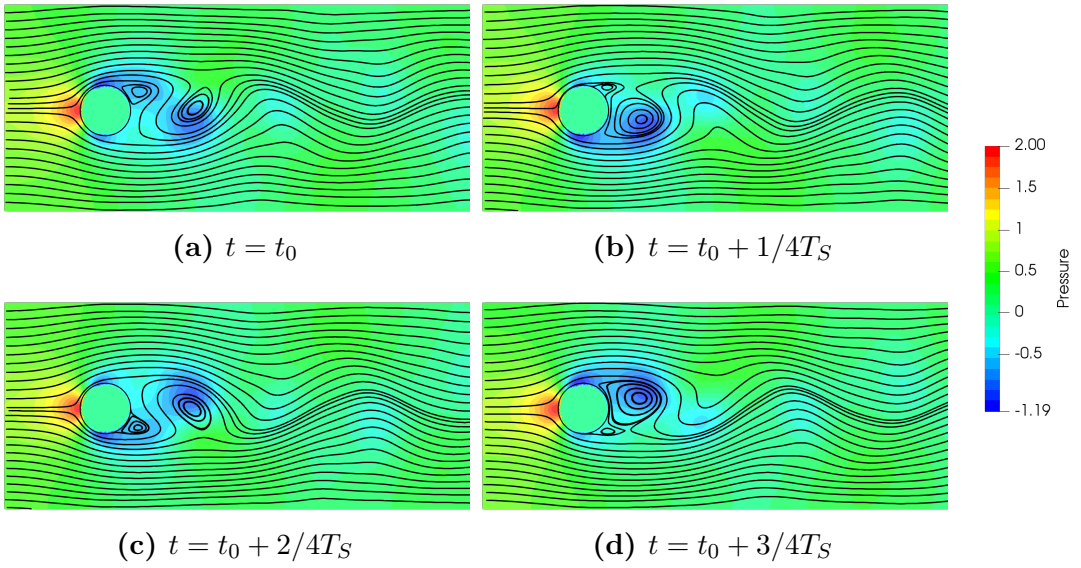


Figure 2.11: Pressure contour and velocity streamlines at different instants of the simulation for the flow past an infinite cylinder at $Re = 100$.

fluid-structure interaction cases because the obstacle movement is known and prescribed *a-priori*. Certainly, the solid deformation is neglected and so, its movement is addressed as a rigid body motion. The interest to solve these problems lies on the dynamic change of the cut elements in the fluid's mesh: the moving obstacle generate cut elements with small β ratios and instabilities are triggered continuously. To overcome the lack of information of the uncovered nodes at each time step, the inverse distance weighting method to interpolate the unknown in those nodes is used.

2.3.3.1 Rising gate in pipe flow

The effect of a single closing gate inside a pipe moving from one wall is first considered. This problem is represented as a two-dimensional flow in a rectangular domain $\Omega =$

$[0, L] \times [0, H] = [0, 4.0]$ m $\times [0, 0.8]$ m. The left-most side of the channel is the inlet boundary, while the right-most side is the outflow. The rising gate is modelled as a chamfered rectangle wall of width $W = 0.15$ m and chamfer radius equal to half the width. The center-line of the rectangle is located at $c_x = 1.0$ m from the inlet (see Figure 2.12). As boundary conditions of the fluid flow, it is imposed a constant velocity $u_1 = 1.0$ m/s at the inlet boundary, a null manometric pressure at the outflow, and no-slip boundary conditions on the bottom, top, and obstacle's walls. The density and dynamic viscosity of the fluid are set to $\rho = 1.0$ kg/m^3 and $\mu = 0.01$ kg/(m·s), respectively. The height h of the gate (as measured from the bottom of the domain) is a linear function expressed as:

$$h(t) = h_{initial} + \frac{t(H - h_{initial})}{t_c} \tag{2.21}$$

where t_c is the closing time and $h_{initial}$ is the initial height of the gate.

The exerted force of the fluid over the gate is calculated, and the pressure difference upwind and downwind the gate. Specifically, the pressure difference $\Delta p = p(\boldsymbol{x}_1) - p(\boldsymbol{x}_2)$ between $\boldsymbol{x}_1 = (0.0, 0.4)$ m and $\boldsymbol{x}_2 = (4.0, 0.4)$ m is measured. The Nitsche parameters are set equal to $s_1 = 500$ and $s_2 = 500$, while the cut stabilization parameters are fixed to $s_G = 0.1$, $s_u = 0.2$, and $s_p = 1.0$. A closing time of $t_c = 2$ s and a time step $\delta t = 0.01$ s are used, where the initial height of the gate is $h_{initial} = 0.1H$. The fluid mesh is composed of linear triangles and $DOF = 17847$. A refinement of the mesh is provided along the wall moving region. Also, finer elements are granted near constriction at the upper wall in order to accurately solve the re-circulation pattern of the flow.

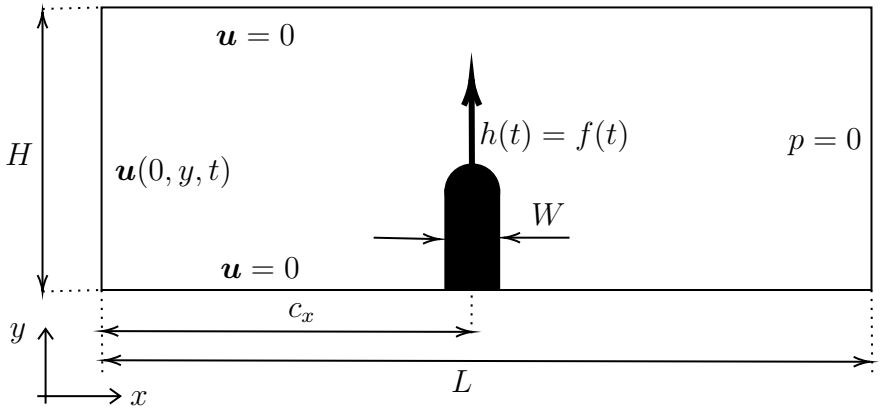


Figure 2.12: Rising gate in pipe flow problem scheme.

The pressure contours and velocity streamlines are presented in Figure 2.13 for different instants of the gate displacement, where numerical results at the time instants corresponding to $h(t)/H = \{0.3, 0.6, 0.9\}$ can be observed. From these results it can be noticed that the pressure increases as the gate closes the channel, with the highest values located upstream. Additionally, a re-circulation flow pattern is created downstream. This re-circulation zone is detailed in Figure 2.14(a), where the horizontal component of velocity is plotted against the vertical dimension at $x_1 = \{1.5, 2.2\}$ m positions. These profiles are plotted corresponding to the same time instants than those of Figure 2.13. Results of the exerted force over the rising gate and the pressure difference are presented

in Figure 2.14(b). Both the forces and pressure difference increase as the gate closes the channel. The plot is restricted to the time interval $(1.7 \leq t \leq 2.0)$ s to avoid a large increase of the results when the gate reaches the wall: the results for $t > 2.0$ s are omitted, since large fluctuations can be observed from $t = 1.98$ s when the gate begins to cover the adjacent elements to the top.

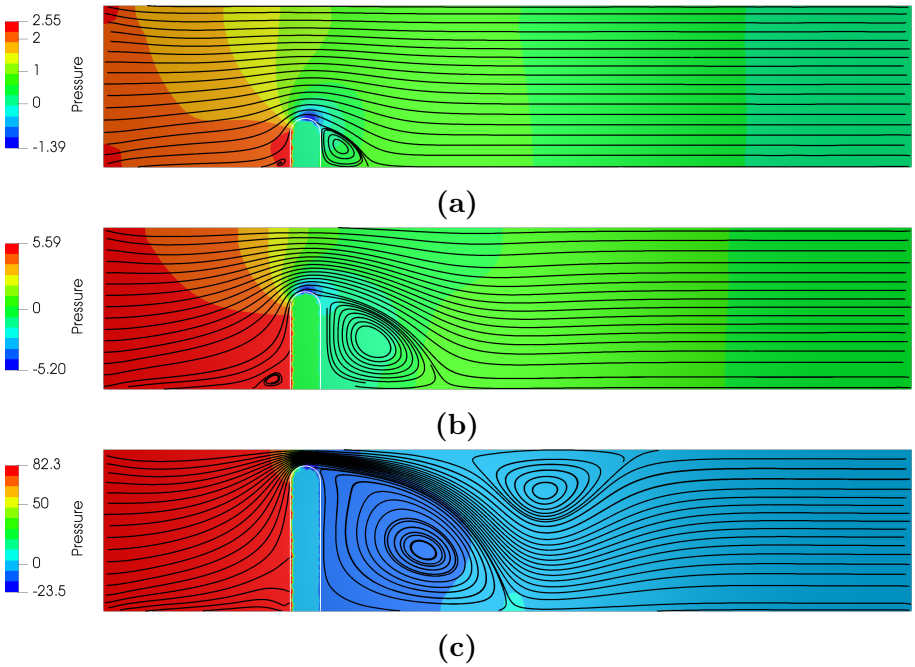


Figure 2.13: Contours of pressure and streamlines for the rising gate in pipe flow at heights: (a) $h/H = 0.30$, (b) $h/H = 0.60$, and (c) $h/H = 0.90$.

2.3.3.2 Closing throat in pipe flow (inlet hole, or venturi tube)

This case is similar to the previous one, but it includes a second obstacle closing the flow. A symmetric definition is set for the second closing gate: a falling obstacle from the top channel wall is defined with the same geometry of the rising obstacle. Also, its closing movement is prescribed to perform at the same speed of the rising obstacle but in the opposite direction. Besides from the no-slip conditions on the closing walls, the same domain and boundary conditions of the rising wall problem are prescribed. A schematic of the problem is depicted in Figure 2.15. The linear expression (2.21) to define the rising and falling movement of the obstacles is defined, but the total vertical distance is considered as being half the channel height. In the case of the falling obstacle, h_t is defined from the top wall. Again, a total closing time $t_c = 2$ s is defined and the initial heights of the two obstacles as $h_{initial} = 0.1H$.

The fluid force along the obstacles and the pressure difference between $\boldsymbol{x}_1 = (0.0, 0.4)$ m and $\boldsymbol{x}_2 = (4.0, 0.4)$ m are evaluated. The same mesh as the one used for the rising wall problem, but refined locally at the middle of the channel where both gates meet, is used. The Nitsche parameters are set to $s_1 = 500$ and $s_2 = 500$, while the cut stabilization parameters are $s_G = 0.1$, $s_u = 0.01$, and $s_p = 1.0$. The time step is $\delta t = 0.01$ s. The

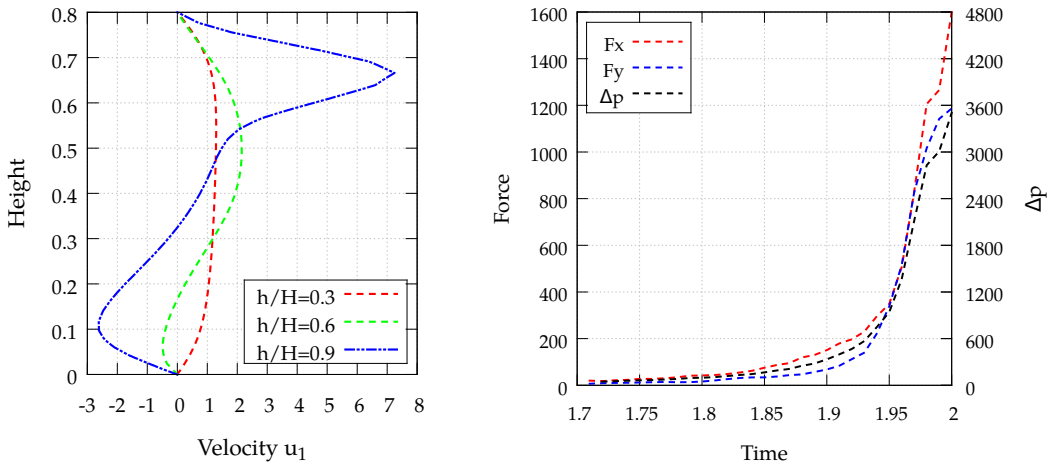


Figure 2.14: Transient results of the rising gate in pipe flow. (a) Horizontal velocity profiles at $x_1 = 1.5$ m and different time instants. (b) Exerted horizontal forces over the gate walls and pressure difference.

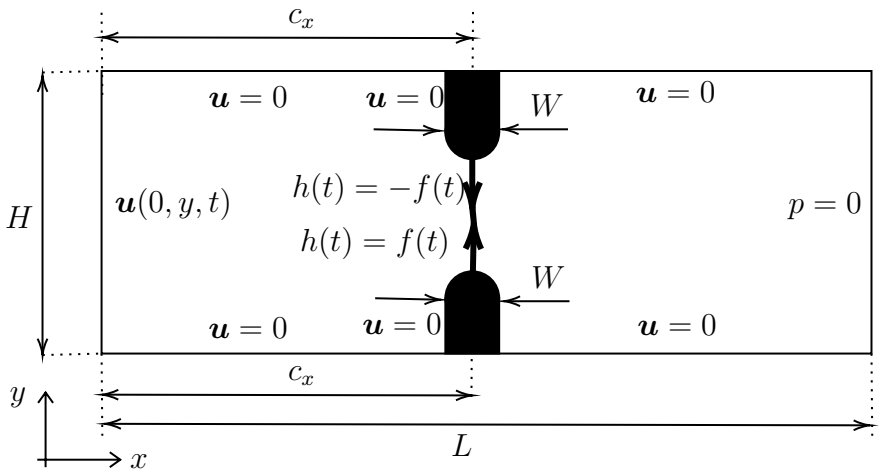


Figure 2.15: Closing throat in pipe flow problem scheme.

pressure contours and velocity streamlines are presented in Figure 2.16 for three time instants corresponding to $h(t)/H = \{0.15, 0.3, 045\}$. A similar trend to the rising wall problem is observed: the upstream pressure increases as the throat closes. Also, a double re-circulation pattern forms downstream of the throat. The re-circulation of the flow past the throat can also be noticed in Figure 2.17(a), where the horizontal velocity profile at $x_1 = 1.5$ m is plotted at the same three time instants commented before. In addition, Figure 2.17(b) shows the transient results of the exerted horizontal forces over the throat walls and the pressure difference upstream and downstream. The transient results are presented in the time interval $(1.7 \leq t \leq 2.0)$ s, since the method is restricted to the simulation before the two walls intersect. It is observed a slight difference between the exerted force at the upper and lower walls. This can be explained by the randomness of the cuts and the different β values of each interface. On the contrary, velocity profiles are axially symmetric and the solution is not affected by the mesh size.

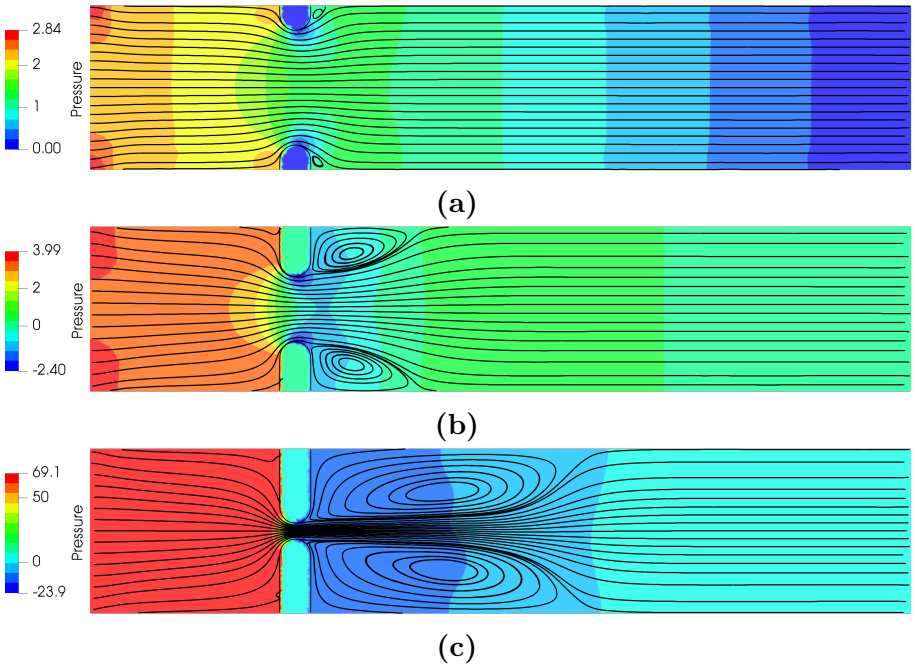


Figure 2.16: Contours of pressure and streamlines for the closing throat in pipe flow at heights: (a) $h/H = 0.15$, (b) $h/H = 0.30$, and (c) $h/H = 0.45$.

2.3.3.3 Rotating flat body in pipe flow (throttle body)

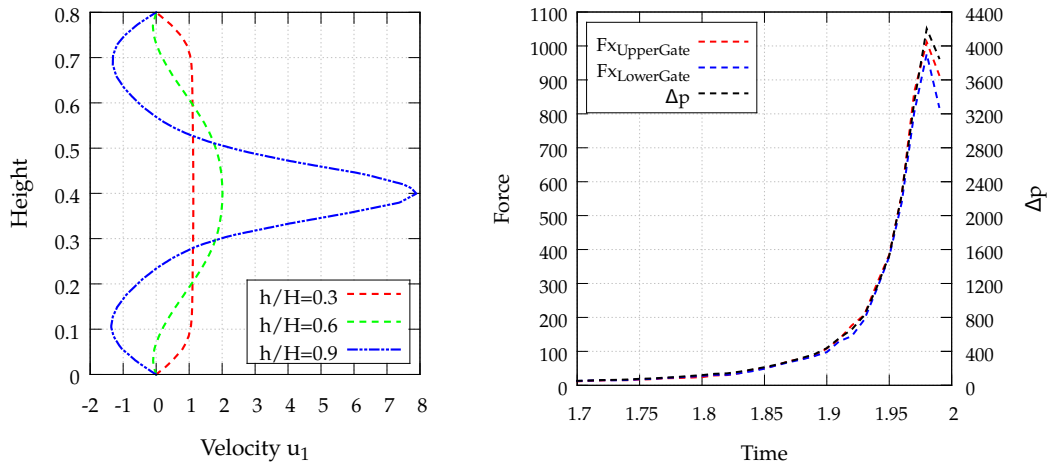
The last problem solved is a rotating flat obstacle inside a pipe. The shape of the obstacle makes this case similar to a throttle body constraining a fluid flow in a pipe. The flow problem is defined in the same two-dimensional rectangular domain as in the previous cases, but including a rotating rectangle of length $I = 0.95$ m and width $l = 0.1$ m with rounded edges of radius $l/2.0$. The center of the rectangle is located at $(c_x, H/2) = (1.0, 0.4)$ m. The center of rotation coincides with the center of the rectangle. A schematic of the problem is depicted in Figure 2.18. The inlet flow is defined at $(0, x_2)$ m, where a constant velocity profile $\boldsymbol{u}(0, x_2, t) = (1.0, 0.0)$ m/s is prescribed, while a null manometric pressure condition is set at the outlet (L, x_2) m. No-slip boundary conditions are imposed on the bottom and top walls of the fluid domain, as well as on the rotating obstacle walls.

The expression that defines the rotating manoeuvre of the body can be expressed as a linear function similar to equation (2.21), as

$$\theta(t) = \theta_{initial} + \frac{t(\pi/2 - \theta_{initial})}{t_c}, \tag{2.22}$$

where $\theta(t)$ is the angular position of the body (as measured with respect to the horizontal axis), $\theta_{initial}$ is the initial angular position of the body, and t_c is the time to complete a $\pi/2$ rotation.

The force exerted on the obstacle and the pressure difference at the same points than the previous examples are evaluated. The same Nitsche and cut stabilization parameters used for the case of the closing throat in pipe flow are applied here. A total rotating time



(a) **(b)**

Figure 2.17: Transient results of the closing throat in pipe flow. (a) Horizontal velocity profiles at $x_1 = 1.5$ m and different time instants. (b) Exerted horizontal forces over the throat walls and pressure difference.

$t_c = 2$ s is considered, where the initial position of the valve is $\theta_{initial} = 0$ which is parallel to the flow. The fluid mesh to be resolved has a total of $DOF = 24507$ and the time step is $\delta t = 0.01$ s. In this case, a local refinement of the mesh is provided inside the rotation area. The mesh is also refined downstream the obstacle and near the channel walls.

The pressure contours and the velocity streamlines are presented in Figure 2.19 for three time instants corresponding to $\theta(t) = \{\pi/12, \pi/4, 5\pi/125\}$. Again, the pressure increases as the gate closes the pipe flow, but re-circulation zones only appear in the last instants of the rotation. The horizontal velocity profile at $x_1 = 2.0$ m is reported in Figure 2.17(a) at the same three time instants. Also, Figure 2.17(b) demonstrates the transient results of the exerted force over the flat body and the pressure difference between $\boldsymbol{x}_1 = (0.0, 0.4)$ and $\boldsymbol{x}_2 = (4.0, 0.4)$. Above $t = 1.85$ s the pressure difference δp and exerted horizontal force F_x show a strong oscillating behavior that can be linked with the throttling and acceleration of the flow. These fluctuations are also related with instabilities triggered by badly cut elements which are not fully controlled by the cut stabilization technique.

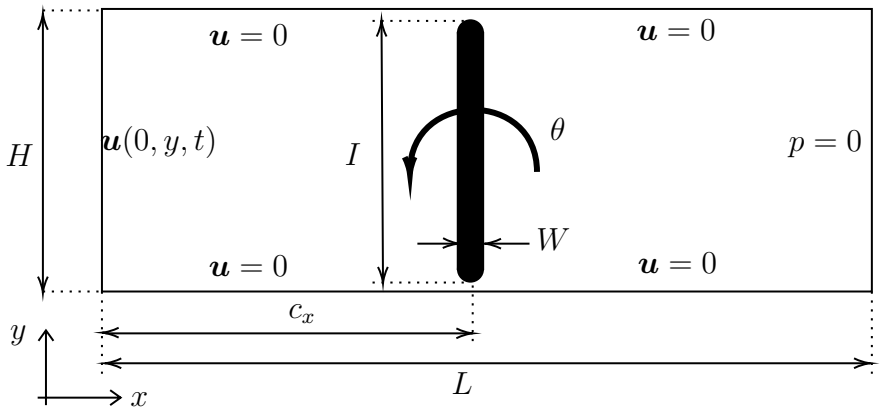


Figure 2.18: Rotating flat body in pipe flow problem scheme.

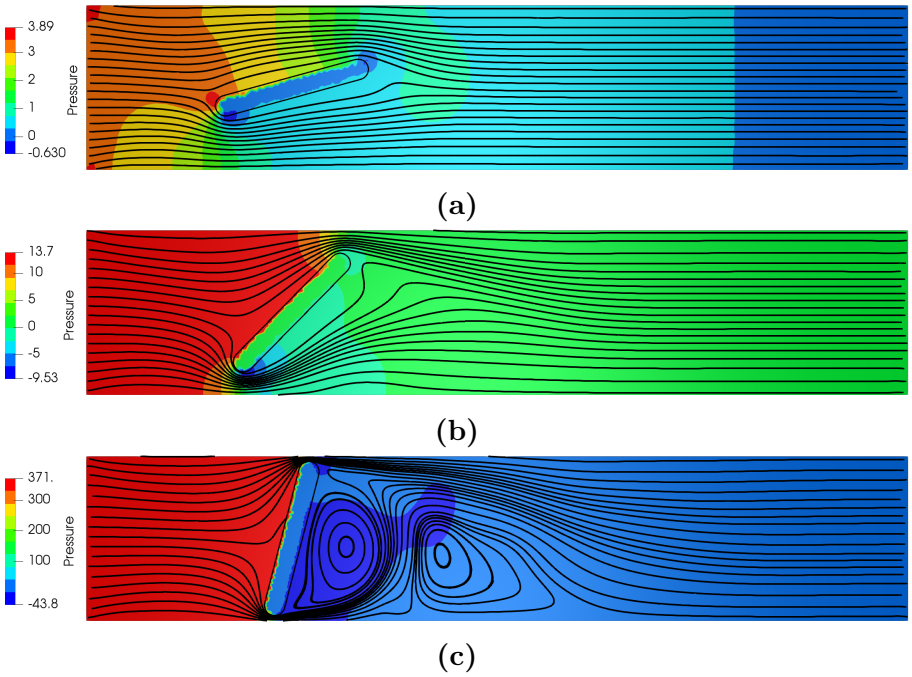
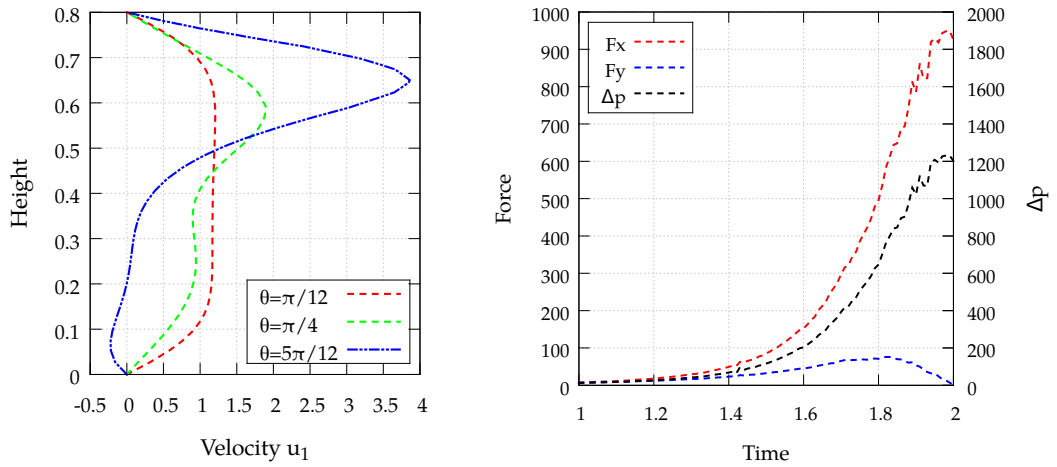


Figure 2.19: Contours of pressure and streamlines for the rotating flat body in pipe flow at angular positions: (a) $\theta = \pi/12$, (b) $\theta = \pi/4$, and (c) $\theta = 5\pi/12$.

2.4 Conclusions

In this work, a flow solver for fluid domains with immersed moving obstacles is developed. A VMS stabilized formulation that defines the sub-scales as both Algebraic (ASGS) and Orthogonal (OSGS) is applied to overcome the instability problems associated with the Galerkin formulation of the fluid flow equations. Weak prescription of no-slip conditions for the fluid velocity at the boundary of the immersed solid is addressed by means of the Nitsche method. The tracking of the immersed solid interface is established by using a sharp interface method, in which the Level-Set method is used for simply-shaped objects, while a mesh that overlaps the fluid mesh is used for complex geometries of the immersed obstacle. Also, the sub-triangle technique is implemented to perform the numerical integration of the cut elements. In order to overcome the instabilities generated by the bad cuts the Ghost Penalty with Orthogonal Projections is used. The numerical study of alternative definitions for the algorithmic parameters of the Nitsche and Ghost Penalty with Orthogonal Projections (GPOP) methods is a fundamental contribution of this study: the use of $\alpha\beta^{-1}$ and $\gamma_1\beta^{-1}$ parameters work as a cut cell sensor that highly improves the accuracy of the numerical method. However, a deeper analysis is needed to understand the lack of stability of GPOP itself and how the β parameter can be modified to eliminate the tendency for ill-conditioning of the system.

The numerical scheme has also demonstrated the ability to simulate complex flow problems, beginning with the transient flow past a cylinder for which accurate results have been obtained compared to the reference solutions. The simulation of cases with continuously moving obstacles have also been addressed, finding transient solutions which are consistent



(a) **(b)**

Figure 2.20: Transient results of the rotating flat body in pipe flow. (a) Horizontal velocity profiles at $x_1 = 2.0$ m and different time instants. (b) Exerted forces over the rotating body walls and pressure difference.

with the governing physics.

Despite the ability of the proposed method to solve these engineering problems, we find it also useful to solve three-dimensional and transient flow problems with immersed moving obstacles. Another future work may be devoted to the solution of the full two-way fluid-structure interaction problem by including the coupled solution of the elasticity problem for the immersed solid.

3 Fixed and non-conforming mesh strategy for incompressible flow and non-linear solid interaction problems

Abstract

In this chapter, the finite element method with a Fixed Mesh strategy using an Non-Conforming Mesh Methods is used to solve the Fluid-Structure Interaction (FSI) problem including large displacements and deformations. The numerical ingredients of the strategy include an implicit temporal integration scheme and a stabilized spatial formulation for the fluid flow problem. The FSI approach includes the computational methodology to deal with the multiple finite element meshes representing the fluid and solid domains, the projection of the fluid tractions onto the solid boundary, and the algorithmic approach for the FSI coupling. Several applications are simulated, some of which make use of a simplified rigid body motion scheme for the simulation of embedded bodies. The solution of well-known benchmark problems assures the accuracy of the developed numerical technique. But, precisely, most applications are conceived as a full two-way FSI coupling with the structure being deformed as a hyperelastic solid.

Highlights

- An overlapping domain method for solving two-way FSI problems is applied.
- The Variational Multi-Scale method is used to solve the incompressible flow.
- An hyperelastic model is used to solve the solid deformations.
- The Nitsche method for the weak imposition of boundary conditions is applied over the solid interface.
- Oscillating cylinders, flapping foils and free flying airfoils are successfully tested.

3.1 Introduction

Fluid-Structure Interaction (FSI) studies are related to the analysis of coupled physical effects that cannot be described by isolated studies of fluid flows or solid deformations. These effects happen in several applications, for instance, in the flight of deformable structures like parachutes. In many cases, such as pressure vessel containers or pipelines, the solid structure can be considered rigid and therefore, can be neglected from the fluid dynamics study. The only interest in those cases is to assess the safe support that has to be given by the solid structure to the fluid flow dynamics. The opposite case is actually the widely and historically studied structural mechanics problem, for which the effect of dynamic hydraulic loads are simplified or neglected from the structural analysis. As a direct consequence, the FSI level of complexity has been usually reduced and those solutions have considered at most one-way interaction problems between the fluid and solid.

However, there are other applications in which the coupled effect of the solid deformation and the fluid flow needs to be accounted for. This second approach is called the full two-way FSI coupling due to its bidirectional information transfer. Scientists has devoted a complete effort on studying two-way applications, e.g. the insect flight [122] or the fish swimming [123] have been studied in biology. FSI studies have been also carried out to analyze aneurysms, artificial valves, cardiovascular stents, among other medical issues [124]. Civil structures, such as buildings or bridges, are also affected by FSI phenomena, for example, the Tacoma Narrows Bridge is a famous historical case of a structure that collapsed due to the flutter effect [125], which is an unwanted aero-elastic phenomenon due to the energy transfer from the fluid flow to the solid that generates increasing oscillations over time. Also, in the aeronautical field, this same flutter effect has been investigated in wings and blades [126]. In the case of wind turbines, FSI analysis has been used to better predict the power coefficient of the wind turbines due to the deformation of the blades and tower [127]. It is also applied in the transitory and more challenging analysis of inflatable structures like parachutes [128] or airbags [11]. Finally, FSI analysis has been implemented to design novel energy harvesting mechanisms. This is mainly explained by the detailed description and cheapness to obtain the phenomenological visualization given by this approach.

Flapping foils are one recently developed energy harvesting mechanism. These use the flutter principle to capture kinetic energy from the fluid flow through oscillatory movements [70]. Even though this technology has less impact on wildlife and has low noise levels, its efficiency is below the rotating wind turbines performance. Recent studies have been focused on the power coefficient increase of the flapping foils. For instance, a finite volume method has been used in [72] to study the efficiency of a flapping foil by adding an adaptive Gurney flap over the trailing edge. This modified mechanism has demonstrated to increase the power output of the flapping foil. Another study conducted a wind tunnel test of a flapping wing with a rigid passive actuator trailing edge. The research in [73] found that efficiency increases when the natural frequency of the trailing edge is closer to the oscillation frequency. Also, the Immersed Boundary Method (IBM) has been applied in [74] to study the aerodynamic effect of a flapping foil with an added flexible thin tail at the end of the trailing edge. These types of modifications have demonstrated a rise in the airfoil efficiency, but those configurations have not been fully described by computational methods because IBM can not consider thick airfoils. Similarly, a flapping airfoil with a spring-connected tail has been tested in [75] using the IBM. The findings showed a 7.24% increase in the airfoil efficiency compared to the fully rigid airfoil. An oscillating airfoil with a prescribed deformation of the shape of the solid body and its displacement was tested using a finite volume formulation with moving and conforming meshes in [76], obtaining a 16.1% raise in the power efficiency in contrast to the rigid wing. Similar analyses have been carried out in [77, 78] using the IBM. The objective has been common in these studies: to increase the instantaneous camber of the airfoil and therefore, to obtain a greater difference of pressure between the upper and lower surface of the foil that allows it to move easily and increases the lift.

But the FSI problem is a challenging physical problem since it involves the coupled

solution of the fluid flow and the deformation of solids. Moreover, applications like the aerodynamic performance of flapping foils can be demanding since those involve great displacements of the solid embedded in the fluid flow. In most flapping airfoil cases, the interaction is fully dynamic, and none steady state of the flow field can be reached. Also, practical applications can be only described by complex (three-dimensional non-cartesian) spatial domains. Hence, the numerical approximation of the FSI problem solution is the only possible approach. One method that can achieve this solution is the Finite Element Method (FEM), due to its geometrical capabilities and high interpolation order solution of any physical problem described by partial differential equations.

Several numerical techniques have been developed in the literature, mostly devoted to the coupling between the numerical approximations of the fluid's and solid's sub-domains. Those can be classified based on the way the interface between the discretized spatial subdomains is treated: there exists Moving Mesh Methods (MMM) and Fixed Mesh Methods. The first approach solves the fluid flow in a deformable media that is completely separated from the solid domain. The second approach, on the contrary, fixes the fluid mesh in the back and leaves the deforming and immersed solid on an overlapping foreground mesh. This method accounts for the overlapping region of the solid embedded in the fluid as part of the discrete problem. Also, these approaches are directly related to the Conforming Mesh Methods (CMM) and Non-Conforming Mesh Methods (NCMM) [1]. The former transmits the boundary conditions between the fluid and solid domains, granting that the discretized meshes match at the interface, while the latter defines the solid as an immersed boundary in the interior of the discretized fluid's domain. In this work, the study of NCMM is of main interest, since they are specially designed to solve problems with large movements of the immersed solid boundaries. The main advantage is that regeneration of the fluid's mesh is not needed and hence, the reduced computational cost. However, these types of methods have problems for tracking the shape of the interface, capturing the physical phenomena in the interface region, and transferring information between domains with high fidelity.

The FSI temporal solution can be classified as monolithic and partitioned [1]. In the first approach, the fluid and solid problems are solved together at each time step in the same discrete matrix, while in the second approach each subdomain is evaluated as a separated discrete problem, but the solutions are linked by boundary conditions or source terms. Monolithic approaches are more robust and efficient, but programming is more demanding and the convergence can be more difficult due to the high non-linearities of each problem. Partitioned approaches are more stable and allow greater design flexibility over the numerical methods applied in each subdomain [66, 67]. In partitioned methods, transfer conditions at the interface directly affect the stability and convergence of the numerical problem. The most commonly implemented transfer conditions are the Dirichlet–Neumann (DN), where solid displacements are strongly prescribed as non-slip conditions of the fluid flow and fluid tractions are imposed on the solid boundary. Dirichlet boundary conditions can also be weakly imposed, resulting in a Neumann-Neumann (NN) coupling. However, other strategies like Robin-Neumann (RN) conditions have shown superior results than DN. In [23] it was shown that RN is better than DN for problems suffering from the added mass

effect, which is a numerical instability for similar solid and fluid densities. This instability usually takes the form of a wrong displacement solution due to an additional mass in the structure arising from the fluid closest to the coupling interface. The conclusion that RN is a better approach than RD or DN in such problems has also been reported in [68], where the authors found that RN has adequate numerical convergence and that it is insensitive to the added mass effect.

Furthermore, partitioned techniques can be classified as explicit or implicit [69] depending on the temporal integration scheme. Explicit techniques resolve the dynamic fluid and solid problems accounting only for the previously resolved solutions at previous time instants. These techniques do not guarantee the continuity and momentum balance between the subdomains at the interface and consequently, these methods are not conservative. Another drawback is related to the stability problems when fixing large time steps in the explicit approach. Perhaps their best feature is the low computational cost since they do not require to solve a linear system of equations at each time step. On the contrary, implicit techniques must solve the discrete system of equations at each time step, implying a higher computational cost, but also enforcing the mass and momentum balance between the fluid and solid subdomains. The important development in implicit techniques must be the way to solve the high non-linearity of the coupled system. Indeed, the FSI problem for incompressible flows interacting with deforming hyperelastic solids leads to a highly non-linear system, which is explained by the inner non-linear condition of each fluid and solid problems, but also by the non-linear interaction between them. Newton's method can solve the coupling non-linearity, but it is expensive and difficult to calculate. Another option is the fixed point method or Picard's scheme, which has a slower convergence, but it is easier to program. In the case of Newton's method, it has traditionally solved the separated fluid and problems, and very efficient algorithms have been developed in this sense. Advances in Newton's methods can, therefore, be leveraged and included together with the iterative coupling solution given by a fixed point method (such as Picard) of the highly non-linear coupled system.

A final numerical issue has to do with the numerical prescription of fluid tractions at the solid boundary. The numerical traction values have been traditionally calculated by accounting for the flow solution provided by cut elements at the interface. Despite the use of stabilization strategies for the badly cut elements, spurious gradient discontinuities may be transferred to the solid boundary tractions, especially if the numerical solution inside the covered fluid elements is accounted for. A consequence of this way of calculating fluid tractions is that it may generate non-physical deformations of the solid. One solution can be to neglect the numerical information of the cut elements at the interface and to only account for the information of their neighboring elements fully belonging to the fluid mesh in the computation of the fluid tractions. This reduced traction field can be then projected onto the solid boundary at the interface avoiding non-physical estimations. An algorithmic challenge in this approach is related to the dynamic update of the reduced traction field when the solid-fluid interface moves. This can be addressed by permanently updating the set of cut and covered fluid elements by the solid mesh and by projecting the traction field onto the displaced solid boundary.

In this chapter, a Fixed Mesh strategy using an NCMM - FEM formulation is proposed to solve the FSI problem including large displacements and deformations of the embedded solid in the fluid flow. An implicit scheme is used in the coupling between fluid and solid solutions. The strategy includes recently developed numerical ingredients to deal with the various issues that appear in this problem. Firstly, the Variational Multi-Scale (VMS) [4] method is implemented to stabilize the numerical solutions for convection-dominated problems or when equally interpolation spaces are used for the different variables of the fluid's problem. Other important parts are the weak imposition of essential conditions at the interface through the Nitsche's method and the cut elements stabilization by the Ghost Penalty method. Moreover, the fluid tractions are computed only in the uncovered fluid elements and then projected onto the solid boundary to accurately estimate physical tractions and avoid the transmission of residual gradient discontinuities in the cut elements. As explained before, these discontinuities do not affect the fluid field solution since they are within the covered domain of the solid in the fluid mesh but if transmitted to the solid boundary these may overestimate the applied tractions. Hence, a NN technique is adopted here as the coupling strategy between the solid and fluid domains. Also, the density between domains is assumed to be different to avoid stability problems in the coupling scheme. An implicit time integration scheme for the dynamical fluid flow and solid problems is used, also, for the coupling between them. In this sense, a Picard's scheme is used to solve the strong coupling between the fluid and solid solutions, but the Newton method is used to solve each specific non-linear problem.

The remaining parts of this chapter are organized as follows. In Section 3.2, the variational formulation of the FSI problem is recalled. In that section, the FEM stabilized formulation of the fluid and non-linear solid models is described. There, the numerical ingredients that are implemented to deal with the NCMM issues are explained, but also the temporal integration scheme and the linearization procedures for each problem. Next, the FSI approach is presented in Section 3.3. This includes the computational methodology to deal with the multiple finite element meshes, the projection of the fluid tractions onto the solid boundary, and the algorithmic approach for the fluid and solid solutions coupling. With the numerical approach in hand, some numerical examples are demonstrated in Section 3.4, including the solution of well-known benchmark problems. A flapping foil is also solved in this section. Finally, some conclusions close the chapter in Section 3.5.

3.2 Finite Element Method approximation of the FSI problem

In this section, the FEM stabilized formulation of the FSI problem is first presented. In the beginning, the strong form of the FSI problem is recalled. Then, the separated FEM formulations of the fluid and solid problems are explained in detail. Finally, the fully discretized spatial and temporal approximation of the fluid-solid coupled problem is given at the end of this section.

3.2.1 Strong form of the FSI problem

In the present chapter, the fluid and structure interaction mechanics are modeled by using the incompressible Navier-Stokes equations for the fluid dynamics and the non-linear solid equations for the deformable structure. Let $d = 2, 3$ be the number of spatial dimensions of the FSI problem. Here, the fluid domain is denoted by $\Omega^f \subset \mathbb{R}^d$ and the immersed solid domain by $\Omega^s \subset \mathbb{R}^d | \Omega^s \in \Omega^f$. The fluid-structure interface is denoted as $\Gamma^{f-s} = \partial\Omega^s$. Also, superscripts f and s denote the fluid and the solid, respectively.

3.2.1.1 Solid problem

The solid deformation problem is such that, given the prescribed displacement $\boldsymbol{d}_D$ in the essential boundary Γ^s_D and the prescribed traction $\boldsymbol{t}^s_N$ in the natural boundary Γ^s_N, find the displacement $\boldsymbol{d} : \Omega^s \times t \in (0, T) \to \mathbb{R}^d$ (not to be confused with the spatial dimension d), such that:

$$\begin{aligned} \rho^s \partial_t^2 \boldsymbol{d} - \nabla \cdot \boldsymbol{P}(\boldsymbol{d}) &= \boldsymbol{f}^s, && \text{in } \Omega^s, t \in (0, T), \\ \boldsymbol{d} &= \boldsymbol{d}_D && \text{on } \Gamma^s_D, t \in (0, T), \\ \boldsymbol{P}(\boldsymbol{d}) \cdot \boldsymbol{n}^s &= \boldsymbol{t}^s_N && \text{on } \Gamma^s_N, t \in (0, T), \\ \boldsymbol{d} &= \boldsymbol{d}^0 && \text{in } \Omega^s, t \in (0, T), \end{aligned} \tag{3.1}$$

where $\boldsymbol{n}^s$ is the unit outward normal to the boundary $\partial\Omega^s$, ρ^s is the density of the solid, $\boldsymbol{f}^s$ is the body force on the solid, and $\boldsymbol{P}(\mathbf{d})$ is the first Piola-Kirchhoff stress tensor of the solid.

The first and second Piola-Kirchhoff stress tensor for a hyper-elastic material can be defined as:

$$\boldsymbol{P}(\boldsymbol{d}) = \boldsymbol{F} \cdot \frac{\partial W(\boldsymbol{E})}{\partial \boldsymbol{E}}, \qquad \boldsymbol{S}(\boldsymbol{d}) = \frac{\partial W(\boldsymbol{E})}{\partial \boldsymbol{E}}, \tag{3.2}$$

respectively, where $W : \mathbb{R}^{d \times d} \to \mathbb{R}^+$ is a given strain-energy density, $\boldsymbol{F} = \boldsymbol{I} + \nabla \boldsymbol{d}$ is the deformation gradient, $\boldsymbol{E} = \frac{1}{2}(\boldsymbol{C} - \boldsymbol{I})$ the Green-Lagrange strain tensor, and $\boldsymbol{C} = \boldsymbol{F}^T \cdot \boldsymbol{F}$ the right Cauchy-Green deformation tensor. One of the simplest hyperelastic models is the St. Venant-Kirchoff for a nonlinear, elastic, and isotropic material, which defines the second Piola-Kirchoff stress as

$$\boldsymbol{S}(\boldsymbol{d}) = 2\mu^s \boldsymbol{E} + \lambda^s \mathrm{tr}(\boldsymbol{E}) \boldsymbol{I}, \tag{3.3}$$

where $\mu^s = \dfrac{E^s}{2(1+\nu^s)}$ and $\lambda^s = \dfrac{E^s \nu^s}{(1+\nu^s)(1-2\nu^s)}$ are the Lamé coefficients given as functions of the Young modulus E^s and the Poisson ratio ν^s. Hence, the first Piola-Kirchhoff stress tensor is computed from this last expression by contracting it with $\boldsymbol{F}$.

3.2.1.2 Fluid problem

The fluid flow problem is such that, given the prescribed velocity $\boldsymbol{u}_D$ in the essential boundary Γ_D^f and the prescribed traction $\boldsymbol{t}_N^f$ in the natural boundary Γ_N^f, find the velocity $\boldsymbol{u} : \Omega^f \times t \in (0,T) \rightarrow \mathbb{R}^d$ and the pressure $p : \Omega^f \times t \in (0,T) \rightarrow \mathbb{R}$ satisfying

$$\begin{aligned} \rho^f \left(\partial_t \boldsymbol{u} + \boldsymbol{u} \cdot \nabla \boldsymbol{u}\right) - \nabla \cdot \left(2\mu \nabla^{sym} \boldsymbol{u}\right) + \nabla p = \boldsymbol{f}^f \quad & \text{in } \Omega^f, t \in (0,T), \\ \nabla \cdot \boldsymbol{u} = 0 \quad & \text{in } \Omega^f, t \in (0,T), \\ \boldsymbol{u} = \boldsymbol{u}_D \quad & \text{on } \Gamma_D^f, t \in (0,T), \\ \boldsymbol{\sigma}^f \cdot \boldsymbol{n}^f = \boldsymbol{t}_N^f \quad & \text{on } \Gamma_N^f, t \in (0,T), \\ \boldsymbol{u} = \boldsymbol{u}^0 \quad & \text{in } \Omega^f, t \in (0,T), \end{aligned} \tag{3.4}$$

where $\boldsymbol{n}^f$ is the unit outward normal to the boundary Γ^f and $(0,T)$ is the time interval, ρ^f is the density of the fluid, μ^f is the dynamic viscosity, and $\mathbf{f}^f$ is the body force. The Cauchy stress tensor is calculated as $\boldsymbol{\sigma}^f = 2\mu \nabla^{sym} \boldsymbol{u} - p\boldsymbol{I}$, where $\nabla^{sym}(\cdot) := \frac{1}{2}\left(\nabla(\cdot) + (\nabla(\cdot))^\top\right)$ is the symmetric gradient operation and $\boldsymbol{I}$ is the identity tensor. For the sake of conciseness, the first two equations of (3.4) can be written in a linearized form as

$$\boldsymbol{M}\partial_t \boldsymbol{U} + \boldsymbol{\mathcal{L}}\left(\boldsymbol{u}; \boldsymbol{U}\right) = \boldsymbol{F} \qquad \text{in } \Omega^f,\ t \in (0,T)\,, \tag{3.5}$$

where $\boldsymbol{U} = [\boldsymbol{u}, p]^T$ is the vector of unknowns, $\boldsymbol{\mathcal{L}}\left(\boldsymbol{u}; \boldsymbol{U}\right)$ is the non-linear differential operator, $\boldsymbol{M}$ is the transient matrix, and $\boldsymbol{F}$ is the vector of forces, which are defined, respectively as

$$\boldsymbol{\mathcal{L}}\left(\boldsymbol{u}; \boldsymbol{U}\right) := \begin{bmatrix} \rho\left(\boldsymbol{u} \cdot \nabla \boldsymbol{u}\right) - \mu \Delta \boldsymbol{u} + \nabla p \\ \nabla \cdot \boldsymbol{u} \end{bmatrix}, \quad \boldsymbol{M} = \begin{bmatrix} \rho \boldsymbol{I} & 0 \\ 0 & 0 \end{bmatrix}, \quad \text{and} \quad \boldsymbol{F} = \begin{bmatrix} \boldsymbol{f}^f \\ 0 \end{bmatrix}. \tag{3.6}$$

3.2.1.3 Transfer conditions

The coupling between fluid Ω^f and solid Ω^s domains can be done in several ways. The preferred approach in this work is to use Neumann-Neumann conditions. It is explained because the moving solid boundary randomly cuts the fluid mesh triggering the numerical difficulty of prescribing Dirichlet conditions over the fluid nodes. Hence, those are imposed weakly by adding terms to the discrete flow equations inside the fluid domain. At the fluid-structure interface Γ^{f-s} the velocity of the fluid and solid coincide, while the tractions must be equal and opposite:

$$\begin{aligned} \alpha\left(\boldsymbol{u} - \boldsymbol{u}^s\right) = 0 \quad & \text{on } \Gamma^{f-s}, t \in (0,T), \\ \boldsymbol{\sigma}^f \cdot \boldsymbol{n}^f + \boldsymbol{P}(\boldsymbol{d}) \cdot \boldsymbol{n}^s = 0 \quad & \text{on } \Gamma^{f-s}, t \in (0,T), \end{aligned} \tag{3.7}$$

where $\boldsymbol{u}^s = \partial_t \boldsymbol{d}$ is the velocity of the solid boundary, α is a penalization parameter, and $\boldsymbol{n}^f = -\boldsymbol{n}^s$.

3.2.2 Weak form of the FSI problem

Some standard notation is introduced to formulate the weak problems for the fluid and the solid. Let $L^2(\Omega)$ be the space of square-integrable functions in the domain Ω, and $H^m(\Omega)$ the space of functions whose derivatives are of order up to $m \geq 0$ and that belong to $L^2(\Omega)$. The space $H_0^1(\Omega)$ is the space of functions in $H^1(\Omega)$ that vanish on Γ. The L^2-inner product in Ω is denoted by $(\cdot,\cdot)$ and its norm by $\|\cdot\|$.

Let $\boldsymbol{\mathcal{W}}_0 = (H_0^1(\Omega))^d$ be the space where the velocity for the fluid and the displacements for the solid for each fixed time t are seek, and $\mathcal{Q} \subset L^2(\Omega)/\mathbb{R}$ as the space where the pressure for each fixed time t is seek.

3.2.2.1 Weak form of the solid problem

The weak formulation of the hyperelastic solid equation (4.2) is written as to find $\boldsymbol{d} \in \boldsymbol{\mathcal{W}}_0$, such that the boundary and initial conditions are satisfied and

$$\left(\rho^s \partial_t^2 \boldsymbol{d}, \boldsymbol{v}\right)_{\Omega^s} + (\nabla \cdot \boldsymbol{P}(\boldsymbol{d}), \boldsymbol{v})_{\Omega^s} = (\rho^s \boldsymbol{b}^s, \boldsymbol{v})_{\Omega^s}\,, \tag{3.8}$$

for all weighting functions $\boldsymbol{v} \in \boldsymbol{\mathcal{W}}_0$, where it is assumed that $\langle \rho^s \boldsymbol{b}^s, \boldsymbol{v}\rangle_{\Omega^s}$ is well defined. The second term on the Left Hand Side (LHS) of the previous expression can be integrated by parts, giving

$$\left(\rho^s \partial_t^2 \boldsymbol{d}, \boldsymbol{v}\right)_{\Omega^s} - (\boldsymbol{S}(\boldsymbol{d}), \boldsymbol{E}(\boldsymbol{d}, \boldsymbol{v}))_{\Omega^{s,0}} = (\rho^s \boldsymbol{b}^s, \boldsymbol{v})_{\Omega^s} + \langle \boldsymbol{t}_N^s, \boldsymbol{v}\rangle_{\Gamma^s}, \qquad \forall \boldsymbol{v} \in \boldsymbol{\mathcal{W}}_0. \tag{3.9}$$

It is noted that the previous equation is based on the initial reference frame $\Omega^{s,0}$ named Total Lagrange formulation, but it can be converted to the current reference frame Ω^s using the expression of the Cauchy stress $\boldsymbol{\sigma}^s$ and the engineering strain $\boldsymbol{\epsilon} := \nabla^{\text{sym}}(\boldsymbol{d})$ which is referred as the Updated Lagrange formulation. Hence, the solid problem is to find $\boldsymbol{d} \in \boldsymbol{\mathcal{W}}_0$ such that

$$\left(\rho^s \partial_t^2 \boldsymbol{d}, \boldsymbol{v}\right)_{\Omega^s} - (\boldsymbol{P}(\boldsymbol{d}), \boldsymbol{\epsilon}(\boldsymbol{v}))_{\Omega^s} = (\rho^s \boldsymbol{b}^s, \boldsymbol{v})_{\Omega^s} + \langle \boldsymbol{t}_N^s, \boldsymbol{v}\rangle_{\Gamma^s}, \qquad \forall \boldsymbol{v} \in \boldsymbol{\mathcal{W}}_0. \tag{3.10}$$

3.2.2.2 Weak form of the flow problem

The weak form of (3.4) can be written in the more convenient vector form as to find $\boldsymbol{U} = [\boldsymbol{u}, p]^\top \in \boldsymbol{\mathcal{W}}_0 \times \mathcal{Q}$, such that appropriate initial and boundary conditions are satisfied, and

$$\left(\boldsymbol{M}\frac{\partial \boldsymbol{U}}{\partial t}, \boldsymbol{V}\right) + B(\hat{\boldsymbol{U}}; \boldsymbol{U}, \boldsymbol{V}) = L(\boldsymbol{V}), \tag{3.11}$$

for all weighting functions $\boldsymbol{V} = [\boldsymbol{v}, q]^\top \in \boldsymbol{\mathcal{W}}_0 \times \mathcal{Q}$. Here $\left(\boldsymbol{M}\frac{\partial \boldsymbol{U}}{\partial t}, \boldsymbol{V}\right)$, $B(\hat{\boldsymbol{U}}; \boldsymbol{U}, \boldsymbol{V})$, and $L(\boldsymbol{V})$ are the transient, bilinear and linear forms, respectively, that for the incompressible

Navier-Stokes equations are defined as

$$\left(\boldsymbol{M}\frac{\partial \boldsymbol{U}}{\partial t}, \boldsymbol{V}\right) = \left(\rho^f \frac{\partial \boldsymbol{u}}{\partial t}, \boldsymbol{v}\right),$$
$$B\left(\hat{\boldsymbol{U}}; \boldsymbol{U}, \boldsymbol{V}\right) = \left(\rho^f \left(\hat{\boldsymbol{u}} \cdot \nabla \boldsymbol{u}\right), \boldsymbol{v}\right)_{\Omega^f} + 2\left(\mu^f \nabla^{\mathrm{sym}}(\boldsymbol{u}), \nabla^{\mathrm{sym}}(\boldsymbol{v})\right)_{\Omega^f} - (p\nabla, \boldsymbol{v})_{\Omega^f} + (q\nabla, \boldsymbol{u})_{\Omega^f},$$
$$L\left(\boldsymbol{V}\right) = (\boldsymbol{f}, \boldsymbol{v})_{\Omega^f} + \langle \boldsymbol{t}_N^f, \boldsymbol{v}\rangle_{\Gamma_N^f}.$$

, where $\widehat{\boldsymbol{U}} = (\widehat{\boldsymbol{u}}, p)^\top$ are the non-linear components in the equations.

3.2.3 Finite Element Formulation of the FSI problem

Let $\boldsymbol{T}_h$ be a finite element partition of the domain Ω composed by non-overlapping elements K with diameter h_k. The diameter of the element partition defined by $h = \max\{h_k | K \in \boldsymbol{T}_h\}$.

Finite Element spaces $\boldsymbol{\mathcal{W}}_h \subset \boldsymbol{\mathcal{W}}$ and $\mathcal{Q}_h \subset \mathcal{Q}$ are constructed made of continuous piece-wise polynomial functions in space. In this work, equal-order Finite Element spaces are used for velocity, displacement and pressure variables, such that:

$$\boldsymbol{\mathcal{W}}_h := \left\{\boldsymbol{v} \in \left[\mathcal{C}^0\left(\Omega\right)\right]^d : \boldsymbol{v} \in \left[\mathcal{Q}^p\left(K\right)\right]^d \forall K \in \mathcal{T}_h\right\}, \tag{3.12}$$
$$\mathcal{Q}_h := \left\{q \in \mathcal{C}^0\left(\Omega\right) : q \in \mathcal{Q}^p\left(K\right) \forall K \in \mathcal{T}_h\right\}. \tag{3.13}$$

These finite spaces are constructed using piece-wise polynomials $\mathcal{Q}^p$ of order p (not to be confused with pressure), such that those spaces tend to the continuous spaces in the case of $h \to 0$ or $p \to \infty$.

3.2.3.1 Galerkin approximation of the solid problem

The Total Lagrange Formulation of the non-linear solid problem is of especial interest in the present work. The Galerkin approximation considers replacing $\boldsymbol{\mathcal{W}}_0$ by the Finite Element space $\boldsymbol{\mathcal{W}}_{h,0}$ in equation (3.10). Therefore, the discrete solid problem consists in finding $\boldsymbol{d}_h \in \boldsymbol{\mathcal{W}}_{h,0}$ such that the boundary and initial conditions are satisfied and

$$\left(\rho^s \partial_t^2 \boldsymbol{d}_h, \boldsymbol{v}_h\right)_{\Omega^s} - \left(\boldsymbol{S}(\boldsymbol{d}_h), \boldsymbol{E}(\boldsymbol{d}_h, \boldsymbol{v}_h)\right)_{\Omega^{s,0}} = (\rho^s \boldsymbol{b}^s, \boldsymbol{v}_h)_{\Omega^s} + \langle \boldsymbol{t}_N^s, \boldsymbol{v}_h\rangle_{\Gamma^s}, \quad \forall \boldsymbol{v}_h \in \boldsymbol{\mathcal{W}}_{h,0}. \tag{3.14}$$

3.2.3.2 Galerkin approximation of the flow problem

The Galerkin approximation of the weak problem (3.11) can be stated as: find $\boldsymbol{U}_h = [\boldsymbol{u}_h, p_h]^T \in \boldsymbol{\mathcal{W}}_{h,0} \times \mathcal{Q}_h$ such that

$$\left(\boldsymbol{M}\frac{\partial \boldsymbol{U}_h}{\partial t}, \boldsymbol{V}_h\right) + B(\boldsymbol{U}_h, \boldsymbol{V}_h) = L(\boldsymbol{V}_h), \tag{3.15}$$

for all weighting functions $\boldsymbol{V}_h = [\boldsymbol{v}_h, q_h]^T \in \boldsymbol{\mathcal{W}}_{h,0} \times \mathcal{Q}_h$.

3.2.3.3 Global stabilization of the flow problem

The Galerkin approximation for the Navier-Stokes equation above presents instability problems for convection dominant flows or when the interpolation spaces used for pressure and velocity variables do not satisfy the *inf-sup* condition. A Variational Multi-Scale (VMS) [4] stabilized formulation is adopted in the present work to overcome these instability problems. The numerical technique consists in adding the following term to the LHS of Galerkin equation (3.15) in order to add numerical diffusivity:

$$S\left(\hat{\boldsymbol{U}}_h; \boldsymbol{U}_h, \boldsymbol{V}_h\right) = \sum_{K \in \boldsymbol{\mathcal{T}}_h} \left(\boldsymbol{\mathcal{L}}^*(\hat{\boldsymbol{u}}_h; \boldsymbol{V}_h), \boldsymbol{\tau}\left(\hat{\boldsymbol{U}}_h\right) \widetilde{\boldsymbol{P}}(\boldsymbol{R}_h)\right)_K, \tag{3.16}$$

where $\boldsymbol{\mathcal{L}}^*(\hat{\boldsymbol{u}}_h; \boldsymbol{V}_h)$ is the adjoint of the non-linear fluid flow differential operator in (3.6), $\boldsymbol{R}_h$ is the residual defined as $\boldsymbol{R}_h = \boldsymbol{F} - \partial_t \boldsymbol{U}_h - \boldsymbol{\mathcal{L}}(\hat{\boldsymbol{u}}_h; \boldsymbol{U}_h)$, and $\widetilde{\boldsymbol{P}}(\boldsymbol{R}_h)$ stands for a L^2-projection of the finite residual into a certain space. Indeed, this last projection term can be defined to be $\widetilde{\boldsymbol{P}}(\boldsymbol{R}_h) = \boldsymbol{I}\boldsymbol{R}_h$ to what is referred as the *Algebraic Sub-Grid Scales* (ASGS) approach. Another recent approach defines the projection of the residual as $\widetilde{\boldsymbol{P}}(\boldsymbol{R}_h) = \boldsymbol{P}_h^{\perp}(\boldsymbol{R}_h) = \boldsymbol{I}\boldsymbol{R}_h - \boldsymbol{P}_h(\boldsymbol{R}_h)$ meaning that the projected space is orthogonal to the finite space. This approach is the so-called *Orthogonal Sub-Grid Scales* (OSGS) method in VMS [129]. Additionally, $\boldsymbol{\tau}\left(\hat{\boldsymbol{U}}_h\right)$ is a matrix of stabilization parameters associated with the momentum and mass conservation equations which depends over the unknowns.

3.2.3.4 Weak imposition of Dirichlet boundary conditions

Dirichlet boundary conditions $\boldsymbol{u}_D$ at the fluid-structure interface Γ^{f-s} are imposed weakly. Specifically, these are prescribed by means of the Nitsche's method [118], which adds $W(\boldsymbol{u}_D; \hat{\boldsymbol{U}}_h; \boldsymbol{U}_h, \boldsymbol{V}_h)$ to the LHS of (3.15):

$$\begin{aligned} W(\boldsymbol{u}_D; \hat{\boldsymbol{U}}_h; \boldsymbol{U}_h, \boldsymbol{V}_h) =& \frac{\alpha(\boldsymbol{u}_h)}{h} \langle (\boldsymbol{u}_h - \boldsymbol{u}_D), \boldsymbol{v}_h \rangle_{\Gamma^{f-s}} - \langle (2\mu \boldsymbol{n} \cdot \nabla^{sym} \boldsymbol{u}_h - p_h \boldsymbol{n}), \boldsymbol{v}_h \rangle_{\Gamma^{f-s}} \\ & - \langle (2\mu \boldsymbol{n} \cdot \nabla^{sym} \boldsymbol{v}_h - q_h \boldsymbol{n}), (\boldsymbol{u}_h - \boldsymbol{u}_D) \rangle_{\Gamma^{f-s}}, \end{aligned} \tag{3.17}$$

where the penalization parameter $\alpha(\boldsymbol{u}_h)$ depends on the viscosity and the velocity.

3.2.3.5 Cut cell stabilization

As instabilities may occur in highly convective flows due to bad cuts elements at the interface, it is necessary to add local stabilization. In this sense, the Ghost Penalty stabilization using orthogonal projections to the FE space proposed in [24] and that was

discussed in Chapter 2 is implemented, by including the term $S_G\left(\hat{\boldsymbol{U}}_h;\boldsymbol{U}_h,\boldsymbol{V}_h\right)$ to the LHS of (3.21):

$$\begin{aligned} S_G\left(\hat{\boldsymbol{U}}_h;\boldsymbol{U}_h,\boldsymbol{V}_h\right) = \sum_{K\in\boldsymbol{T}_h^c} \sigma_G[\gamma_1\left(\boldsymbol{u}_h\right)\left\langle\nabla\boldsymbol{v}_h,\boldsymbol{P}_h^{\perp}(\nabla\boldsymbol{u}_h)\right\rangle_K \\ +\gamma_2\left(\boldsymbol{u}_h\right)\left\langle\nabla q_h,\boldsymbol{P}_h^{\perp}(\nabla p_h-\rho\boldsymbol{f})\right\rangle_K], \end{aligned} \tag{3.18}$$

where σ_G is a penalty parameter defined by the user, and $\gamma_1\left(\boldsymbol{u}_h\right)$, $\gamma_2\left(\boldsymbol{u}_h\right)$ are algorithmic constants that depend on the matrix of stabilization parameters $\boldsymbol{\tau}\left(\boldsymbol{u}_h\right)$. Note that this term is acting only in the subset of cut fluid elements by the interface $K\in\boldsymbol{T}_h^{cut}\subset\boldsymbol{T}_h$. Even though better results could be found by penalizing the weak imposition and ghost penalty terms with $\alpha\beta^{-1}$ and $\gamma_1\beta^{-1}$ instead of α and γ_1 alone, where β is calculated as the fluid's fraction at each cut element, this method is not implemented here for the sake of algorithmic easiness.

3.2.3.6 Stabilized FEM formulation of the flow problem

Let the non-linear terms in the previous formulation be arranged in the following bi-linear form

$$\begin{aligned} A(\boldsymbol{u}_D;\hat{\boldsymbol{U}}_h;\boldsymbol{U}_h,\boldsymbol{V}_h) :=& B\left(\hat{\boldsymbol{U}}_h;\boldsymbol{U}_h,\boldsymbol{V}_h\right)+S\left(\hat{\boldsymbol{U}}_h;\boldsymbol{U}_h,\boldsymbol{V}_h\right) \\ &+W(\boldsymbol{u}_D;\hat{\boldsymbol{U}}_h;\boldsymbol{U}_h,\boldsymbol{V}_h)+S_G\left(\hat{\boldsymbol{U}}_h;\boldsymbol{U}_h,\boldsymbol{V}_h\right). \end{aligned}$$

Hence, the stabilized formulation of the fluid flow problem is given as to find $\boldsymbol{U}_h=[\boldsymbol{u}_h,p_h]^T\in\boldsymbol{\mathcal{W}}_{h,0}\times\mathcal{Q}_h$ such that:

$$\left(\boldsymbol{M}\frac{\partial\boldsymbol{U}_h}{\partial t},\boldsymbol{V}_h\right)+A(\boldsymbol{u}_D;\hat{\boldsymbol{U}}_h;\boldsymbol{U}_h,\boldsymbol{V}_h)=L\left(\boldsymbol{V}_h\right), \tag{3.19}$$

for all $\boldsymbol{V}_h\in\boldsymbol{\mathcal{W}}_{h,0}\times\mathcal{Q}_h$, subjected to the appropriate boundary and initial conditions.

3.2.4 Time integration of the FSI problem

An implicit partitioned time integration scheme is used to integrate the time derivatives of Eqs. (3.14) and (3.19). This methodology is chosen since large time steps can be used due to its unconditional stability. The way to integrate the time derivatives of the separated problems is explained next.

Let the time interval $(0,t_f)$ be partitioned in a sequence of discrete time steps $0=t^0<t^1<...<t^N=t_f$, with $\delta t>0$ the (constant) time step-size defining $t^{n+1}=t^n+\delta t$ for $n=0,1,2,...,N$.

3.2.4.1 Time integration of the solid problem

The discretization method for the second-order time derivative in the solid problem is done employing the second-order central difference scheme. The time and spatial discrete form of the solid problem consists in solving the dynamic equation (3.14) at t^{n+1}: given the initial conditions $\boldsymbol{d}^0$, find $\boldsymbol{d}_h^{n+1} \in \boldsymbol{\mathcal{W}}_{h,0}$ for $n = 1, 2, ..$, such that the boundary conditions are satisfied and

$$\left(\rho^s \frac{\boldsymbol{d}_h^{n+1} - 2\boldsymbol{d}_h^n + \boldsymbol{d}_h^{n-1}}{(\delta t)^2}, \boldsymbol{v}_h\right)_{\Omega^s} - (\boldsymbol{S}(\boldsymbol{d}_h^n), \boldsymbol{E}(\boldsymbol{d}_h^n, \boldsymbol{v}_h))_{\Omega^{s,0}} = (\rho^s \boldsymbol{b}^s, \boldsymbol{v}_h)_{\Omega^s} + \langle \boldsymbol{t}_N^s, \boldsymbol{v}_h \rangle_{\Gamma^s}, \quad \forall \boldsymbol{v}_h \in \boldsymbol{\mathcal{W}}_{h,0}. \tag{3.20}$$

To begin the recursion at $n = 1$, two successive displacements are required, one of which is precisely the initial condition for displacements $\boldsymbol{d}^0$, and another in the fictitious previous time instant, which is considered as zero.

3.2.4.2 Time integration of the fluid problem

In the case of the time discretization method of the first-order time derivative of the stabilized fluid equations, the Backward Differentiation Formula of order two (BDF2) is adopted in the present work. Hence, in the time and spatial discrete form of the fluid problem consists in solving the fluid equation (3.19) as follows: given the initial conditions $\boldsymbol{U}^0$, for $n = 0, 1, 2, ..$, find $\boldsymbol{U}_h^{n+1} \in \boldsymbol{\mathcal{W}}_{h,0}$, such that the boundary conditions are satisfied and

$$\left(\boldsymbol{M}\left(\frac{3\boldsymbol{U}_h^{n+1}}{2\delta t} - \frac{2\boldsymbol{U}_h^n}{\delta t} + \frac{\boldsymbol{U}_h^{n-1}}{2\delta t}\right), \boldsymbol{V}_h\right)_{\Omega^f} + A\left(\boldsymbol{u}_D^{n+1}; \boldsymbol{U}_h^{n+1}; \boldsymbol{U}_h^{n+1}, \boldsymbol{V}_h\right) = L\left(\boldsymbol{V}_h\right), \tag{3.21}$$

In this case, two successive starting values are required when $n = 1$ to solve $\boldsymbol{U}^{n+1}$: the initial condition $\boldsymbol{U}^0$ and the solution at the fictitious and previous time instant, which is exactly zero.

3.2.5 Linearization schemes

The coupling between fluid and solid equations is extremely nonlinear. This is explained by the non-linearity of each separate problem, and the rise in complexity given by the coupling. Hence, careful linearization schemes must be implemented to achieve a converged solution of the coupled problem. The linearization techniques for each solid and fluid problem are explained in the next lines.

3.2.5.1 Linearization scheme of the solid problem

The non-linear variational form arising from the St. Venant-Kirchoff elastic model is highly non-linear. Hence, the solution of the hyperelastic problem requires a linearization scheme to be able to determine the displacement field at each coupling iteration. The Newton-Rhapson method over the dicrete formulation is implemented to deal with the non-linearities arising from the hyperelastic problem: at each step $n+1$, the i superscript is introduced as an iteration counter, and given $\boldsymbol{d}_h^{n+1,i}$ (initial guess), $\boldsymbol{d}_h^n$, $\boldsymbol{d}_h^{n-1}$, and $\boldsymbol{t}_N^s$, the solid finite element unknowns at $i+1$ are solved with

$$\boldsymbol{d}_h^{n+1,i+1} = \boldsymbol{d}_h^{n+1,i} - J^s(\boldsymbol{d}_h^{n+1})^{-1} F^s(\boldsymbol{t}_N^s; \boldsymbol{d}_h^{n+1,i}, \boldsymbol{d}_h^n, \boldsymbol{d}_h^{n-1}, \boldsymbol{V}_h), \tag{3.22}$$

by considering $F^s(\boldsymbol{t}_N^s; \boldsymbol{d}_h^{n+1,i}, \boldsymbol{d}_h^n, \boldsymbol{d}_h^{n-1}, \boldsymbol{V}_h)$ as the complete variational form in (3.20) and

$$J^s(\boldsymbol{d}_h^{n+1}) = \frac{\partial F^s(\boldsymbol{t}_D^s; \boldsymbol{d}_h^{n+1}, \boldsymbol{d}_h^n, \boldsymbol{d}_h^{n-1}, \boldsymbol{V}_h)}{\partial \boldsymbol{d}_h^{n+1}}. \tag{3.23}$$

The automated differentiation in [130] is used to calculate the *Jacobian* in Eq. (3.23) with respect to the displacements.

3.2.5.2 Linearization scheme for the fluid problem

The implicit temporal scheme also brings the difficulty of solving the non-linearities of the discrete fluid flow problem. In this case, the Newton-Rhapson scheme is also implemented to solve the non-linearities of Eq. (3.21). Indeed, this linearization strategy is done as follows: at each step $n+1$, an iteration counter i is used as superscript of the discrete solutions, and given $\boldsymbol{U}_h^{n+1,i}$ (initial guess), $\boldsymbol{U}_h^n$, $\boldsymbol{U}_h^{n-1}$, and $\boldsymbol{u}_D^{n+1}$, the fluid finite element unknowns at $i+1$ are computed with

$$\boldsymbol{U}_h^{n+1,i+1} = \boldsymbol{U}_h^{n+1,i} - J^f(\boldsymbol{U}_h^{n+1})^{-1} F^f(\boldsymbol{u}_D^{n+1}; \boldsymbol{U}_h^{n+1,i}, \boldsymbol{U}_h^n, \boldsymbol{U}_h^{n-1}, \boldsymbol{V}_h), \tag{3.24}$$

by considering $F^f(\boldsymbol{u}_D^{n+1}; \boldsymbol{U}_h^{n+1,i}, \boldsymbol{U}_h^n, \boldsymbol{U}_h^{n-1}, \boldsymbol{V}_h)$ as the complete variational form in (3.21) and

$$J^f(\boldsymbol{U}_h^{n+1}) = \frac{\partial F^f(\boldsymbol{u}_D^{n+1}; \boldsymbol{U}_h^{n+1}, \boldsymbol{U}_h^n, \boldsymbol{U}_h^{n-1}, \boldsymbol{V}_h)}{\partial \boldsymbol{U}_h^{n+1}}. \tag{3.25}$$

In this case, the discrete linearized equation for the fluid flow problem is detailed in [131]. In that same article, it has been explained that the Newton-Raphson scheme applied to the system of fluid equations is the most efficient one.

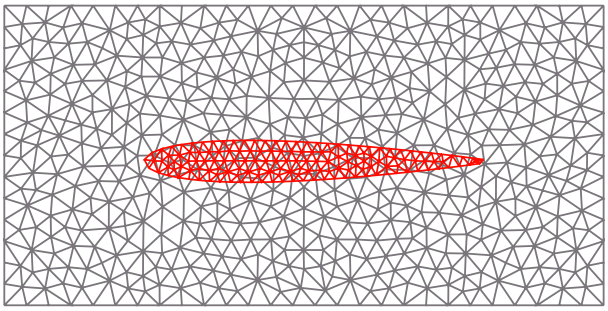


Figure 3.1: Scheme of overlapping meshes

3.3 FSI computational strategy

In this section, the numerical and algorithmic strategies to deal with the FSI coupling between the fluid and solid sub-domains are presented. First, the approach to deal with the collection of fluid and solid meshes is explained. Especially with the interface definition on the fluid's mesh using computational geometry. Then, the numerical treatment of the transfer conditions at the interface is detailed. Finally, the coupling scheme and its algorithmic are described.

3.3.1 Multi mesh finite element method

In the discrete approach, an overlapped mesh method using two different meshes is implemented, to what is referred as to *multi mesh* finite element methods. Especially, the solid's mesh overlaps arbitrarily the fluid's elements at the background mesh (see Figure 3.1). These two meshes together define the computational domain in which the FSI problem is solved. Indeed, the geometric tracking of the immersed boundary in the collection of two meshes can be challenging in the algorithmic sense. Here the sharp interface approach is adopted, which defines the position of the interface exactly by cutting the fluid elements in the background mesh. Hence, this approach leads to three subsets of different type of fluid elements in the background mesh: the cut fluid elements by the interface $\boldsymbol{T}_h^{cut}$, the completely covered fluid elements by the solid mesh $\boldsymbol{T}_h^{cov}$, and the remaining (uncut and uncovered) fluid elements $\boldsymbol{T}_h \setminus \boldsymbol{T}_h^c$, where $\boldsymbol{T}_h^c = \boldsymbol{T}_h^{cut} \cup \boldsymbol{T}_h^{cov}$. The sharp interface approach over the multi mesh finite method has been traditionally tackled through the development of computational geometry methods. For instance, by efficiently and robustly computing intersections and quadrature points. In this work, the efficient computational geometry and algorithmic approach of two-dimensional mesh-mesh intersections in FEniCS library [130] are adopted. The implementation in FEniCS relies on efficient generation and traversal of Axis-Aligned Bounding Box Trees (AABB trees). This methodology allows for low-level operations for computing intersections of triangles and tetrahedra and generation of quadrature points on cut cells.

3.3.2 Neumann-Neumann method for transfer conditions

As explained before, spurious fluid tractions may be transferred to the solid boundary tractions if the numerical solution inside the covered fluid elements is accounted for. The approach in the present chpater is to neglect the numerical information of the set of cut and covered fluid elements $\boldsymbol{T}_h^c$ in the computation of the fluid stresses. Hence, only the information of their neighboring uncovered elements —fully belonging to the fluid mesh— is used in the computation of the stress field. This stress field $\boldsymbol{\sigma}_{h\setminus c}^f$ is computed using the fluid flow solution $\boldsymbol{U}_h$ in $\boldsymbol{T}_h \setminus \boldsymbol{T}_h^c$ and projected onto the solid boundary at the interface Γ^{f-s} avoiding non-physical estimations of $\boldsymbol{t}_N^s$. An algorithmic challenge in this approach is related to the dynamic update of the $\boldsymbol{T}_h \setminus \boldsymbol{T}_h^c$ sub mesh when the solid-fluid interface moves. This issue is addressed by permanently updating the set of cut and covered fluid elements $\boldsymbol{T}_h^c$ in the multi mesh finite approach. In practice, this is done by recalculating the overlapping part of the fluid mesh each time the solid mesh deforms and creating a sub mesh contained in the background fluid mesh, which consists of $K \in \boldsymbol{T}_h \setminus \boldsymbol{T}_h^c$. Projections of fluid stresses from this space to the solid boundary is done by means of a L^2-projection onto the solid finite space of stresses $\boldsymbol{P}_h^s(\boldsymbol{\sigma}_h^f)$. Hence, the following variational problem is solved:

$$\int_{\Omega^s} \boldsymbol{X}_h : \boldsymbol{P}_h^s(\boldsymbol{\sigma}_h^f)\, d\Omega = \int_{\Omega^s} \boldsymbol{X}_h : \boldsymbol{\sigma}_h^f\, d\Omega \qquad \forall \quad \boldsymbol{X}_h \in \boldsymbol{\mathcal{Y}}_{h,0}.$$

Here $\boldsymbol{\mathcal{Y}}_{h,0}$ is the discrete second rank tensor space made of picewise polynomials. Finally, the tractions over the interface can be computed from this projected fluid stress field.

3.3.3 Coupling treatment

An implicit partitioned approach is used to integrate in time, such that the interface coupling must balance between the fluid and the solid solutions at each time step. As described in the introduction section, Neumann-Neumann coupling conditions are implemented in this work. In this regard, a Picard's scheme at each time step is adopted to deal with the implicit coupling of the transfer conditions: at each Picard's iteration, the fluid flow is solved considering the weak imposition of no-slip conditions on the fluid elements next to the interface. Also, the fluid flow solution is stabilized by the Ghost Penalty technique at the cut elements. Then, the fluid flow tractions are prescribed to deform the solid domain. The complete Fluid-Structure Interaction (FSI) implicit scheme is presented in Algorithm 1.

Another important ingredient in the iterative Picard's scheme is the restoration of the solid deformed mesh in case of not satisfying the traction balance at the interface. That is, if in a given iteration the deformation of the solid mesh does not meet the balance, then it must be restored to the previous time conditions to be able to test the next deformation solution given the recently resolved flow field. This eliminates the possibility of overlapping multiple mesh strains in the same Picard's iteration, which would result in an overestimation and non-physical results of the solid mesh strains. Also, the discrete

fluid flow solution must be interpolated in the newly guessed active fluid mesh after deforming the solid domain.

This iterative scheme results in a decreasing relative error between consecutive coupling iterations, with the subsequent convergence of the numerical method: the scheme is iterated until the L^2-norm of the difference between consecutive Finite Element solutions (of the flow field) is below a certain convergence criteria.

Additionally, a penalization for the solid displacements (see α in equation (4.5)) has been applied to reduce errors in the calculation of the solid domain that would affect the convergence of the problem. This has been mandatory in 2-way FSI problems, where the fluid tractions of the the first iterations could generate over-deformations in the solid.

3.4 Numerical examples

This section demonstrates the application of the proposed methodology to some practical problems. First, the formulation is applied to a falling cylinder in a non-inertial reference frame, where only the inlet velocity is updated, so although the immersed interface does not move, the formulation of our NCMM method is used. Then, the formulation is applied to the two dimensional FSI Turek Benchmark problem [132] where all the FSI-NCMM proposed methodology is applied. The third problem to be solved is the oscillating cylinder with a predefined movement where the interface is updated every time a cylinder movement occurs. The fourth case is the mass-spring system of a cylinder affected by vortex-induced vibration, where the movement of the cylinder, unlike the previous case, is affected by the forces of the fluid. Finally, the numerical techniques are applied to the case of interest, flapping foils. Here is shown the ability of the proposed methodology in the simulation of a technological application. Also, the formulation is applied to a full two-way FSI problem of the free flight of an elastic airfoil. In all cases, the meshes are composed of triangular elements.

3.4.1 Gravity falling cylinder

The first case is an infinitely long cylinder that falls inside a tank due to the gravitational force. As the cylinder moves through the fluid, the location of the immersed boundary would need to be updated each time step. However, this numerical problem considers a non-inertial reference frame, in which the cylinder is fixed and the velocity of the fluid considers its movement. Hence, for the gravity falling cylinder problem a two-dimensional rectangular domain $\Omega = [0, L] \times [0, H] = [0, 4.0]$ m $\times [0, 2.0]$ m with a cylinder diameter $d^s = 0.05$ m located at $[c_x, c_y] = [0.4L, 0.5H]$ is adopted (see Figure 3.2). The density and dynamic viscosity of the fluid are set to $\rho^f = 1.0$ kg/m^3 and $\mu^f = 0.01$ kg/(m$\cdot$ s^2), respectively. The density of the cylinder is $\rho^s = 10.0$ kg/m^3 and the gravity is assumed to act on the horizontal axis as $\boldsymbol{g} = -9.81\hat{\mathbf{i}}$ m/(s^2). The cylinder is considered with a null velocity at $t = 0$. A mesh with $DOF = 41400$ and a constant time step equal to

$dt = 0.005$ are used. The Nitsche parameters are set to $s_1 = 500$ and $s_2 = 100$, while the cut stabilization parameters are $s_G = 0.01$, $s_u = 1.0$, and $s_p = 0.1$. As boundary conditions over the external computational domain, a null manometric pressure and a fixed horizontal velocity $[\boldsymbol{u}_D]_1$ of the fluid are imposed over the walls of the computational domain according to the relative falling motion of the cylinder.

In this sense, the force balance over the solid cylinder on the horizontal axis (where the falling motion occurs), aims to counteract the cylinder's weight mg with the drag force D as $ma = mg - D$. Hence, the numerical integration of the solid's acceleration in order to prescribe it as the inlet flow, as follows:

$$\boldsymbol{u}_D\left(t^{i+1}\right) = \left[[\boldsymbol{u}_D]_1\left(t^i\right) + g\delta t - \frac{D(t^i)}{m^s}\delta t, 0\right]^\top. \tag{3.26}$$

The analytical terminal velocity of the cylinder u_t occurs when $ma = 0$, hence:

$$[\boldsymbol{u}_t]_1 = \left(\frac{\pi}{2}\frac{\rho^s}{\rho^f}\frac{d^s g}{C_D}\right)^{1/2}, \tag{3.27}$$

where the drag coefficient C_D can be defined as a function of the *Reynolds* number and graphically obtained by experimental curves of the cylinder [133]. Therefore, calculating the transient *Reynolds* number $Re_t = d^s u_t \rho^f / \mu^f$ based on the terminal velocity given in equation (3.27), and interpolating a spline that describes the experimental $C_D(Re)$ curve, a simple iterative process is carried out to obtain the analytical terminal velocity and its drag coefficient. These are equal to $u_t = 1.3556$ and $C_D = 4.1927$, respectively.

The time history results of the velocity, acceleration, and drag coefficient of the cylinder are shown in Figure 3.3. These results are compared with the analytical values. Several fluctuations in the velocity are encountered in the numerical simulation, which finally converges to the analytical solution after approximately 6 s, obtaining a maximum relative error of less than 3%. This behavior has been observed in the falling of solid objects in viscous fluids [134].

3.4.2 Turek and Hron FSI Benchmark

The second problem evaluated is the flow past a rigid cylinder with an elastic bar attached to it, which is a well known FSI benchmark for computational fluid dynamics solvers. This problem is based on the Turek's work [132], and it is modeled as a two-dimensional flow inside the rectangular domain $\Omega = [0, L] \times [0, H] = [0, 2.5]\text{m} \times [0, 0.41]$ m, with a cylinder of diameter $d = 0.1$ m located at $[c_x, c_y] = [0.2, 0.2]$ m and a hyperelastic bar of length $l = 0.35$ m and width $w = 0.02$ m attached to the cylinder (see Figure 3.4). The boundary conditions and material properties are defined based on the reported second case (FSI2) in [132]. The fluid density is set to be $\rho^f = 1.0 \times 10^3$ kg/m^3 and the kinematic viscosity is $\nu^f = 0.001$ m^2/s. The Nitsche parameters are set to $s_1 = 200$ and $s_2 = 50$, while the cut stabilization parameters are $s_G = 0.01$, $s_u = 1.0$, and $s_p = 0.1$. The solid

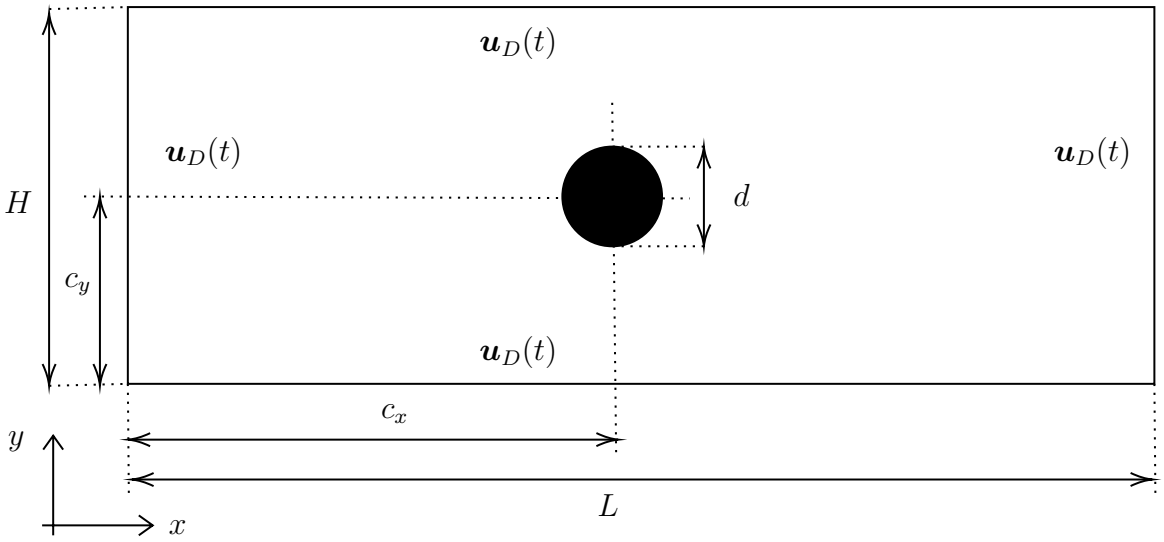
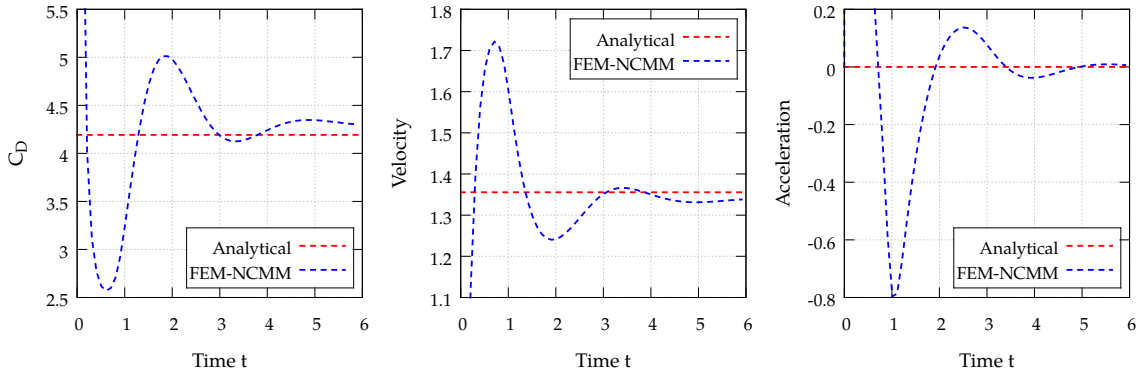


Figure 3.2: Gravity falling cylinder.



(a) **(b)** **(c)**

Figure 3.3: Transient results of the (a) Drag coefficient C_D, (b) velocity, and (c) relative acceleration of the cylinder.

density is $\rho^s = 10 \times 10^3$ kg/m^3, the Poisson's ratio is $\nu^s = 0.4$ and the shear modulus is $\mu^s = 0.5 \times 10^6$ kg/(m$\cdot$ s^2), giving an elasticity modulus of $E^s = 1.4 \times 10^6$ kg/(m$\cdot$ s^2). No-slip boundary conditions are set for the upper and lower walls, as well as for the circle surface and the elastic bar. In the case of the inlet face (left-most wall), the following parabolic profile for the horizontal component of velocity is imposed:

$$u_1^f(0, y) = 1.5\bar{U}\frac{y(H-y)}{(H/2)^2}, \tag{3.28}$$

where, the mean inflow velocity is $\bar{U} = 1.0$ m/s. Additionally, at the inlet wall a null value for the vertical component of velocity is imposed. In the case of the outlet boundary, the tractions are defined as $\boldsymbol{\sigma} \cdot \boldsymbol{n} = 0$, which is a free boundary condition.

The computational problem is addressed by using a background mesh for the fluid with 34788 elements and a foreground mesh with 5457 elements. The time step is δt=0.01 s, and for the first 2 s the problem the solid beam is considered rigid, after that it is free to

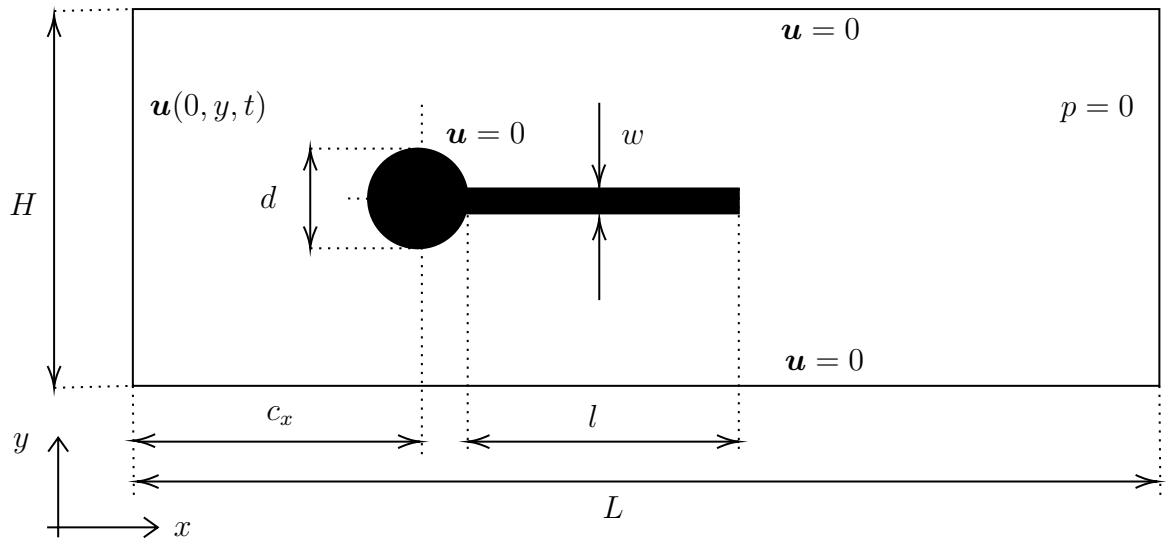


Figure 3.4: Scheme of the boundary condtions for the FSI benchmark Turek.

interacting with the fluid flow. A time instant of the contours of velocity magnitude and pressure for the fluid field are shown in Figure 3.5, where is noticeable the vortex generation downstream the flexible beam. In the same sense three contours of displacements for the solid field in a Y-direction movement cycle (T_{Sy}) is shown in Figure 3.6, displaying the effect of the fluid tractions on the shape of the beam. Figure 3.7 shows the current results compared to the reference case. In Figures 3.7(a)-(b) the displacements of the point $A = (0.6, 0.2)$ are presented, and in Figures 3.7(c)-(d) the results for the lift and drag forces on the solid are shown. Here one sees a big difference between the two results for all the tested variables, we notice that the forces calculated on the surface of the solid are not sufficient for a complete deformation of the solid and, at the same time, this sub-deformation does not disturb the flow of the fluid enough to increase the forces. This configuration of the discrete problem gives the best approximation given the mesh sizes and algorithmic constants. Some findings indicate that some relaxation methods, like Aitken's method in [135], are mandatory when refined meshes are used. Hence, a finer description of the FSI case and the error convergence analysis for smaller mesh sizes are left as future work.

Although the solid problem was validated with the Turek benchmark (CSM2) [132] with a maximum error equal to 4.5% for the solid displacements, there are still some problems that have not been solved when the values of the fluid tractions are prescribed in the solid formulation and that are still under study. In addition, some instabilities are observed in the force graphs due to the residual instabilities in the cuts discussed in the previous sections.

3.4.3 Flow past a transversely oscillating cylinder

The third example is a cylinder that vibrates in the direction perpendicular to the flow. Two scenarios are analyzed here. The first considers the oscillation of the cylinder as

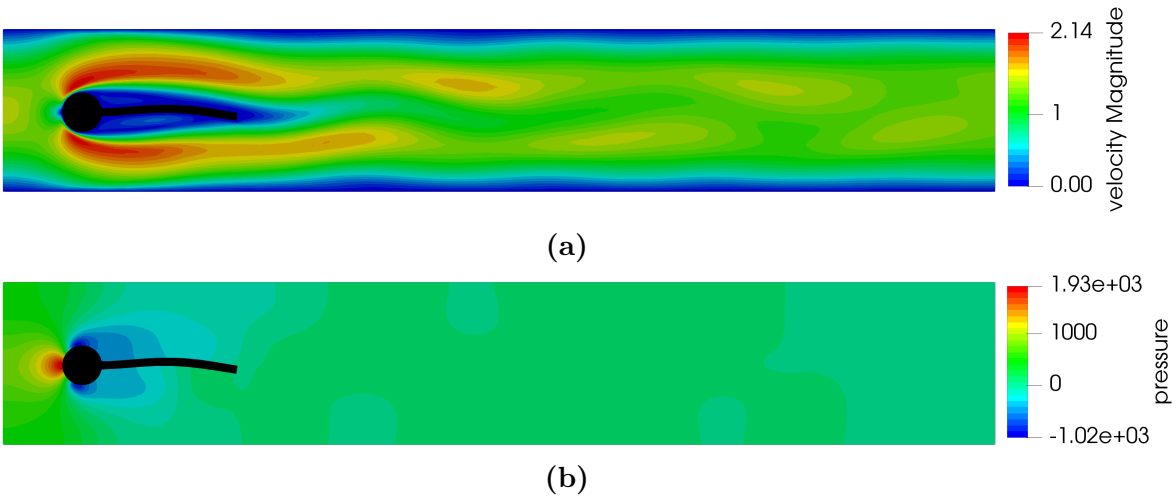


Figure 3.5: Contours of velocity magnitude (a) and pressure (b) for the fluid field of Turek benchmark FSI2

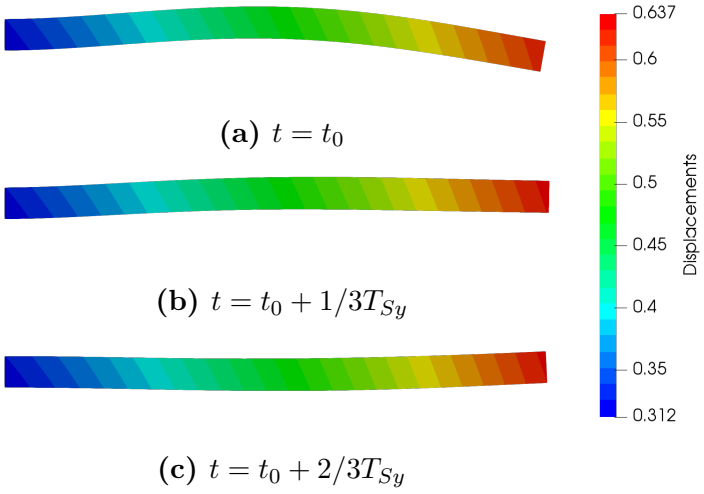


Figure 3.6: Contours of displacement magnitude at different instants for the solid field of the Turek benchmark FSI2.

predefined, while the second computes the vibrations as a result of the balance of forces from the fluid flow and the cylinder spring-mass system. One of the most relevant features is the synchronization, or "*lock-in*", between vortex shedding and vibration frequencies of the body. For the predefined oscillation, this occurs when the vortex shedding (*Strouhal*) frequency f_S of the body equals the imposed oscillation frequency of f_0. Hence, in the first type of simulation this oscillation frequency is imposed and it differs from the vortex shedding frequency f_S obtained in the fixed cylinder case. The second type of simulation has to do with a dynamical system composed by the immersed cylinder subjected to a vertical spring attached in parallel. If the fluid flow is sufficiently inertial, then the cylinder begins to vibrate in the fluid flow. This so-called Vortex-Induced Vibration (VIV) occurs when the flow induces the cylinder to oscillate at a certain frequency f which depends on the flow regime, the cylinder mass, the damping constant and the spring rigidity. These two oscillation scenarios are inside the *lock-in* zone.

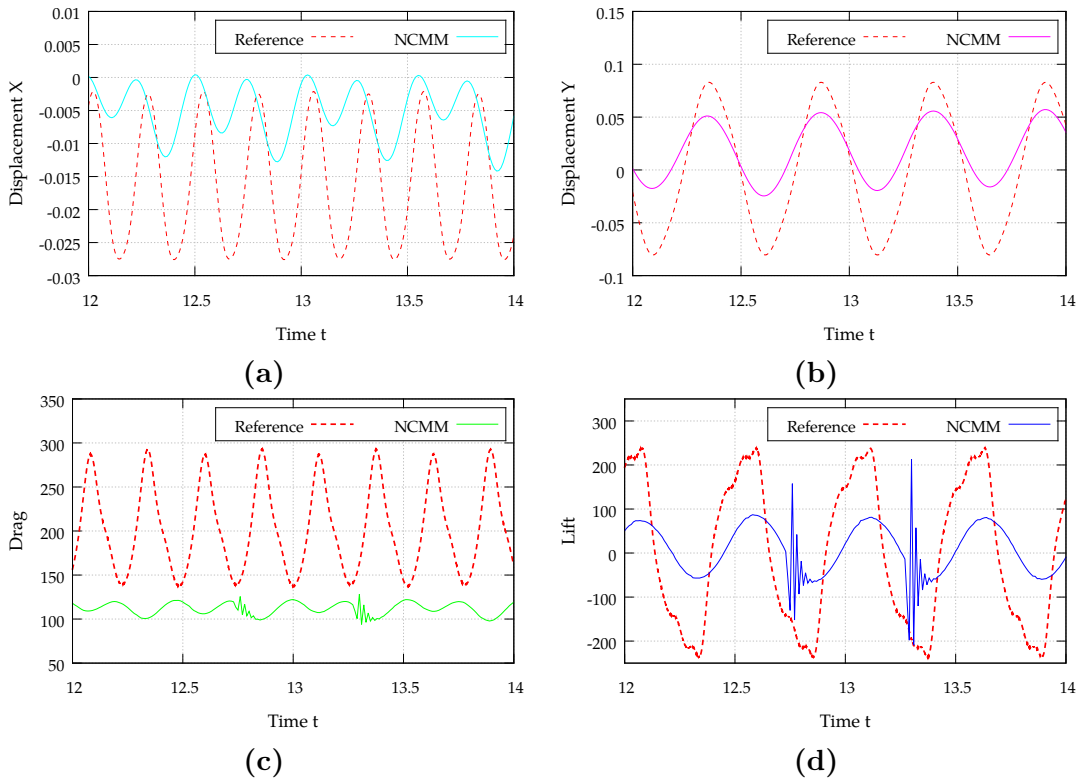


Figure 3.7: Transient results of x (a) and y (b) displacements of the point A, and lift (c) and drag (d) forces on the cylinder and flag

The computational domain must be large enough to avoid a negative effect of the boundaries on the results. It has been concluded in [136, 137] that the wall viscous effects can affect the simulation accuracy. They indicated that the lower the *Reynolds* number, the greater the influence of viscous effects, because the error generated by the artificial boundary conditions perturbs the solution. Therefore, the computational domain is a two-dimensional rectangular domain $\Omega = [0, L] \times [0, H] = [0, 32.5d]\text{m} \times [0, 22d]$ m with the oscillating cylinder initially located at $[c_x, c_y] = [12.5d, 11d]$ m. The boundary conditions of the problem are given as follows: no-slip boundary conditions are set for the upper and lower walls, while the no-slip condition for velocity is weakly prescribed over the circle surface. In the case of the left-most wall, a constant and homogeneous inlet velocity is set. Also, a manometric pressure condition is defined on the right-most (outlet) boundary.

3.4.3.1 Predefined oscillation motion

The first oscillating cylinder case prescribes the motion of the solid cylinder. Hence, the cylinder vibrates independently of the flow traction impinging over its surface. This methodology of rigid bodies with predefined motion inside a fluid is similar to the one in Chapter 2. The predefined vertical $x_2(t)$ motion of the cylinder is considered to be a sinusoidal motion:

$$x_2(t) = h_{max} \sin(2\pi f_0 t), \tag{3.29}$$

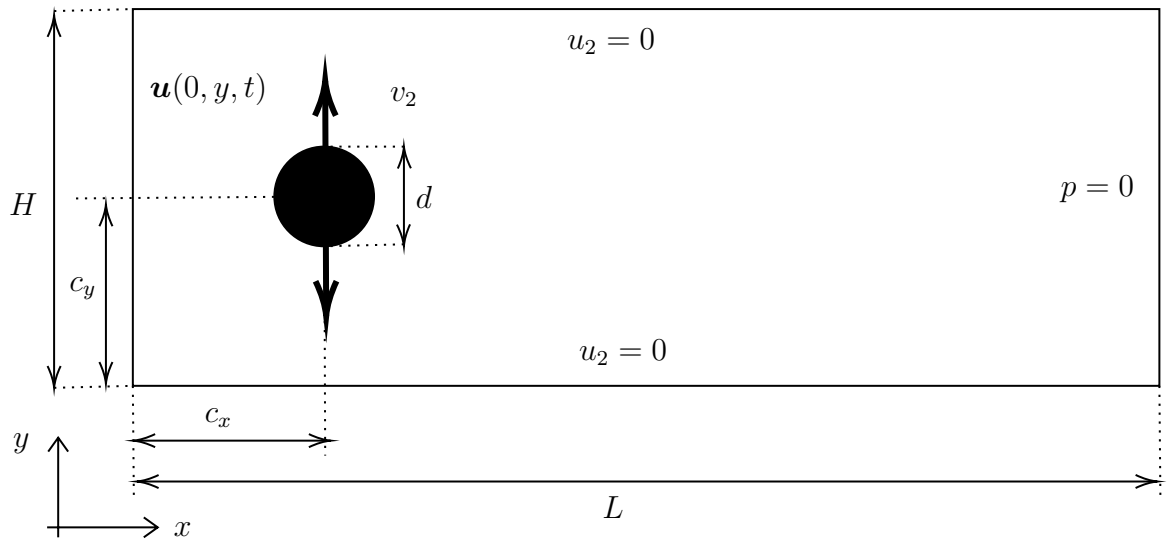


Figure 3.8: Flow past a transversely oscillating cylinder.

where h_{max} is the maximal vertical displacement, and f_0 is the oscillation frequency of the motion of the cylinder. Both h_{max} and f_0 are user-defined parameters. In this case, the values of the spring rigidity, the cylinder mass, or the damping are neglected because the whole movement of the cylinder is only dependent on equation (3.29).

The wake regimes are analyzed using the frequency ratio $F = f_0/f_S$ and the adimensional amplitude $A = h_{max}/d$. This case is evaluated at $Re = 100$ by fixing the cylinder diameter equal to $d = 1.0$ m, the kinematic fluid viscosity of $\nu^f = 0.01$ m/s^2, the fluid density $\rho^f = 1.0Kg/m^3$, and the horizontal component of the inlet velocity equal to $u_1 = 1.0$ m/s. For a fixed cylinder at $Re = 100$, the *Strouhal* frequency is $f_S = 0.166$, which comes from a previously solved transient simulation. The constant adimensional amplitude $A = 0.25$ problem together with a frequency ratio $F = 0.90$ leads to a *lock-in* situation which results in $h_{max} = 0.25$ and $f_0 = 0.1494$. The mesh size gives a total $DOF = 27735$, being radially refined in a homogeneous fashion near the influence zone of the immersed object. The time step is fixed to $\delta t = 0.02$ s. The Nitsche parameters are set to $s_1 = 500$ and $s_2 = 100$, while the cut stabilization parameters are $s_G = 0.01$, $s_u = 1.0$, and $s_p = 0.1$.

The lift coefficient (C_L), the phase portraits, and the frequency of the C_L curve results are plotted in Figure 3.9 and contrasted to the reference solution [136]. The results obtained show similar trends to those of the reference case, however, inaccuracies represented as steep peaks and valleys occur due to small local instabilities that persist despite the application of the ghost penalty. Furthermore, as the interface moves within the fluid mesh, changes in the element cuts are generated, causing abrupt modifications in the shape of the interface (see Figure 2.2 in Chapter 2) that is demonstrated because the peaks and valleys do not achieve perfect symmetry. These instabilities do not globally affect the physics of the problem, but may affect the calculation of forces at specific points on the immersed boundary which for an elastic solid may generate an overestimation of the deformation of the solid. A practical way to reduce these instabilities with the current methodology is to reduce the size of the elements, use larger time steps or increase the

velocities of the immersed boundary to avoid that it remains several time steps in the same element, preventing a scattered calculation of the variables in similar times. The best option to maintain the same reference parameters would be the size reduction of the elements, however, due to the limitation of the computational resources it is not feasible in this work, therefore it will be left for future analysis.

Figure 3.9(a) shows that the $C_{L_{max}}$ is about twice the reference value. This important difference is directly affected by the ratio between the inlet velocity u_1 and the vertical velocity of the cylinder $\partial x_2/\partial t$, which is not specified in the reference, therefore is assumed in this work. The phase portraits in Figure 3.9(b) represents the transfer of energy between the motion of the cylinder and the fluid. A single limit cycle is the result of perfect sinusoidal response, however in our case, slightly different cycles were found, but all similar to the plotted. The irregular shape is due to the sharp oscillations in the C_L curve. In Figure 3.9(c) is plotted the spectrum of frequency for C_L curve, where it is showed that the force is oscillating at the same frequency that the cylinder $f^* = 1$, as reported by the reference. Also, as expected, other small frequencies arise due to the noise generated by the steep peaks and valleys of the C_L curve.

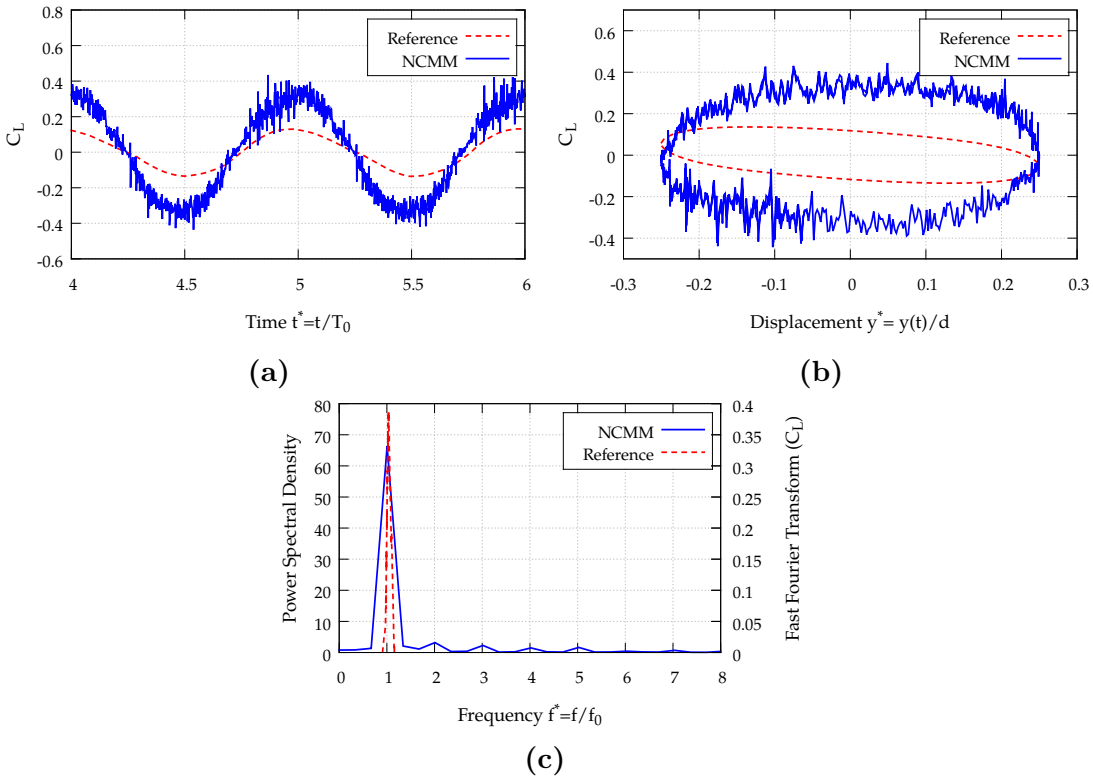


Figure 3.9: Results of (a) Lift coefficient C_L vs normalized time t^*, (b) power spectral density of the C_L vs frequency f^*, and (c) phase portraits of the cylinder movement.

3.4.3.2 Vortex-Induced Vibrations (VIV)

In this second case of oscillating cylinders the movement depends on the vortex released downstream that will lead to fluctuating drag and lift forces that make the cylinder to

oscillate. In this case, the vibration will also be dependent on spring attached to the cylinder with a rigidity ($k \neq 0$), and any damping affecting the cylinder will be neglected ($b = 0$). Cylinder translations in direction of the flow and rotations will also be restricted. Unlike the first case, this is a 2-way FSI problem because the fluid forces are affecting the displacement of the cylinder $x_2(t)$ which, as a result, modifies the flow field and, hence, the fluid forces F_2, and so on. If the flow induces the vibrations over the cylinder, as this case, amplitude and frequency can not be known *a priori*. It is also known [138] that the oscillation amplitude is self-limited to an upper limit. Vertical vibration is governed by the equation of motion:

$$m\frac{d^2x_2}{dt^2} + b\frac{dx_2}{dt} + kx_2 = F_2 \tag{3.30}$$

where m is the system mass, b is the damping, and k the spring constant only acting in vertical direction. In order to compare the obtained results against [136], adimensional parameters for frequency f^*, mass m^*, and rigidity k^* are calculated as follows:

$$f^* = \frac{fd}{\boldsymbol{u}_\infty}, \tag{3.31}$$

$$m^* = \frac{m^s}{1/2\rho^f d^2}, \tag{3.32}$$

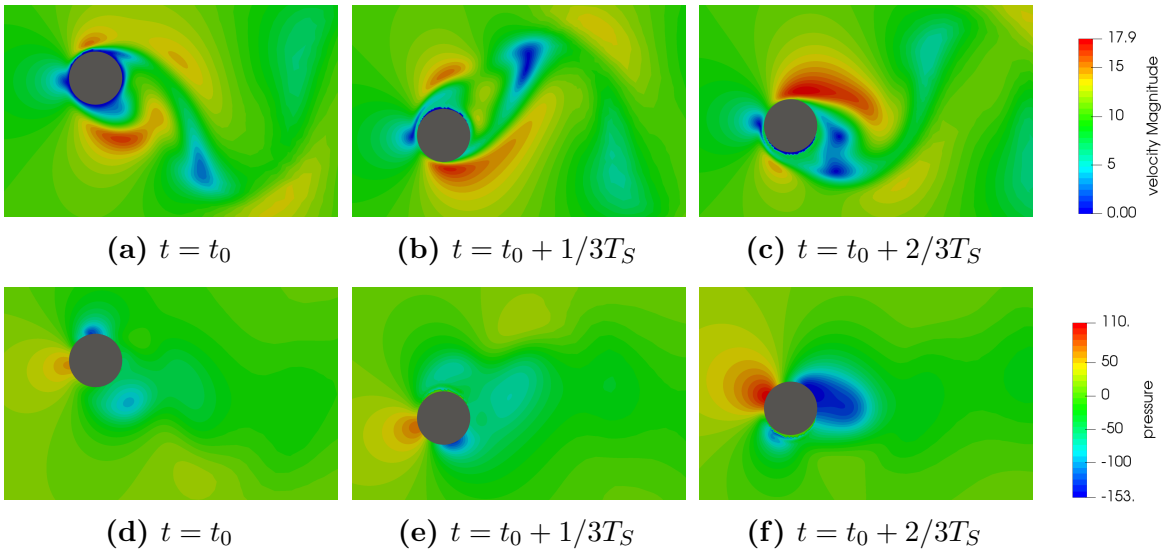
$$k^* = \frac{k}{1/2\rho^f \boldsymbol{u}_\infty^2}, \tag{3.33}$$

where $\boldsymbol{u}_\infty$ is the free stream velocity, m^s the cylinder mass, f is the actual oscillation frequency of the cylinder. Additionally, the "effective rigidity" parameter k^*_{eff} introduced in [139] to fold the influence of k^* and m^* in a single parameter is defined to be:

$$k^*_{eff} = k^* - 4\pi^2 f^{*2} m^* \tag{3.34}$$

For this case, a problem is tested inside the resonant zone observed by [136] in a range of $k^*_{eff} \approx [0, 5]$. The reference case analyzed was taken from [139] and has an effective rigidity equal $k^*_{eff} = 2.14$ which use a non-dimensional elasticity constant $k^* = 9,88$ and a mass $m^* = 5.0$. For a $Re = 100$, we use the kinematic fluid viscosity equal to $\nu^f = 0.1$ m/s^2, and the horizontal component of the inlet velocity equal to $u_1 = 10.0$ m/s. Also, the fluid and solid densities were defined as $\rho^f = 1.0kg/m^3$ and $\rho^s = 3.18kg/m^3$ respectively. Finally, based on the adimensional equations (3.32) and (3.33), the spring rigidity is equal to $k = 494N/m$ and the cylinder mass equal to $m^s = 2.5kg$. For this case the mesh size gives a total $DOF = 116697$. The Nitsche parameters are set to $s_1 = 500$ and $s_2 = 100$, while the cut stabilization parameters are $s_G = 0.01$, $s_u = 1.0$, and $s_p = 0.1$.

The problem was evaluated for $10s$ with a time step $\delta t = 0.01$ s, and during the first $2s$ the cylinder was fixed to avoid high forces due to the instabilities of the first iterations and also to allow the flow to develop the *Von Kármán* vortices downstream of the cylinder. After that the cylinder was free to move in the direction perpendicular to the flow. The velocity magnitude and pressure contours of the cylinder undergoing VIV at different



(a) $t = t_0$ **(b)** $t = t_0 + 1/3T_S$ **(c)** $t = t_0 + 2/3T_S$

(d) $t = t_0$ **(e)** $t = t_0 + 1/3T_S$ **(f)** $t = t_0 + 2/3T_S$

Figure 3.10: Velocity magnitude and pressure contour at different instants of the simulation for the flow past an infinite cylinder undergoing VIV at $Re = 100$.

instants is shown in Figure 3.10. Also, Figure 3.11 shows the graphs over time for the lift and drag coefficients as well as for the displacements. The frequencies of the C_L and C_D curves are also shown. It can be noted that the C_L and C_D curves (see Figures 3.11(a)-(b)) have small unwanted oscillations in their peaks and valleys that are generated by the presence of the immersed boundary several time steps in the same cell as discussed in the previous section. However, they do not affect the displacement behavior of the cylinder (see Figure 3.11(c)) which is smoothed by the spring force. These three graphs also show the proximity of the results obtained to those of the reference case. Despite the numerical instabilities, the main C_L and C_D frequencies (see Figure 3.11(d)) show that the forces on the cylinder are governed by the fluid and solid properties but not by the numerical instabilities.

In addition, Table 3.1 compares the present results with the reference, as we see, the results are close. However as mentioned, they can be improved by correcting the problems mentioned above but also by working with a larger domain that reduces the viscous effect of the walls.

Case	f^*	k^*_{eff}	$C_{L,max}$	$C_{L,W}$	$C_{D,mean}$	A
Reference [139]	0.198	2.140	1.350	-0.020	2.230	0.570
NCMM	0.201	1.864	1.493	0.103	2.230	0.661

Table 3.1: Results for the cylinder undergoing VIV at $Re = 100$.

3.4.4 Flapping foil with prescribed motion

Another relevant case in this study are the flapping wings which are oscillation mechanisms designed to propulsion purposed or power extraction from a fluid flow, by means of

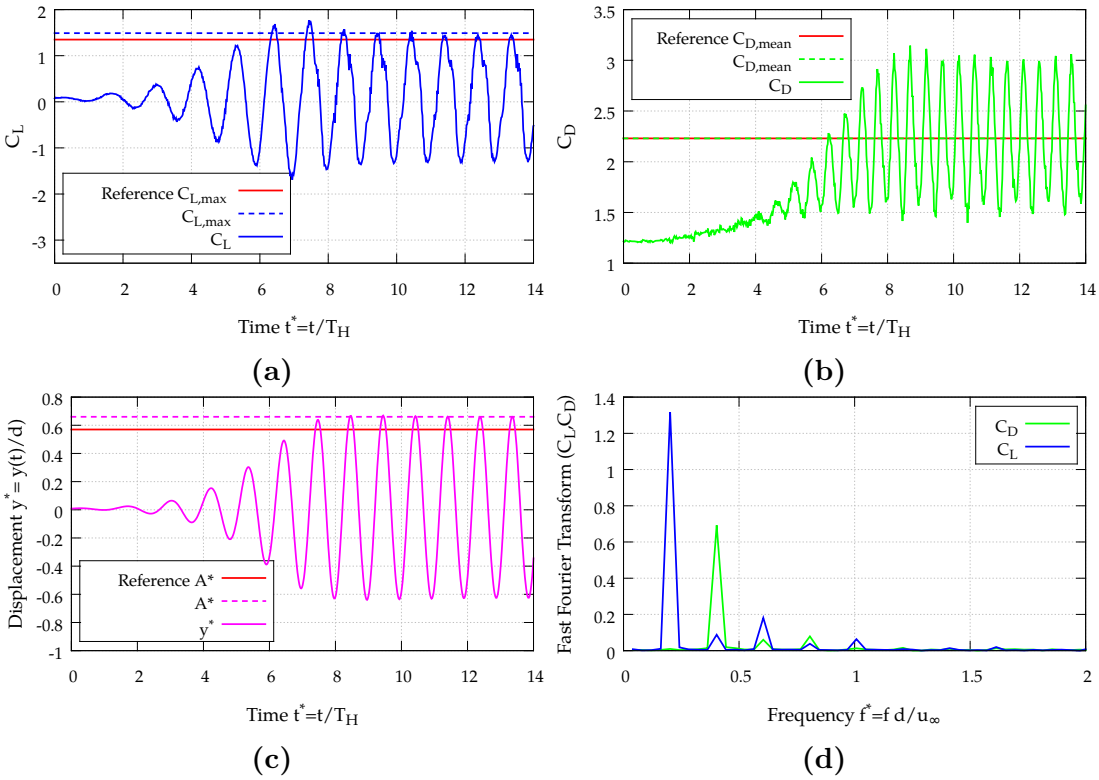


Figure 3.11: Normalized transient results of the VIV for the (a) lift coefficient C_L, (b) drag coefficient C_D, (c) vertical displacement y^*, and (c) frequency f^* vs FFT of the C_L and C_D.

combined oscillatory movements of heaving and pitching of a rectangular wing. Flapping wings are an alternative way to conventional rotational turbines to generate clean energy. The main advantages of these mechanisms over existing turbines is the low impact on wildlife and reduced noise levels due to their relatively low blade tip speed. The numerical approach in this work is applied to analyze the efficiency effect on an oscillating harvesting system in a laminar flow.

The pitching and heaving motions of the airfoil can be passive, meaning that the position of the foil is mainly governed by the fluid forces on the foil, or active, i.e. when all movement is controlled externally. Active control is used when the heaving (vertical translation) $h(t)$ and pitching (rotational) $\theta(t)$ motions of the flapping foil structure are imposed by the equations:

$$\begin{aligned} h(t) &= h_0 \sin(2\pi f_0 t + \phi), \\ \theta(t) &= \theta_0 \sin(2\pi f_0 t), \end{aligned} \tag{3.35}$$

where h_0 and θ_0 are the heaving and pitching amplitudes, respectively, f_0 is the prescribed oscillation frequency, ϕ is the phase difference between the two motions, for which is kept constant at 90° as is recommended in [72] for optimum power extraction performance. Figure 3.12 shows an scheme of the heaving and pitching motions of the flapping airfoil.

The propulsion or power extractions regimes are related to the non dimensional frequency

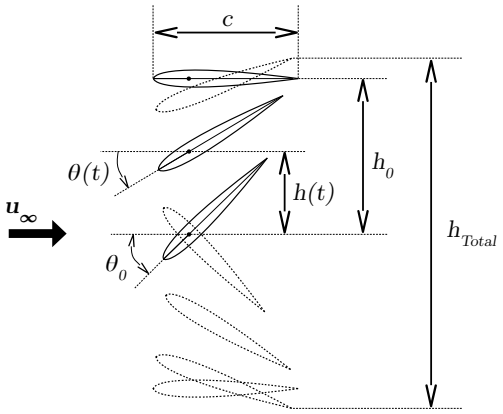


Figure 3.12: Heaving and pitching motions of a flapping wing.

$f^* = f_0 c/u_\infty$, and the pitching amplitude θ_0. Which are summarized in a parameter called *feathering* (χ), which means that when $\chi < 1$ the system is associated with propulsion, and when $\chi > 1$ is related to power extraction [140]. On the other hand, the power extracted by oscillation can be calculated as the sum of the pitching and heaving contributions. The power from the heaving motion $P_2(t)$ is obtained as the product of the heaving force $F_2(t)$ times the heaving velocity V_2 as $P_2(t) = F_2(t)V_2(t)$, while the power from the pitching motion $P_\theta(t)$ is calculated multiplying the pitching moment $M_\theta(t)$ times the angular velocity $\omega(t)$ about the pitching center as $\theta(t) = M_\theta(t)\omega(t)$. Therefore, the total average power extracted over one flapping period T can be integrated as the sum of these contributions:

$$\overline{C}_P = \overline{C}_{P_2} + \overline{C}_{P_\theta} = \int_0^1 \left(C_2(t)\frac{V_2(t)}{u_\infty} + C_M(t)\frac{\omega(t)c}{\boldsymbol{u}_\infty} \right) d(t/T), \tag{3.36}$$

where $C_2(t) = F_2(t)/(0.5\rho^f c\boldsymbol{u}_\infty^2)$ and $C_M = M_\Theta(t)/(0.5\rho^f c^2\boldsymbol{u}_\infty^2)$ are the power coefficients of heaving and pitching respectively. If $\overline{C}_P > 0$, the foil captures the kinetic energy available in the fluid flow, but, if $\overline{C_P} < 0$ means the foil is providing energy to the fluid. The total energy harvesting efficiency η for an oscillating foil is defined as:

$$\eta = \eta_2 + \eta_\theta = \frac{\overline{P}_2 + \overline{P}_\theta}{1/2\rho\boldsymbol{u}_\infty^3 h_{Total}} = \overline{C}_P \frac{c}{h_{Total}} \tag{3.37}$$

which is the ratio between the average total power extracted and the total power available in the fluid area in which the foil moves. And where h_{Total} is the overall vertical extent of the foil motion.

A 2D flapping foil in a laminar flow for a *Reynolds* numbers equal to 1100 is set following the reference [140]. The fluid field is defined much smaller than the reference due to the computational resources, therefore, some inaccuracies arose due to the viscous effect from the walls. A rectangular two-dimensional domain $\Omega = [0, L] \times [0, H] = [0, 10c] \times [0, 7c]$ is set as the computational domain with a NACA0015 airfoil of chord length $c = 1.0$ located at $[c_x, c_y] = [0.35L, 0.5H + c]$ at a time $t = 0$ (see Figure 3.13). The boundary conditions of the oscillating airfoil sets no-slip boundary conditions for the upper and

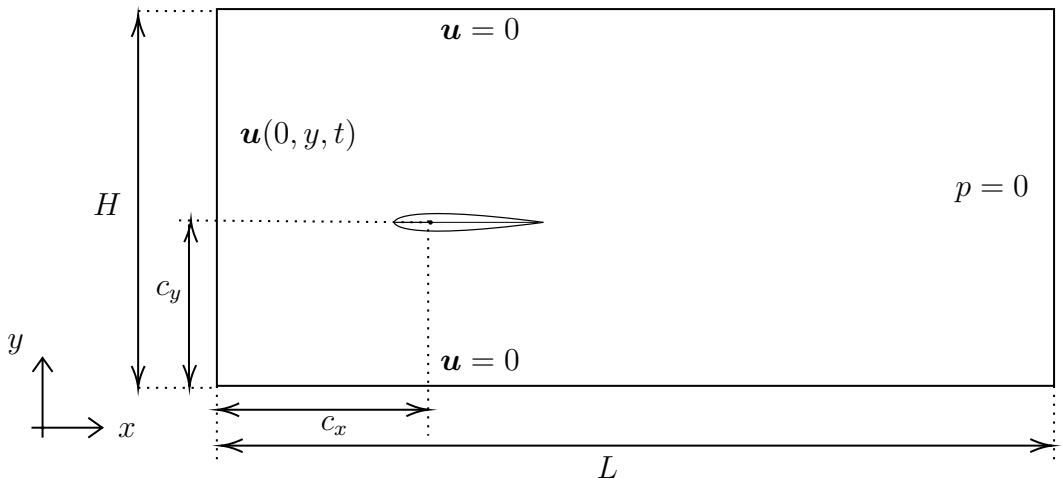


Figure 3.13: Flapping airfoil scheme.

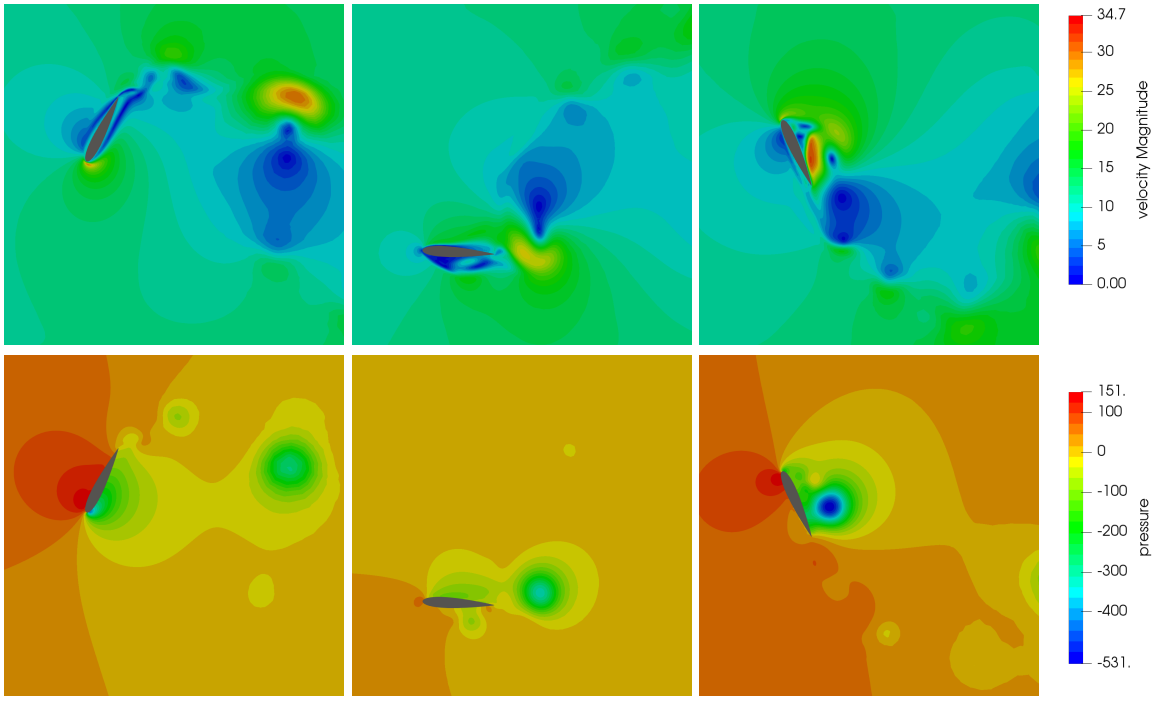
lower walls, as well as for the foil surface. In the case of the inlet face (left-most wall), a constant velocity $u_2 = 11.0$ is imposed, while for the outlet boundary, the tractions are defined to be $\boldsymbol{\sigma} \cdot \boldsymbol{n} = 0$. The amplitude is $h_0 = c$, the pitching amplitude is $\Theta_0 = 76.3^o$ and the center of pitching rotation x_p is at $c/3$ from the leading edge. The prescribed oscillating frequency is $f_0 = 1.54$, which is equivalent to a non-dimensional frequency $f^* = 0.14$. The kinematic viscosity is $\nu^f = 0.01$. The time step was equal to $\delta t = 0.005$ during $T = 5$ so that a stabilization of the fluid flow periodicity was achieved. The Nitsche parameters are set to $s_1 = 1000$ and $s_2 = 500$, while the cut stabilization parameters are $s_G = 1.0$, $s_u = 1.0$, and $s_p = 0.1$. The velocity magnitude and pressure contours of the airfoil oscillating for a time cycle T at different instants is shown in Figure 3.14. Figure 3.15 shows the C_D, C_L and C_{P_2} curves over time for the reference case and those obtained by our model. It can be seen that the current numerical results show trends to those of the reference case for all plotted curves, however, there is still a substantial difference between the two results. This is mainly due to the lack of fluid elements in the airfoil area, which was not not enough to accurately capture the shape of the airfoil due to limited computational resources, which also restricted the size of the domain. For this case it can be noted that the momentum and the power coefficient by the pitching movement are neglected, which represents less than 5% of the total contribution to the power. Table 3.2 shows the efficiencies of the two cases compared.

Case	$\overline{C}_{P_2}$	$\overline{C}_{P_\Theta}$	$\overline{C}_P$	$\eta_2(\%)$
Reference	0.80	0.06	0.86	30.9
NCMM	0.88	–	–	33.9

Table 3.2: Efficiency and power coefficcient for flapping foils.

3.4.5 Free flight airfoil

The last problem is the simulation of the free flight of an airfoil that is similar to a two-dimensional kite with no attachment points. In this problem the computational

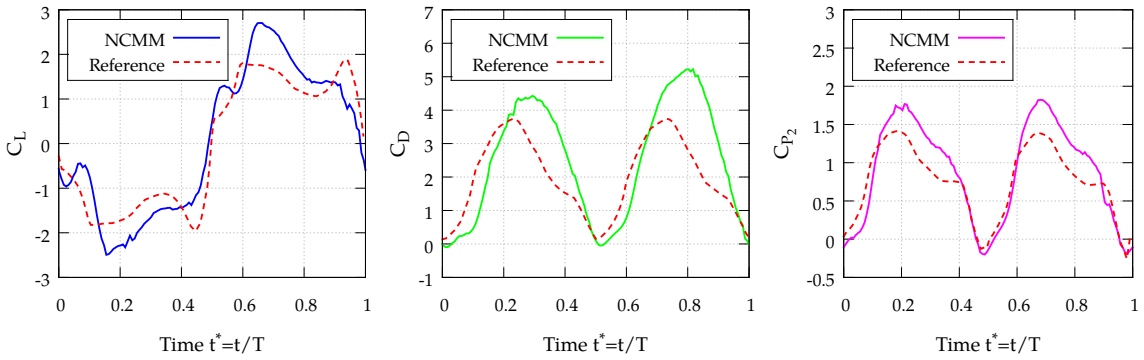


(a) $t = t_0$ **(b)** $t = t_0 + 1/3T$ **(c)** $t = t_0 + 2/3T$

Figure 3.14: Velocity magnitude and pressure contour at different instants of the simulation for the flow past an infinite aifoil flapping at $Re = 1100$.

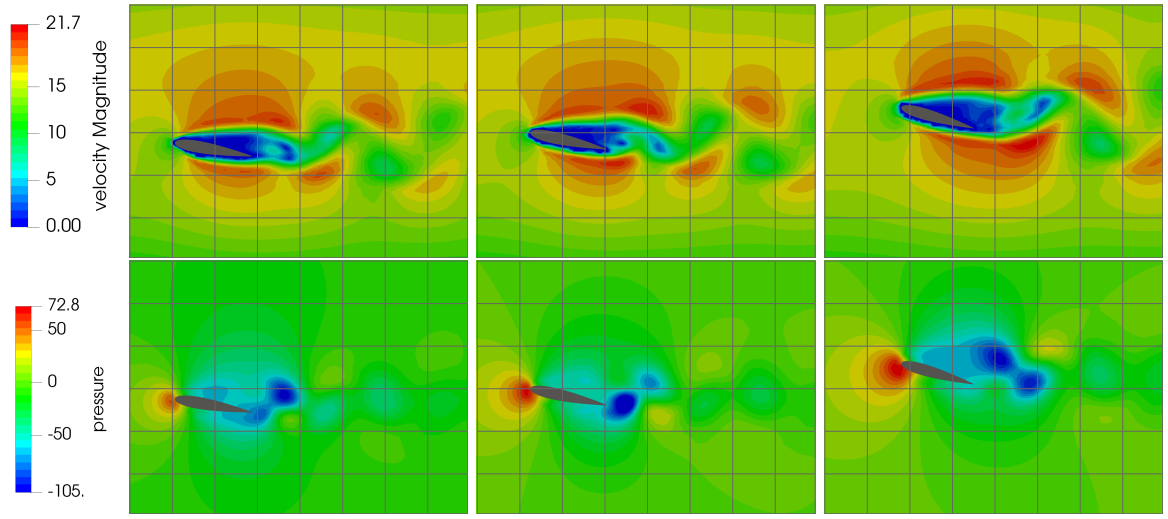
domain and boundary conditions are like the case of the flapping foil (see Figure 3.13). The difference here is that the displacements of the airfoil are not controlled by heaving and pitching functions but by fluid forces acting on the elastic airfoil.

The rectangular two-dimensional domain is $\Omega = [0, L] \times [0, H] = [0, 10c] \times [0, 7c]$ an the NACA0015 airfoil of chord length $c = 1.0$ m is located at $[c_x, c_y] = [3c, 3c]$ with angle of attack $\alpha = 10^o$ at a time $t = 0$ s. No-slip boundary conditions are set for the upper and lower walls, as well as for the foil surface. The inlet face imposed a parabolic velocity profile with a mean velocity $u_{mean} = 11.0$ m/s, while for the outlet boundary, the tractions are defined to be $\boldsymbol{\sigma} \cdot \boldsymbol{n} = 0$. The dynamic viscosity is $\mu^f = 2.0 \times 10^6$ Kg/(m·s) and the density is $\rho^f = 1.0$ Kg/m^3. The solid density is $\rho^s = 10^3$ Kg/m^3, the Poisson's ratio is $\nu^s = 0.3$ and the shear modulus is $\mu^s = 2.0 \times 10^6$ kg/(m· s^2). The airfoil was analyzed during a time $T = 5$ s with time steps equal to $\delta t = 0.01$ s. The Nitsche parameters are set to $s_1 = 1000$ and $s_2 = 500$, while the cut stabilization parameters are $s_G = 1.0$, $s_u = 1.0$, and $s_p = 0.1$. Figure 3.16 shows the velocity magnitude and pressure contours for different time instants during the free flight of the airfoil. The displacement of the airfoil due to the aerodynamic forces can be observed in those figures, where a rectangular grid is printed on the contours to allow the visualization of the movement of the airfoil in the different instants of time. Additionally, Figure 3.17(a) shows the curves of the displacements and rotation of the airfoil over time, and Figure 3.17(b) plots the lift and drag forces in the same time range.



(a) **(b)** **(c)**

Figure 3.15: Normalized transient results of the flapping airfoil at $f^* = 0.14$ of (a) lift coefficient C_L , (b) drag coefficient C_D, (c) power coefficient.



(a) $t = 1$ s **(b)** $t = 3$ s **(c)** $t = 5$ s

Figure 3.16: Contours of velocity magnitude and pressure results for the free flight airfoil for three instants of time.

3.5 Conclusions

This chapter has demonstrated the application of the proposed FSI methodology in solving complex fluid problems. This FSI methodology uses a FEM approach for both the incompressible Navier-Stokes equations and the equations of non-linear hyperelastic solids. A stabilized VMS formulation is used to overcome Galerkin's instability problems in the fluid flow equations. This methodology was based on fixed mesh methods, where the fluid and solid meshes overlap and the solid boundary is tracked by the fluid domain as a sharp interface. In addition, the Neumann-Neumann boundary conditions are used for the coupling, with fluid tractions imposed in one direction and solid displacements applied by the Nitsche method in the other direction.

The methodology has been tested on problems for oscillating cylinders with predefined motion and Vortex Induced Vibration (VIV), with satisfactory results close to reference

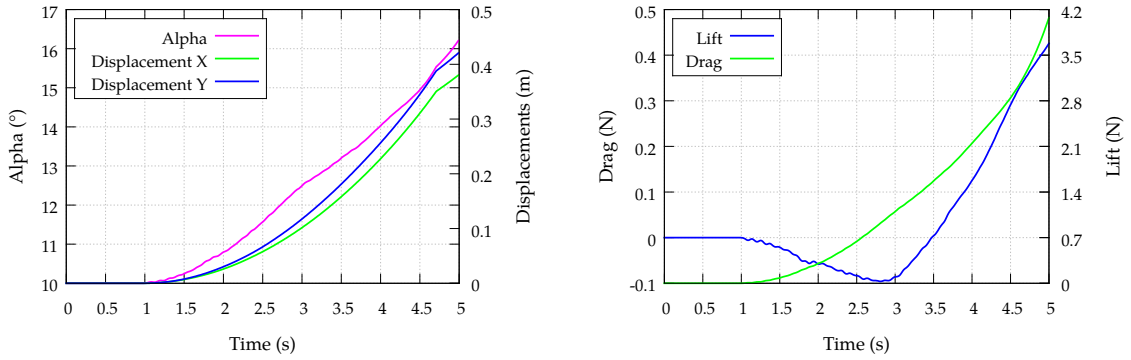


Figure 3.17: Angle of attack and displacements vs time (a), and lift and drag forces vs time for the airfoil (b).

values. In the same sense, a flapping foil has been tested with similar results. In all these cases the solid was considered as a rigid body, however, the methodology was also tested on the Turek 2-way FSI benchmark, where solids and fluids resolve completely, requiring iterative coupling at the interface. The results show the ability of the method to solve the physics of the problem, however, the underestimation of the displacements and forces shows some issues to be solved. In the same order, the residual instabilities of the ghost penalty also need to be controlled in the solution.

Finally, from an engineering perspective, this methodology can be successfully applied to more complex FSI problems in order to develop new technologies or improve them, such as flapping foils with flexible parts, which will be analyzed in future work.

Choose initial guess: $\boldsymbol{U}_h^{-1} = \boldsymbol{U}_h^0, \boldsymbol{d}_h^{-1} = \boldsymbol{d}_h^0$;
for *unsteady iterations:* $n = 0$ *to* N **do**
  Assign $\boldsymbol{U}_h^{n+1,c} = \boldsymbol{U}_h^n, \boldsymbol{d}_h^{n+1,c} = 0$;
  for *coupling iterations:* $c = 0$ *to* C **do**
    if $c = 0$ **then** use the solid's mesh of the previous time step $\boldsymbol{x}_h^{n+1,c+1} = \boldsymbol{x}_h^n$;
    Restore previously undeformed solid's mesh $\boldsymbol{x}_h^{n+1} = \boldsymbol{x}_h^{n+1,c+1} - \boldsymbol{d}_h^{n+1,c}$;
    for *solid's non-linear iterations:* $i = 1$ *to* I **do**
      Solve solid's displacements $\boldsymbol{d}_h^{n+1,i+1}$ from (3.22) by imposing the tractions $\boldsymbol{t}_h^s$;
      Calculate error of non-linear solution $\epsilon_{solid} = \|\boldsymbol{d}_h^{n+1,i+1} - \boldsymbol{d}_h^{n+1,i}\|_2$;
      if $\epsilon_{solid} < \mathrm{Tol}_{solid}$ **then**
        Exit solid loop;
      else
        Update previous non-linear solid solution: $\boldsymbol{d}_h^{n+1,i} = \boldsymbol{d}_h^{n+1,i+1}$;
        Keep iterating;
      end
    end
    Calculate solid's deformation $\boldsymbol{d}_h^{n+1,c+1} = \boldsymbol{d}_h^{n+1,c} + \beta^s(\boldsymbol{d}_h^{n+1,i+1} - \boldsymbol{d}_h^{n+1,c})$;
    Deform solid's domain $\boldsymbol{x}_h^{n+1,c+1} = \boldsymbol{x}_h^{n+1} + \boldsymbol{d}_h^{n+1,c+1}$;
    Update the set of covered fluid elements $\boldsymbol{T}_h^c$ and the interface Γ^{f-s};
    Interpolate $\boldsymbol{U}_h^{n-1}, \boldsymbol{U}_h^n$, and $\boldsymbol{U}_h^{n+1,c}$ in the new active fluid mesh Ω^f;
    for *fluid's non-linear iterations:* $i = 0$ *to* I **do**
      Solve the fluid flow field $\boldsymbol{U}_h^{n+1,i+1}$ from (3.24) by weakly imposing the Dirichlet conditions over Γ^{f-s};
      Calculate error of non-linear solution $\epsilon_{fluid} = \|\boldsymbol{U}_h^{n+1,i+1} - \boldsymbol{U}_h^{n+1,i}\|_2$;
      if $\epsilon_{fluid} < \mathrm{Tol}_{fluid}$ **then**
        Update coupling from non-linear solid solution: $\boldsymbol{U}_h^{n+1,c+1} = \boldsymbol{U}_h^{n+1,i+1}$;
        Exit fluid loop;
      else
        Update previous non-linear fluid solution: $\boldsymbol{U}_h^{n+1,i} = \boldsymbol{U}_h^{n+1,i+1}$;
        Keep iterating;
      end
    end
    Project $\boldsymbol{U}_h^{n+1,c+1}$ onto $\boldsymbol{T}_h \setminus \boldsymbol{T}_h^c$ and calculate $\boldsymbol{\sigma}_{h\setminus c}^{f,n+1,c+1}$ over this subset;
    Project (Extrapolate) $\boldsymbol{\sigma}_{h\setminus c}^{f,n+1,c+1}$ at the solid surface Γ^{f-s} to obtain $\boldsymbol{t}_h^s$;
    Calculate error of coupling $\epsilon_{coupling} = \|\boldsymbol{U}_h^{n+1,c+1} - \boldsymbol{U}_h^{n+1,c}\|_2$;
    if $\epsilon_{coupling} < \mathrm{Tol}_{coupling}$ **then**
      Update fluid's transient solutions: $\boldsymbol{U}_h^{n-1} = \boldsymbol{U}_h^n, \boldsymbol{U}_h^n = \boldsymbol{U}_h^{n+1,c+1}$;
      Update solid's transient solutions: $\boldsymbol{d}_h^{n-1} = \boldsymbol{d}_h^n, \boldsymbol{d}_h^n = \boldsymbol{d}_h^{n+1,c+1}, \boldsymbol{x}_h^n = \boldsymbol{x}_h^{n+1,c+1}$;
      Exit coupling loop;
    else
      Update previous coupling solution: $\boldsymbol{U}_h^{n+1,c} = \boldsymbol{U}_h^{n+1,c+1}$;
      Keep iterating;
    end
  end
end

Algorithm 1: FSI coupling scheme.

4 Simulation of the *Triplaris Americana* seed flight using multiple approaches for the solution of the aerodynamic fluid-structure interaction

Abstract

In this work, the auto-rotation of the *Triplaris* seed during its free-fall flight is evaluated by two dimensional simulations. In particular, the fluid-structure interaction that occurs between the seed and its environment when it is carried by the air. To directly solve this problem, a computational fluid-structure interaction framework which includes the Finite Element approximation over Non-Conforming Overlapping Meshes of the incompressible Navier-Stokes equations and the Non-linear hyperelastic solid equations for the seed deformation is implemented. In the first part of this chapter, the numerical methods that have been used to deal with the challenges that arise in the fluid-structure simulations are presented. Among the most important challenges that have been addressed: the implicit solution of the dynamic equations coupling and the algorithmic developments to deal with a collection of colliding meshes are recalled. In the second part, a two-dimensional characterization of the cross profile of the *Triplaris Americana* and an improved Blade Element Momentum Theory analysis for non-planar actuator disks representing the leaves are used to evaluate its aerodynamic performance. Predictive information for such phenomena is developed based on the simulations.

Highlights

- A two-dimensional simulation to analyze the cross-section of the leaf of the auto-rotating *Triplaris Americana* seed.
- A Blade Element Momentum theory modified is used to analyze non-planar rotating geometries like the seed.
- An overlapping domain method for two-dimensional FSI problems for describing the rotation of the seed.
- The Variational Multi-Scale method is used to solve the incompressible flow.
- An hyperelastic model is used to solve the elastic deformation of the seed.

4.1 Introduction

Tree-leafed seeds are naturally designed to move away from their origin. These aerodynamic seeds move away from the parent plant, since the new plant might not get the adequate conditions (sun, water or nutrients from the soil) being near the grown specimen. The flight is the most common mechanism in nature to disperse (move) the seeds, but others like water, explosions or animals, are also common [141]. A schematic of the seed falling is presented in Figure 4.1.

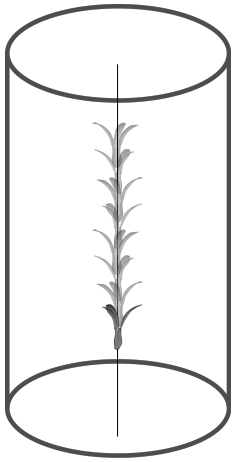


Figure 4.1: Representation of the seed falling.

Few efforts have been given to investigate the flight of Tree-leafed seeds from the mechanic's point of view. That can be explained by the complexity of the fluid and structural mechanics involved: the incompressible Navier-Stokes equations together with the structural response of the seed portray as the mathematical model that completely describes the flight. Indeed, this problem is similar to the aeroelastic description of wings, which is studied for developing wind turbines or aircraft. The balance between the structural resistance and aerodynamic loads for the turbine blades design, i.e., light and strong blades with high aerodynamic efficiency, is one related problem [91]. Since no analytic solutions are found for the complex flight phenomena present in flexible wings, the engineering field has relied on numerical methods.

But the computational approach is still challenging. Since the Fluid-Structure Interaction (FSI) problem involves the solution of the fluid flow and the deformation of solids, it is a challenging problem. The kinetic and kinematic fields of both media are tightly interconnected, where the deformations of the solid modify the flow domain, and the tractions of the fluid at the interface stress the solid. In most cases, the interaction is fully dynamic, and none steady state can be reached. To achieve the complete description of the flight, it is necessary to perform an iterative analysis that involves the solution of fluid dynamics and structural simulations, which require time and computational resources. Most descriptions [94, 93] have not taken into account the complex three-dimensional geometry of the seed nor its transient behavior, which leads to neglect the important dynamics of flight. The numerical approximation of the incompressible Navier-Stokes equations represents an active research topic itself in computational mechanics, and its interaction with a deforming solid's structure makes it even harder. Moreover, the possibility to apply the numerical approximation of transient fluid-structure problems onto a predictive model (that brings clues to the description of the flight) is emerging nowadays.

Ongoing numerical simulations intended to describe the flight of seeds has been mostly dedicated to the *Maple* seed (see for example the references [86, 87, 36, 88, 89]). In those simulations, the flight is described as the flow past the rigid solid body; interactive

mechanisms related to the fluid-structure interaction have been, therefore, not relevant in those approaches. Instead, the seed's wings are commonly hyperelastic whose lift is driven by the wing's deformation, so that, the duration of the flight is tightly related to the complete description of the deforming solid. Experimental studies [82, 83, 84] have been used as a source of inspiration in the development of flight vehicles and wind turbines [80, 81].

From an analytical point of view, blade momentum and blade element momentum theory have been the classic approaches to evaluate fluid-dynamics rotational components, and have also been used to evaluate the *Maple* seed as in [82], or [142] which evaluates *Maple*-like *Mahogany* seed. In addition, other kinematic and dynamic descriptions attempt to analyze the aerodynamic response of this seed in the air [85].

A more complicated approach is the numerical solution of the fluid flow equations around the wing profiles, named as the direct numerical simulation of turbulence. Direct numerical simulations (of the turbulence generation and dynamical deformation) during the seed's flight have been only attempted in [36, 87, 86]. An Immersed Boundary Method was used in those works to approximate the incompressible flow inside a simplified domain of the *Maple* seed flight. But the major drawback of those numerical methods is the limitation to Cartesian meshes, and therefore, to simple geometries that differ from the actual mechanisms of the seed's flight.

The interest in the present work is the two dimensional simulation and analysis of a three-leafed seed with autorotation properties called *Triplaris Americana*, which uses the rotation effect to fly away from the parent plant and maximize its reproductive possibilities. We achieve this study by numerically approximating the incompressible flow and the seed's hyperelastic deformation with the actual conditions of the falling trajectory. This work may serve in understanding the relevant physics of seed's flight: due to the cumbersome of *in situ* measurements of the airflow past the seed, the numerical simulation of the seed's flight becomes an alternative option. The particular characteristics of this seed have also been used to design a bio-inspired a wind turbine [90, 91].

In this chapter, recently developed numerical ingredients that overcome the various numerical issues related to the FSI problem related to flight are included. The present methodology solves the solid's domain immersed in the fluid's background. Principally, through the Finite Element Method (FEM), as it is useful in describing complex non-cartesian geometries and the multiple coupled physics (i.e. fluid-structure). Firstly, a Fixed Mesh Method and a Non-Conforming Mesh Method (NCMM) are applied between the solid and the fluid discrete domains. NCMM are suitable for analyzing problems that involve large deformation and great changes in the structure topology, mainly because the fluid's mesh does not need to follow the fluid-solid interface; therefore, there is no fluid's mesh deformation or mesh reconstruction. Secondly, the Variational Multi-Scale (VMS) method is implemented to stabilize the numerical solutions for convection-dominated problems or when equally interpolation spaces are used for the different variables of the fluid's problem. In terms of the discrete stabilized formulation, the weak imposition of the fluid's essential conditions using the Nitsche method at the FSI interface is crucial to

define the no-slip conditions of the fluid over the moving solid boundary. Cut elements are stabilized by using the Ghost Penalty method [64]. The approach to deal with the fluid and structure coupling is done through an implicit coupling which is detailed in Chapter 3. Finally, the algorithmics is implemented in the FEniCS library, where multi mesh methods are optimized by means of Computational Geometry methods and automated partial differential equations solution [143, 144].

This chapter is organized as follows. In Section 4.2, the methodology that is implemented to numerically simulate the seed's flight is recalled. The computational model of the *Triplaris Americana* seed and the flight problem setting are first presented. Then, the numerical approach for solving the FSI-FEM problem in the overlapping domains are described. In Section 4.3, the numerical results are presented. A complete numerical study of the mechanisms that generate the lift in the realistic Triplaris seed flight is carried out. Some concluding aspects of the direct numerical simulation of tree-leafed seed flight problems with the present computational approach are stated in Section 4.4.

4.2 Methodology

In this section, the process of how to achieve a feasible simulation of the seed flight is commented. First, the characteristics and properties of the *Triplaris* seed are presented. Then, the FSI problem and the detailed description of the numerical strategies (to overcome the difficulties typically encountered for numerically simulating FSI) are presented. Finally, the procedure to achieve the numerical simulation is described.

4.2.1 Aerodynamic and mechanical properties of the *Triplaris* seed

Triplaris Americana is a genus of plants in the family *Polygonaceae*, with about 25 species distributed in America. The *Triplaris* seed is one of the multiple seeds with autorotation properties that we can find in nature [145, 141].

The *Triplaris* has three radially arranged leaves connected to a bulb, when it falls from the tree and interacts with the air creates auto-rotation effects that allow it to travel more time in the air before landing in the ground. In Table 4.1 there is a summary of the characteristics of the *Triplaris*. An important issue of the simulation is the complex three-dimensional geometry of the seed. The main difficulty is to represent in a computational environment the complexity of the seed's shapes and tissues.

Radius (cm)	2.15
Weight (g)	0.08
Falling speed (m/s)	1.13
Angular speed (RPM)	1322
Average Reynolds number	600

Table 4.1: Characteristics of the *Triplaris Americana* seed.

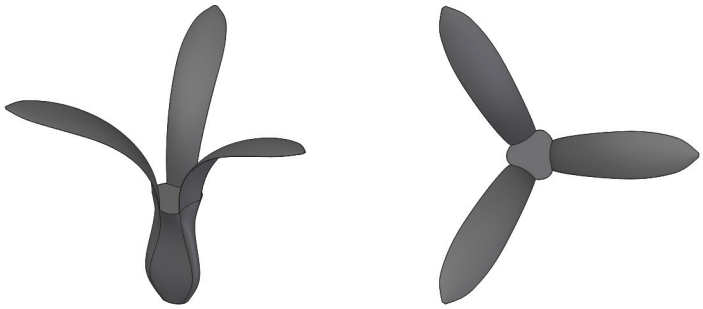


Figure 4.2: 3D model of *Triplaris Americana* seed.

The seed CAD model used in this work (and display partially in Fig. 4.2) is a clean copy of a 3D imaging, specifically from a three-dimensional scanner, made in [146], and a geometry characterization of the seed developed in [147]. This pre-processed model eliminates leaf veins and assumes a symmetry in the distribution of leaves that are arranged every 120 degrees. About the 50% of the *Triplaris* seeds analyzed in [92] had this symmetric configuration. Besides, any distortion or malformation of the leaves is omitted, and all three leaves are drawn the same. However, the main characteristics of the seed, such as the curvature of the leaves or the bulb shape, are preserved. These simplifications are necessary to avoid information transfer problems in the numerical simulation when the seed and fluid meshes overlap. As no information regarding the mechanical properties of the seed is available, the averaged properties of different dry sheets from [148, 149] are used as reference values. In this sense, the Poisson's ratio is set to $\nu^s = 0.5$ and the elasticity modulus is given by $E^s = 33.4 \times 10^6$ kg/(m$\cdot$ s^2). These properties give a shear modulus of $\mu^s = 11.1 \times 10^6$ kg/(m$\cdot$ s^2). Additionally, the density of the seed is set to $\rho^s = 241$ kg/m^3, which has been calculated based on the weight of the seed (see Table 4.1) and its CAD-obtained volume.

For the sake of the aerodynamic characterization of the seed, the cross profile of the leaves is simplified from the three-dimensional model. Figure 4.3 shows the simplified airfoil shape form the cross-section of the *Triplaris* leaves. This seed foil has a maximum camber of 9.2% and a thickness $t/c = 0.021$.

4.2.2 Fluid Structure Interaction problem

The complex fluid and structure mechanics related to the seed's flight are completely described by the incompressible Navier-Stokes equations coupled with the hyperelastic

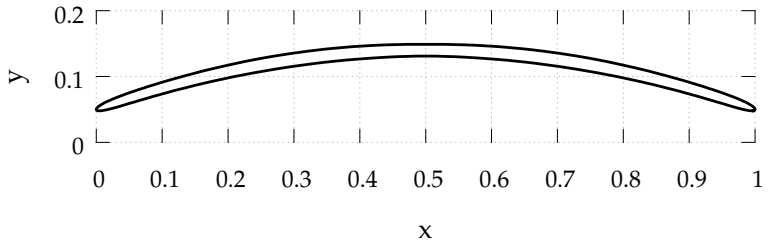


Figure 4.3: Airfoil from the cross section of the *Triplaris Americana* leaves.

solid equations. Considering the fluid domain $\Omega^f \subset \mathbb{R}^2$ and the immersed solid domain $\Omega^s \subset \mathbb{R}^2 | \Omega^s \in \Omega^f$. The fluid-structure interface is denoted as $\Gamma^{f-s} = \partial\Omega^s$. Here we use superscripts f and s to denote the fluid and the solid, respectively.

4.2.2.1 Fluid flow problem

The fluid flow is defined in the fluid domain Ω^f where $\boldsymbol{n}^f$ is the unit outward normal to the boundary Γ^f and $(0, T)$ is the time interval. The strong form of the incompressible Navier-Stokes equation is such that, given the prescribed velocity $\boldsymbol{u}_d$ in the essential boundary Γ^f_d and the prescribed traction $\boldsymbol{t}^f_n$ in the natural boundary Γ^f_n, find the velocity $\boldsymbol{u} : \Omega^f \times t \in (0, T) \to \mathbb{R}^2$ and the pressure $p : \Omega^f \times t \in (0, T) \to \mathbb{R}$ satisfying

$$\begin{aligned} \rho^f \left(\partial_t \boldsymbol{u} + \boldsymbol{u} \cdot \nabla \boldsymbol{u}\right) - \mu \Delta \boldsymbol{u} + \nabla p = \boldsymbol{f}^f \quad & \text{in } \Omega^f, t \in (0, T), \\ \nabla \cdot \boldsymbol{u} = 0 \quad & \text{in } \Omega^f, t \in (0, T), \\ \boldsymbol{u} = \boldsymbol{u}_d \quad & \text{on } \Gamma^f_d, t \in (0, T), \\ \boldsymbol{\sigma}^f \cdot \boldsymbol{n}^f = \boldsymbol{t}^f_n \quad & \text{on } \Gamma^f_n, t \in (0, T), \\ \boldsymbol{u} = \boldsymbol{u}_0 \quad & \text{in } \Omega^f, t \in (0, T), \end{aligned} \tag{4.1}$$

where, ρ^f is the density of the fluid, μ^f is the dynamic viscosity, and $\boldsymbol{f}^f$ is the body force. The Cauchy stress tensor is calculated as $\boldsymbol{\sigma}^f = 2\mu \nabla^{sym} \boldsymbol{u} - p\boldsymbol{I}$, where $\nabla^{sym}(\cdot) := \nabla(\cdot) + (\nabla(\cdot))^\top$ is the symmetric gradient operation and $\boldsymbol{I}$ is the identity tensor.

4.2.2.2 Hyperelastic solid problem

The solid deformation problem is such that, given the prescribed displacement $\boldsymbol{d}_D$ in the essential boundary Γ^s_D and the prescribed traction $\boldsymbol{t}^s_N$ in the natural boundary Γ^s_N, find the displacement $\boldsymbol{d} : \Omega^s \times t \in (0, T) \to \mathbb{R}^2$, such that:

$$\begin{aligned} \rho^s \partial_t^2 \boldsymbol{d} - \nabla \cdot \boldsymbol{P}(\boldsymbol{d}) = \boldsymbol{f}^s, \quad & \text{in } \Omega^s, t \in (0, T), \\ \boldsymbol{d} = \boldsymbol{d}_D \quad & \text{on } \Gamma^s_D, t \in (0, T), \\ \boldsymbol{P}(\boldsymbol{d}) \cdot \boldsymbol{n}^s = \boldsymbol{t}^s_N \quad & \text{on } \Gamma^s_N, t \in (0, T), \\ \boldsymbol{d} = \boldsymbol{d}_0 \quad & \text{in } \Omega^s, t \in (0, T), \end{aligned} \tag{4.2}$$

where $\boldsymbol{n}^s$ is the unit outward normal to the boundary $\partial\Omega^s$, ρ^s is the density of the solid, $\boldsymbol{f}^s$ is the body force on the solid, and $\boldsymbol{P}(\mathbf{d})$ is the first Piola-Kirchhoff stress tensor of the solid.

The first and second Piola-Kirchhoff stress tensor for a hyper-elastic material can be defined as:

$$\boldsymbol{P}(\boldsymbol{d}) = \boldsymbol{F} \cdot \frac{\partial W(\boldsymbol{E})}{\partial \boldsymbol{E}}, \qquad \boldsymbol{S}(\boldsymbol{d}) = \frac{\partial W(\boldsymbol{E})}{\partial \boldsymbol{E}}, \tag{4.3}$$

respectively, where $W : \mathbb{R}^{2\times 2} \to \mathbb{R}^+$ is a given strain-energy density, $\boldsymbol{F} = \boldsymbol{I} + \nabla \boldsymbol{d}$ is the deformation gradient, $\boldsymbol{E} = \frac{1}{2}(\boldsymbol{C} - \boldsymbol{I})$ the Green-Lagrange strain tensor, and $\boldsymbol{C} = \boldsymbol{F}^T \cdot \boldsymbol{F}$ the right Cauchy-Green deformation tensor. One of the simplest hyperelastic models is the St. Venant-Kirchoff for a nonlinear, elastic, and isotropic material, which defines the second Piola-Kirchoff stress as

$$\boldsymbol{S}(\boldsymbol{d}) = 2\mu^s \boldsymbol{E} + \lambda^s \mathrm{tr}\,(\boldsymbol{E})\,\boldsymbol{I}, \tag{4.4}$$

where $\mu^s = \dfrac{E^s}{2(1+\nu^s)}$ and $\lambda^s = \dfrac{E^s \nu^s}{(1+\nu^s)(1-2\nu^s)}$ are the Lamé coefficients given as functions of the Young modulus E^s and the Poisson ratio ν^s. Hence, the first Piola-Kirchhoff stress tensor is computed from this last expression by contracting it with $\boldsymbol{F}$.

4.2.2.3 Fluid-Structure coupling

The coupling between fluid Ω^f and solid Ω^s domains can be done in several ways. The preferred approach in this work is to use Neumann-Neumann conditions. It is explained because the moving solid boundary randomly cuts the fluid mesh triggering the numerical difficulty of prescribing Dirichlet conditions over the fluid nodes. Hence, those are imposed weakly by adding terms to the discrete flow equations inside the fluid domain. At the fluid-structure interface Γ^{f-s} the velocity of the fluid and solid coincide, while the tractions must be equal and opposite:

$$\begin{aligned} \alpha\,(\boldsymbol{u} - \boldsymbol{u}^s) = 0 \quad &\text{on } \Gamma^{f-s}, t \in (0,T), \\ \boldsymbol{\sigma}^f \cdot \boldsymbol{n}^f + \boldsymbol{P}(\boldsymbol{d}) \cdot \boldsymbol{n}^s = 0 \quad &\text{on } \Gamma^{f-s}, t \in (0,T), \end{aligned} \tag{4.5}$$

where $\boldsymbol{u}^s = \partial_t \boldsymbol{d}$ is the velocity of the solid boundary, α is a penalization parameter, and $\boldsymbol{n}^f = -\boldsymbol{n}^s$.

4.2.3 Numerical Strategy

The incompressible Navier-Stokes equations together with the hyperelasticity problem presented before, portray as the mathematical model that is used to describe the complex fluid-structure interaction problem that occurs during the seed's flight. These equations

are too complex that it is impossible to obtain an analytical solution for the transient character of the flow involved in the aeroelastic setting. Numerical methods remain to be the practical approach when one aims to approximate the solution of these equations.

4.2.3.1 Finite element approximation

The straightforward Galerkin approximation of the Navier-Stokes equation above presents instability issues for convection dominant flows or when equal interpolating spaces are used for the pressure and velocity variables, to what is referred to as the *inf-sup* condition. In this work, the Variational Multi-Scale (VMS) finite element formulation of the incompressible equations that has been studied in [150] is adopted. This globally stabilized finite element method combines the possibility of describing complex geometries with high order interpolations, and it is accurate in multi-physics applications. Another numerical ingredient of this stabilized formulation is the weak imposition of no-slip boundary conditions at the solid interface. This is motivated by the diffuse localization of the interface with respect to the fixed fluid mesh nodes. In this line, the weak imposition of the no-slip conditions for the fluid at the moving solid interface is achieved by means of the Nitsche's method. Nevertheless, bad cut elements at the interface (with a small portion of fluid) need to be treated to avoid instabilities in highly convective flows. These local instability issues are generated by the small support of their convection terms in the stiffness matrix, therefore, the lack of control may lead to oscillations in the solution near to the interface. To this end, the importance of stabilizing the Fluid FEM formulation for the cut cells is related to the correct transfer of information between fluid and the solid boundary. In the present work, the Ghost Penalty (GP) method [64] is adopted as a local stabilization complement to the VMS stabilized fluid flow formulation, which is defined by including some jump penalties acting on inter-element faces [151].

The design of the stabilization method leads to an accurate description of transient incompressible flows. In particular, including the VMS global stabilization, the weak imposition of no-slip boundary conditions through the Nitsche's methods, and the local stabilization of bad cut elements at the moving interface by the GP method lead to an accurate fluid flow solver for fixed mesh methods. Algorithmically, stabilization parameters for all the VMS, Nitsche, and GP methods have been taken from [113, 24].

In the case of the hyperelastic solid, the Galerkin method is applied straightforwardly in the numerical solution of the solid problem. This method is capable of correctly describing the solid deformations when the fluid tractions are prescribed on the solid boundary.

Since both the fluid and the solid problems are non-linear problems, a Newton numerical scheme is adopted to solve each non-linear problem. An implicit second-order BDF scheme for the fluid flow problem and the second-order central differential scheme for the solid problem are used to discretize the time derivatives of those dynamical formulations.

4.2.3.2 FSI treatment

In the discrete approach, an overlapped mesh method using two different meshes is implemented, to what is referred as to *multi mesh* finite element methods. Especially, the solid's mesh overlaps arbitrarily the fluid's elements at the background mesh (see Figure 4.4). These two meshes together define the computational domain in which the FSI problem is solved. In this work, the efficient computational geometry and algorithmic approach of two and three-dimensional mesh-mesh intersections in FEniCS library [130] are adopted. The implementation in FEniCs relies on efficient generation and traversal of Axis-Aligned Bounding Box Trees (AABB trees). This methodology allows for low-level operations for computing intersections of triangles and tetrahedra and generation of quadrature points on cut cells.

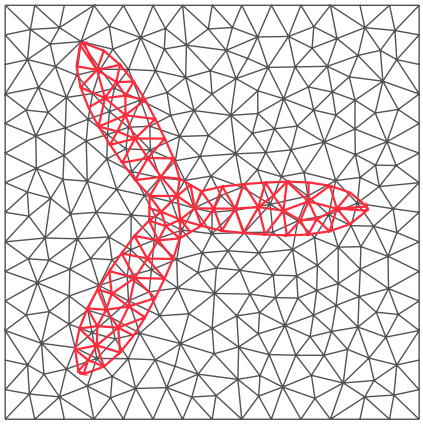


Figure 4.4: Scheme for the overlapping meshes.

An implicit partitioned approach is used to integrate in time, such that the interface coupling must balance between the fluid and the solid solutions at each time step. As described in the preceding section, Neumann-Neumann coupling conditions are implemented due to the impossibility of strongly defining the displacement conditions at the interface. In this regard, a Picard's scheme at each time step is adopted to deal with the implicit coupling of the transfer conditions: at each Picard's iteration, the fluid flow is solved considering the weak imposition of no-slip conditions on the fluid elements next to the interface. Also, the fluid flow solution is stabilized by the global VMS formulation and the Ghost Penalty technique at the cut elements. Then, the fluid flow tractions are prescribed to deform the solid domain. The iterative process advances until an FSI coupling convergence criteria is satisfied. Then, the next time step is solved for both the fluid and the solid problems.

Another FSI issue has to do with the inclusion of the fully covered fluid solution in the stress computation near the interface. Hence, the present approach addresses this issue by permanently updating the set of cut and covered fluid elements in the multi mesh finite approach. In practice, this is done by recalculating the overlapping part of the fluid mesh each time the solid mesh deforms and creating a sub mesh contained in the background fluid mesh. Projections of fluid stresses from this space to the solid boundary

is done by means of a L^2-projection onto the solid finite space of stresses. This avoids the transmission of overestimated stresses to the solid boundary, which may also cause non-physical deformations in the solid domain.

Finally, the restoration of the deformed mesh in case of not satisfying the traction balance at the interface at each Picard's iteration is another key development of the present methodology. This eliminates the possibility of overlapping consecutive mesh deformations in the Picard's iterative scheme, which would result in an overestimation and non-physical results of the mesh strains. Also, care must be taken with the discrete fluid flow solution at each guess solid deformation since it must be interpolated in the newly guessed active fluid mesh after deforming the solid domain.

The previously exposed numerical strategies have been validated in several fluid-structure interaction cases. The variational multi-scale finite element formulation of the incompressible Navier-Stokes equations with the weak imposition of boundary conditions has been presented in Chapter 2 and extended to the solution of fully FSI problems in Chapter 3. The selected approaches for overcoming most of the numerical challenges arising in the FSI problems are explained in detail in that last chapter.

4.3 Results

This section presents an in-depth analysis of the aerodynamics of the seed. First, an aerodynamic characterization of the cross-section of the *Triplaris Americana* seed leaf is presented and compared with other airfoils. Then, an analytical and stationary model of the seed is described and applied using a classical theory to analyze rotational blades. Finally, the two-dimensional FSI simulation using the overlapping meshes approach is presented.

4.3.1 Aerodynamic characterization of the cross-sectional profile of the *Triplaris Americana* leaf

The seed profile is characterized by solving a two-dimensional flow past the leaf. Figure 4.3 shows the simplified airfoil shape form the cross-section of the *Triplaris* leaf that is used in this analysis. The computational domain Ω^f is defined as a circular domain of diameter $D = 10c$ with the center of the simplified airfoil chord coinciding at the center of the circular domain, as schematically shown in Figure 4.5. As boundary conditions of the problem, the velocity inlet conditions for the left half of the circumference are known, while $\boldsymbol{\sigma} \cdot \boldsymbol{n} = 0$ tractions are defined for the right half of the circumference, which is a free boundary condition. In this first analysis, the flow is simulated using a single fixed mesh for the fluid, such that no-slip boundary conditions are fixed strongly for the velocity at the airfoil surface.

On average, the seed falls with a *Reynolds* number of 600, but the flow stream can impinge on the airfoil at very different angles. Hence, several angles of attack α between $[0°, 40°]$

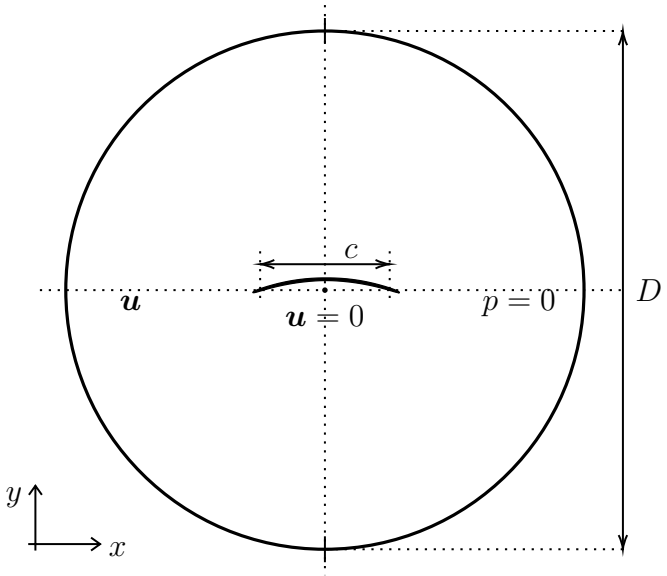


Figure 4.5: Flow past the cross section of the *Triplaris Americana* seed leaf.

are simulated at this flow condition to characterize the airfoil performance. This transient analysis of the continuous variation of the attack angle is achieved by modifying the components of the inlet velocity through the simulation. Two seed airfoil shapes with different edges are compared: one considers sharp leading and trailing edges, while the second round up those edges. These two airfoil cases are selected because they represent the extreme configurations that could naturally occur in the cross-section geometry of the leaves of the seed. Note that the shape of the seed has been represented with rounded edges in Figure 4.3.

Depending on the case, the meshes in these simulations are composed of around 34000 triangular elements, which are arranged in an unstructured fashion with a refinement close to the surface of the airfoil. The time step is chosen to be $\delta t = 0.1$ s. The C_D vs α, C_L vs α, and L/D vs α curves for these two profile configurations are presented in Figure 4.6 for the $Re = 600$ condition. It can be observed that the foil with rounded edges demonstrates a wider α range of high L/D ratios which makes it more efficient and less susceptible to changes, especially in the $\alpha = [10°, 25°]$ range.

Moreover, to analyze the performance of these two geometries, they can be compared against both a flat plate and the NACA4702 profile. Figure 4.7 shows the L/D ratio curves for the 600 and 1000 *Reynolds* numbers, respectively. It can be seen that the L/D ratio increases with the *Reynolds* in all cases, as it is expected. Also, that the NACA4702 profile shows a slightly better performance than the seed foil with rounded edges. However, the NACA4702 demonstrates a rapidly descending L/D curve in higher α values. It is also observed that the efficiency seed profile does not improve as much as that of the NACA4702 with the increase in *Reynolds* number, which shows that this seed foil is specially adapted to work at very low *Reynolds* numbers. It is worth mentioning that the oscillating behavior that appears on most curves for $\alpha > 20°$ can be explained by the occurrence of stream vortices that are generated by the flow detachment on the upper

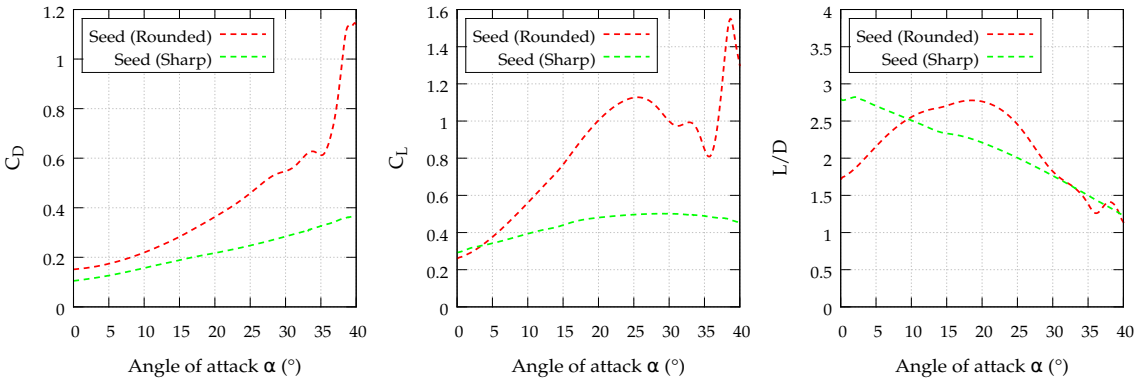


Figure 4.6: Seed airfoil aerodynamic results at $Re = 600$ and different α attack angles: (a) C_D , (b) C_L , and (c) L/D.

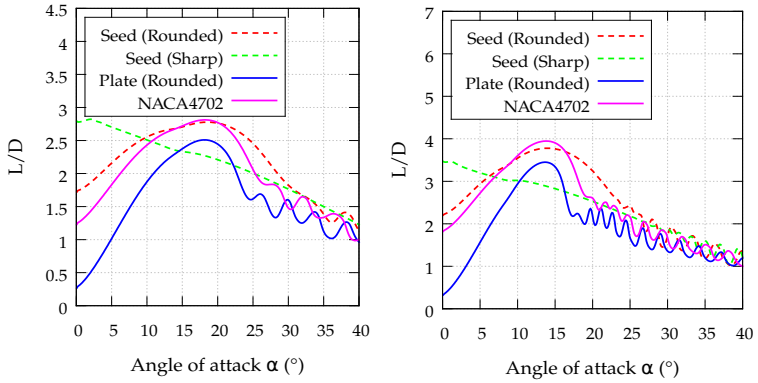


Figure 4.7: Aerodynamic L/D results of different airfoils operating at several α attack angles and (a) 600 and (b) 1000 *Reynolds* numbers.

airfoil surface.

4.3.2 Analytical model of the *Triplaris* seed flight

An analytical analysis of the flight of the seed can be performed by using a stationary aerodynamic model known as the Blade Element Momentum Theory (BEMT). Although new complex and semi-empirical analytical models have been developed to better predict blade performance, BEMT remains a baseline wind turbine and propeller analysis method due to its robustness and accurate estimation of the actual performance of the analyzed geometry. The classical BEMT has been developed to analyze geometries which are included in the same plane of rotation. However, due to the curvature of the *Triplaris Americana* seed leaves, it is necessary to modify the classical BEMT and adopt the novel version in [152] that allows conical blade geometries; the modified theory includes a geometric parameter that defines the ψ angle between the planar disc and the conical surface generated by the rotation of the deformed blade. In the present case, this ϕ angle is used to describe the curvature of the leaves. One of the major limitations of BEMT is

the inability to consider the three-dimensional effects that can underestimate the results, as it will be discussed later.

In the classic analysis of momentum theory, rotational blades are idealized as an actuating disk through which the free-stream velocity U_∞ impinges with a decreased velocity at the disk position U_{Disk}. This condition is expressed as:

$$U_{Disk} = U_\infty(1 - a), \tag{4.6}$$

where the axial induction factor $a < 1$ is the fraction of the decrease in wind speed between the free-stream and the rotor plane. Additionally, when the wake rotation is included in the momentum theory, a similar analogy for the modified angular impinging velocity of the airflow at the rotating disk must be considered. Hence, the angular induction factor a' is introduced in the theory, which is defined as the ratio between the angular velocity of the impinging flow stream and the angular velocity of the rotor Ω. Therefore, the tangential velocity U_T on the rotor position is calculated with:

$$U_T = \Omega r(1 + a'), \tag{4.7}$$

where r is the local radius of a section of the blade. As the angular momentum theory is applied together with the blade theory, the resulting methodology is called BEMT. The BEMT divides each blade into $i = 1, 2, ..., N$ sections of length l_i, where the mean section is located at a radius r_i from the rotor shaft. The stream velocities and exerted forces are calculated in each section i, and then integrated over the entire blade to calculate the rotor thrust and torque.

Indeed, the modified BEMT analysis in [152] which is suitable for non-planar actuating disk geometries is applied here to analyze the *Triplaris* seed autorotation. In this sense, Figure 4.8(a) shows a side view of the seed leaf that is segmented in different sections and which includes the ψ_i angle defining the incidence of the free-stream flow over each section of the leaf. The velocity U_{Disk} can be divided into two components based on the ψ_i angle: the normal component $U_{N,i}$, and the radial component $U_{R,i}$. The normal component of velocity is the effective component of the free-stream velocity that is included in the BEMT calculation on each section of the leaf.

For non-planar geometries, the normal and tangential velocities defined above must be particularly considered in a separate balance. These are calculated at each section i by means of the $\cos\psi_i$ quantity:

$$U_{N,i} = U_\infty(1 - \mathrm{a}_i)\cos\psi_i, \tag{4.8}$$

$$U_{T,i} = r_i\Omega(1 + \mathrm{a}'_i)\cos\psi_i, \tag{4.9}$$

and the relative velocity $U_{rel,i}$ at each section i is defined as:

$$U_{rel,i} = \sqrt{U_{N,i}^2 + U_{T,i}^2}. \tag{4.10}$$

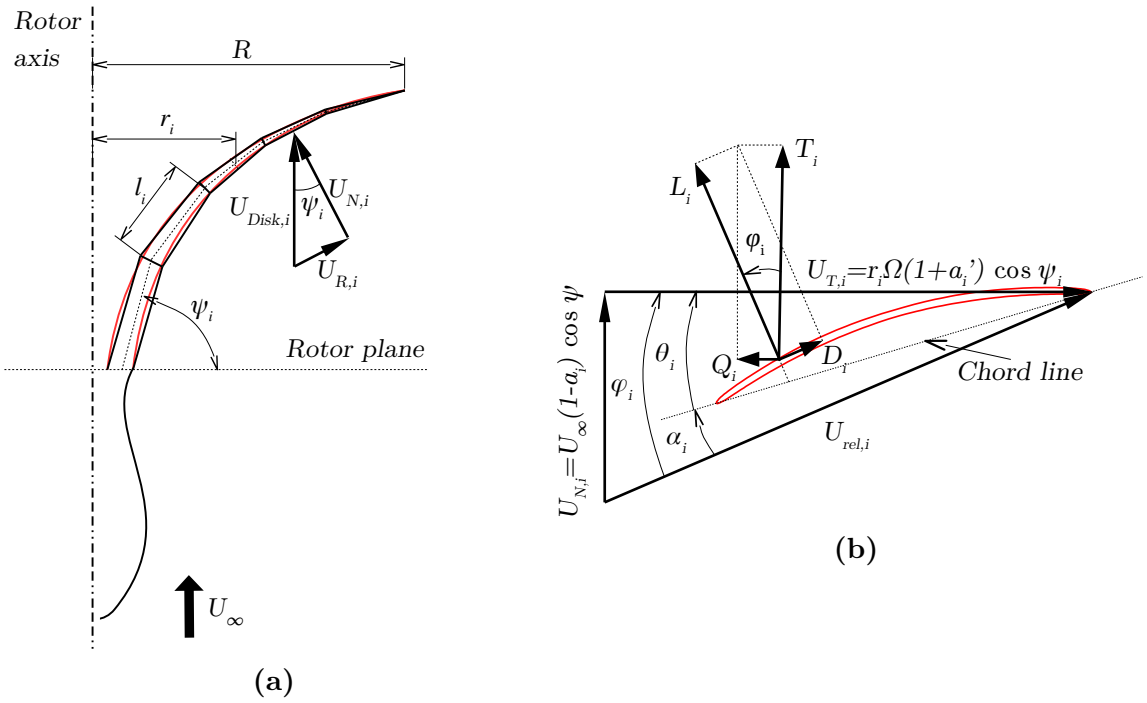


Figure 4.8: BEM leaf seed model: (a) segmentation in several blade sections, and (b) velocity and force diagrams for a given radial leaf station.

In the same way, the inflow angle ϕ_i is defined with the normal and tangential velocities by:

$$\phi_i = \tan^{-1}\left(\frac{R}{r_i}\frac{(1-a_i)\cos\psi_i}{\lambda(1+a_i')}\right), \tag{4.11}$$

where λ is the tip speed ratio defined as $\Omega R/U_\infty$, and R is total radius of the leaf. The angle of attack α_i can be computed from $\alpha_i = \phi_i - \theta_i$ (see Figure 4.8(b) for a schematic depiction), where θ_i is the pitch angle of each section.

Finally, the BEMT theory is only complete when the induction factors a_i and a_i' are calculated in terms of the variables of the aerodynamic problem. By equating the thrust expressions that can be obtained independently of blade theory and momentum theory by balancing the energy of the system, an expression for the axial induction factor can be defined as follows:

$$a_i = \frac{2S_{w,i} + F_i - \sqrt{F_i^2 + 4S_{w,i}F_i(1-F_i)}}{2(S_{w,i}+F_i^2)}, \tag{4.12}$$

where $F_i < 1$ is the tip loss factor and $S_{w,i}$ is:

$$S_{w,i} = \left(\frac{\sigma_i'}{4}\right)\frac{C_{T,i}\cos\psi_i{}^2}{\sin\phi_i{}^2}, \tag{4.13}$$

σ_i' is the local solidity of the seed, defined as $Bc_i/2\pi r_i$, B is the number of blades, c_i the mean chord, and $C_{T,i}$ is the local thrust coefficient that in this case is described by the

Glauert model [153] as:

$$C_{T,i} = 4a_i \left(1 - \frac{1}{2}(5 - 3a_i)a_i F_i\right). \tag{4.14}$$

On the other hand, the radial induction factor is computed from:

$$a'_i = \frac{c_1}{1 - c_1}, \tag{4.15}$$

where c_1 can be defined as:

$$c_1 = \left(\frac{\sigma'_i}{4F_i}\right) \frac{C_{Q,i}}{\sin\phi_i \cos\phi_i}, \tag{4.16}$$

and $C_{Q,i}$ is the local forward force coefficient. To obtain the induction factors a_i and a'_i, a fixed point method is used to solve the non-linear system of equations (4.8)-(4.16). The forces in each section are then calculated and integrated in the rotor thrust and torque analysis. Indeed, the thrust T_i and forward Q_i forces for each section can be calculated from:

$$T_i = q_i C_{T,i} = q_i(C_{L,i}\cos\phi_i + C_{D,i}\sin\phi_i), \tag{4.17}$$

$$Q_i = q_i C_{Q,i} = q_i(C_{L,i}\sin\phi_i - C_{D,i}\cos\phi_i), \tag{4.18}$$

where $q_i = 1/2\rho U^2_{rel,i} c_i l_i$ is the dynamic pressure in each section, and $C_{L,i}(\alpha_i)$ and $C_{D,i}(\alpha_i)$ are, respectively, the lift and drag coefficients that are brought to this calculation directly from the aerodynamic characterization of the leaf profile in Section 4.3.1.

Since the thrust T_i at each section is dependent on the ψ angle, only the parallel component to the free-stream velocity contributes to the total thrust T:

$$T = \sum_{i=1}^{N} T_i \cos\psi_i. \tag{4.19}$$

In the same way, the total torque Q is calculated by two terms:

$$Q = \sum_{i=1}^{N} Q_i r_i + T_i \frac{c_i}{4} \sin\psi_i, \tag{4.20}$$

where the first term on the Right Hand Side (RHS) of the previous equation is the summation of the torque at each i-th section (Q_i force cross the radius r_i), and the second term is the torque due to the thrust generated by the curvature of the leaf. This torque due to the thrust of the curvature is perpendicular component of T_i cross the distance from airfoil pressure center to the middle of the airfoil ($c/4$). However, a sensibility analysis based on the numerical results demonstrate that this last term can be neglected from the formulation.

The application of the BEMT analysis in the *Triplaris Americana* seed flight considers the aerodynamic results of the seed foil with rounded edges from the previous section at a $Re = 600$ and the geometric and kinematic data from Table 4.1. The chord distribution is also taken from the three-dimensional model. However, there is not enough information to completely define a reliable leaf curve regarding the pitch angle distribution.

To study the rotation of the seed, three cases are simulated using the BEMT approach. The first type of case fixes the pitch angle $\theta_i = 3$ at all sections. The second case aims to obtain a maximum L/D ratio at each section of the leaf by finding the α_i angle of attack which maximizes this effect. The third case considers a straight blade shape as the baseline blade performance, again using a pitch angle distribution that guarantees the maximum L/D ratio in each section and keeping the same chord distribution through the leaf to make the solidity σ' comparable for all geometries. It is worth mentioning that the first type of configuration could agree to the experimental one since very pronounced pitch angles are not noticed from visual inspection. The third case is intended to evaluate the performance of the curved geometry concerning conventional blade shapes. All inputs were taken from Table 4.1.

The numerical results of the total thrust, total torque, thrust coefficient, and power coefficient are listed in Table 4.2 for all geometries. Also, the results of all cases are presented in Figure 4.9. The pitch angle θ across the leaf span r/R for all three configurations is presented in Figure 4.9(a). This plot shows the increasing difference between the planar and the curved geometries as they approach the root of the blade due to the gaps in the ψ angles between the cases. Also, the local thrust coefficient $C_{T,i}$ is plotted through the leaf span r/R in Figure 4.9(b), where it is observed that the curved leaf with constant pitch obtains the greatest performance. This is confirmed with the results in Table 4.2, where this geometry demonstrates the maximum thrust. However, it is also noted that the power coefficient C_P for the curved geometry with optimal θ is greater than that of the curved geometry with constant θ. This is understood from the great angles of attack present in the curved geometry with constant θ, which produce greater lift and drag forces than the other geometries. This can be noted in Figure 4.9(c)-(d), where the Lift and Drag forces are plot against the leaf span r/R, respectively. Finally, it can be observed in all cases that the drag is proportionally greater than the lift, therefore generating lower L/D values. This leads to a greater thrust driven mainly by the larger drag force.

Table 4.2: Thrust, torque and coefficients for the different configurations.

Geometry	**Thrust** $(N \times 10^{-4})$	**Torque** $(N \cdot m \times 10^{-7})$	C_T	C_P
Curved blade with $\theta = 3$	5.5555	0.6782	0.0202	0.0089
Curved blade with optimal θ	4.9214	1.1986	0.0179	0.0158
Planar blade with optimal θ	4.9987	3.0879	0.0182	0.0408

From Table 4.1 it is expected that at a falling speed $U_\infty = 1.13$m/s the thrust becomes equal to the gravity force F_g, and therefore they balance giving zero acceleration $T + F_g = 0$.

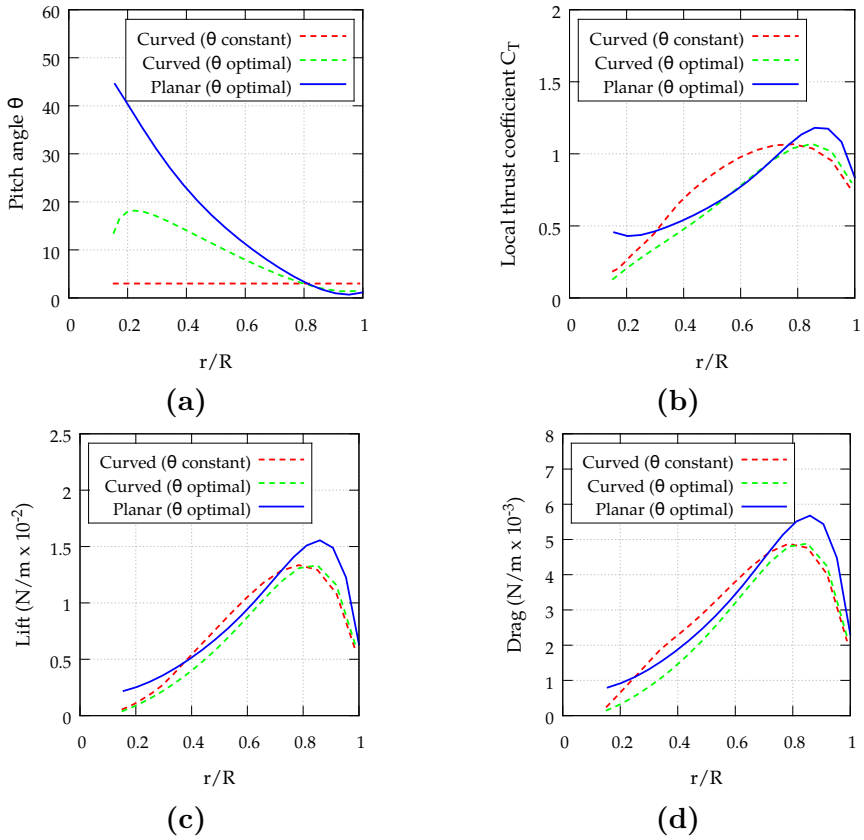


Figure 4.9: Pitch angle θ (a), local thrust coefficient $C_{T_{Local}}$ (b), lift coefficient C_L, and drag coefficient C_D against r/R for the three geometries evaluated.

This falling velocity is similar to that reported in [154]. Therefore, taking the mass of the seed as $m = 0.08$ g (from Table 4.1) and the gravity equal to $g = 9.81$ m/s, the weight results in $F_g = 7.8 \times 10^{-4}$ N. It is observed that this value is of the same magnitude as the thrust generated by any geometry in Table 4.2. however, it is almost twice as much as any of them. Therefore, zero acceleration is not reached for this falling velocity. If the curved blade with $\theta = 3$ is evaluated through a manual iterative process keeping the tip speed ratio λ constant, it is found that the seed reaches the equilibrium of forces at a velocity of approximately 1.34 m/s. However, this value is somewhat above the expected velocity. Note that the BEMT thrust is function of the C_{L_i} and C_{D_i} and therefore, are below the values needed to reach the equilibrium of forces at $U_\infty = 1.13$m/s. This can be due to the fact that a two-dimensional analysis of the cross leaf profile performance underestimates these coefficients, especially the C_{L_i}. The fully three-dimensional characterization of the seed leaf would be necessary to obtain the auto-rotating flight phenomena that is involved in the *Triplaris Americana* fall from the mother tree.

Surfaces that can generate very high lift coefficients constitute a particular phenomenon restricted to low *Reynolds* numbers. There are reports of this behavior in auto-rotating seeds [80, 155], insects [156], birds [157], and others. In all those cases there is a magnifying effect of the lifting force called Leading Edge Vortex (LEV), which is a stable vortex generated on the upper surface of the leading edge at high angles of attack. This effect has been understood as a vortex that delays the flow detachment, but it has been also

described as a zone of lower pressure that increases the lifting force [158]. The main characteristic of the LEV is that the flow should separate after passing the LEV when there are high angles of attack, but, it has been demonstrated that it reattaches to the profile. Note, however, that the LEV is a phenomenon that can not be described by the simplified BEMT and two-dimensional profile characterization analysis. Hence, one can consider the radial flow in the BEMT analysis as a future improvement of the seed flight description using this approach. It has been described in [159] that the radial flow can be related to the delay of the flow separation, and the present findings support that affirmation. Finally, a three-dimensional numerical simulation is required to make an accurate prediction of the forces and fluid flow phenomena involved in the auto-rotating flight of the seed.

4.3.3 Two-dimensional FSI analysis of the flight of the *Triplaris* seed

Finally, the most interesting case is the complete interaction between the deformable seed and the wind flow around its fall. In this last example, the FSI analysis of the seed is achieved using the numerical methodology in Section 4.2. Three different problems are solved to analyze the flight of the seed. The first is a two-dimensional simulation of the cross-section of the seed when observed from its top. The second case is a two-dimensional simulation of the cross-sectional area of the root of the seed. In these first two simulations, a constant rotation velocity is prescribed over the seed. The third case is a fully-coupled FSI simulation of the cross-section of the seed when observed from its top. This last two-dimensional simulation allows both the seed displacement and deformation as the result of its interaction with the fluid flow. In all these cases, the fluid is considered to be air. Hence, the fluid density is set to $\rho^f = 1.0$ kg/m^3 and the dynamic viscosity to $\mu^f = 1.789 \times 10^{-5}$ Kg/(m·s).

4.3.3.1 Top view and root cross-sectional view of the seed

The two fixed rotation cases are evaluated at the time instant of the flight when the seed has reached the constant falling speed, and therefore a constant angular velocity can be prescribed to the seed. These two-dimensional analyses are focused on the angular effect of the flow in the seed rotation. Therefore, any axial phenomenon on the seed has to be necessarily neglected from this analysis. The problem settings for both the top view and root cross-sectional cases are shown in Figure 4.10. Circular fluid domains are implemented for both simulations, considering an external diameter of $D_T = 0.1$ m for the top view case and $D_R = 0.012$ m for the root cross-sectional case. Unstructured meshes composed of 42941 triangular elements for the top view case and 82964 for the root case are used. As boundary conditions, a null velocity $\boldsymbol{u} = 0$ and pressure $p = 0$ conditions are prescribed at the outer boundary of the fluid domain, while a constant rotational velocity $\omega = 1322$ RPM is imposed over the immersed object together with the

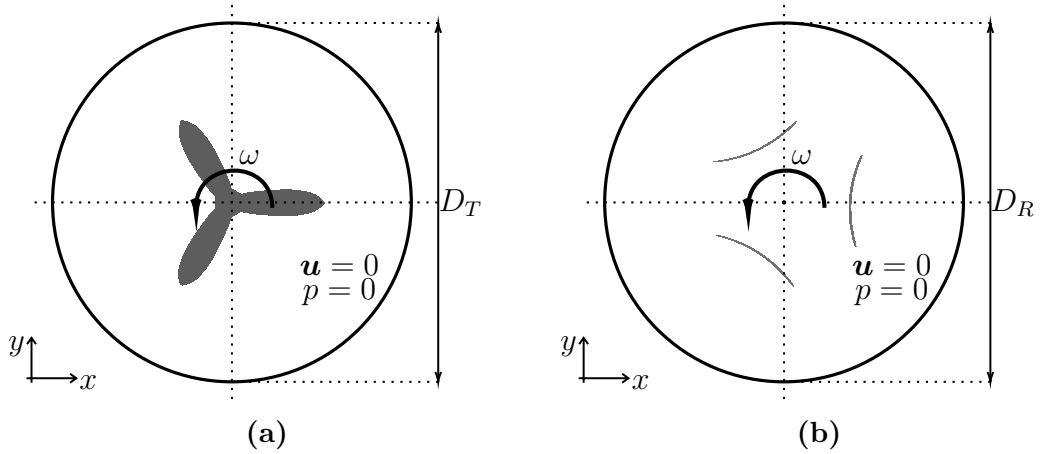
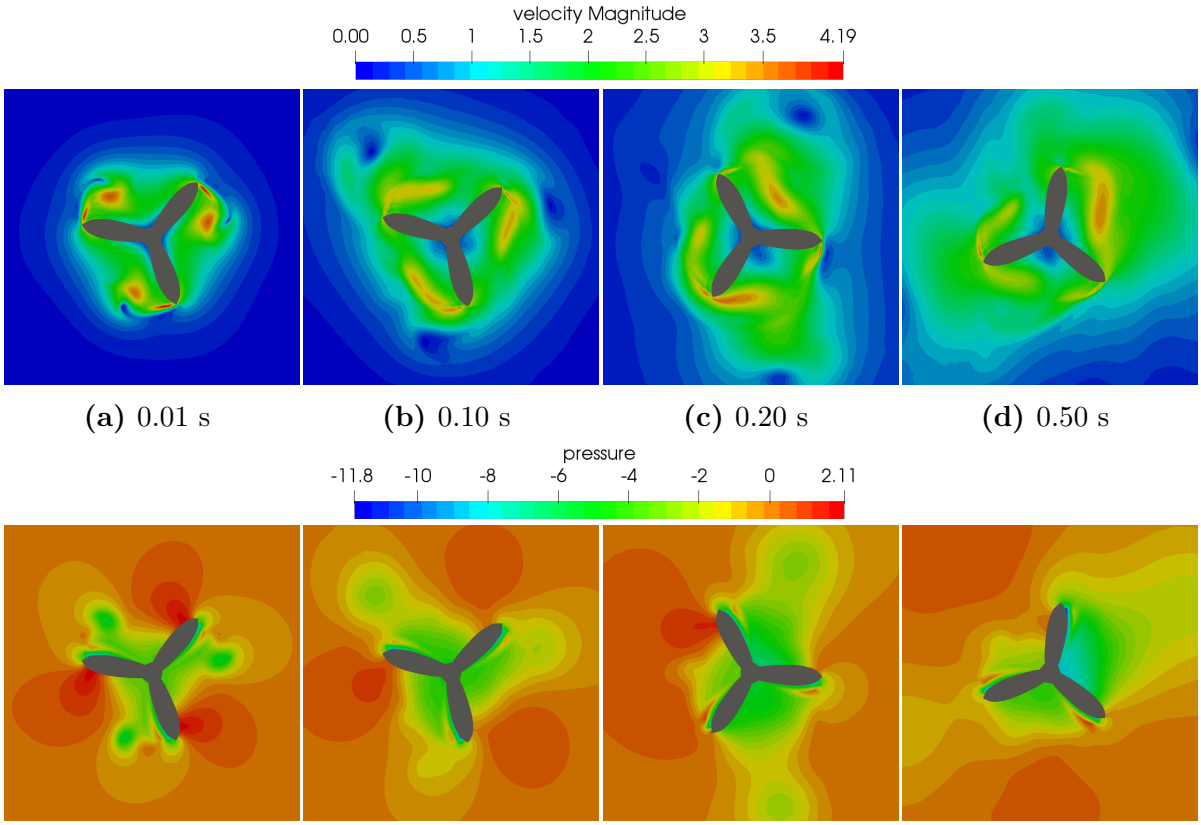


Figure 4.10: Fixed rotation schemes for: (a) the top view and (b) the root cross-section of the *Triplaris Americana* seed cases.

corresponding non-slip condition over the fluid velocity at the seed surface. Both case are simulated during a total simulation time of $T = 1.0$ s using a time step of $\delta t = 0.001$ s. The Nitsche parameters are set to $s_1 = 1000$ and $s_2 = 500$, while the cut stabilization parameters are $s_G = 1.0$, $s_u = 1.0$, and $s_p = 0.1$. Figures 4.11 and 4.12 demonstrate the top view and the root cross-sectional numerical results, respectively. These figures show the velocity and pressure contours of the flow at different instants of the rotation. In the case of the top view case, the flow past the leaves shows a fully transient effect due to the continuous interaction between the moving leaves and the precedent detached wakes. On the other hand, a pseudo-stationary state of the flow around the interface can be noticed for the root case. As pointed out in the BEMT analysis, the root area barely contributes to the self-rotation of the seed. However, the present fluid flow results show that this part may provide angular momentum to the flow that can be harnessed by further radial sections, allowing the seed to rotate more freely. Hypothetically, it may also be the beginning of the transition zone that produces disturbances allowing the rotation. A fully three-dimensional simulation is suggested to investigate this flow pattern.

4.3.3.2 Fully-coupled FSI simulation of the top view of the seed

The last simulation is the fully-coupled 2-way FSI analysis of the top view of the seed. In this case, the seed is represented as a hyperelastic solid, which is free to move and deform as the result of its interaction with the surrounding fluid flow. The problem is modeled as a two-dimensional rectangular domain $\Omega = [0, L] \times [0, H] = [0, 1.0]$m $\times [0, 0.5]$ m, with the seed located at $[c_x, c_y] = [0.2, 0.25]$ m at $t = 0$ s (see Figure 4.13). No-slip boundary conditions are set for the upper and lower walls, as well as for the seed surface. In the case of the inlet face (left-most wall), there is a parabolic horizontal velocity profile with a mean velocity of $u_1 = 1.0$ m/s. Three cases with different initial angular positions are evaluated (0^o, 45^o, and 90^o), which allow us to analyze the rotational effect of the seed in the fluid flow. All cases are simulated using a single unstructured mesh composed of 23338 triangular elements. The simulations are solved for a total time of $T = 3.0$ s using



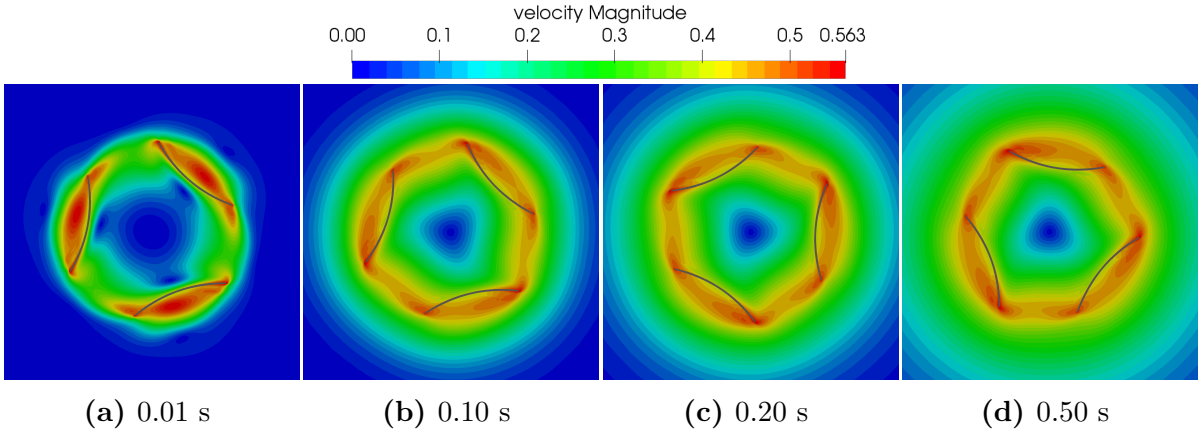
(a) 0.01 s **(b)** 0.10 s **(c)** 0.20 s **(d)** 0.50 s

(e) 0.01 s **(f)** 0.10 s **(g)** 0.20 s **(h)** 0.50 s

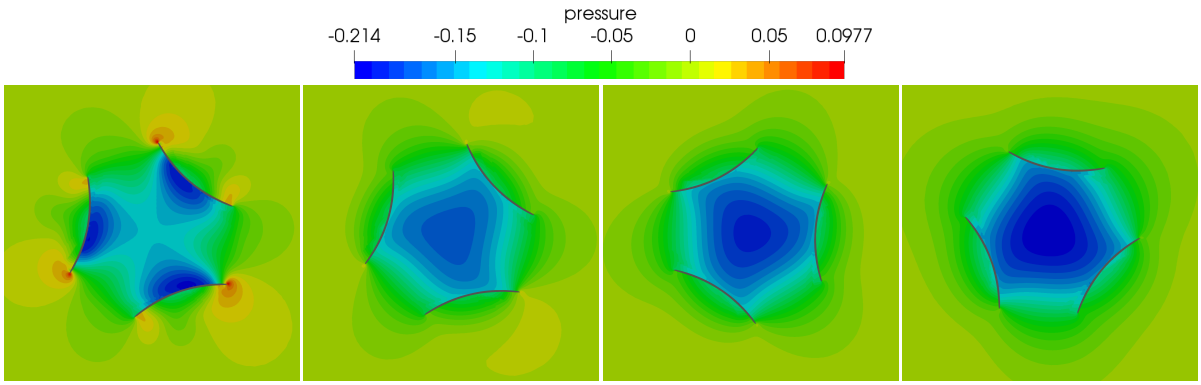
Figure 4.11: Flow results of the seed top view fixed rotation: (Top) velocity magnitude contours, and (Bottom) pressure contours at different time instants.

a time step of $\delta t = 0.01$ s. The Nitsche parameters are set to $s_1 = 1000$ and $s_2 = 500$, while the cut stabilization parameters are $s_G = 1.0$, $s_u = 1.0$, and $s_p = 0.1$.

Figure 4.14 and 4.15 show the contours for the velocity magnitude and pressure results, respectively, for the three different seed starting positions at different times. The seed displacement due to the fluid flow drag can be observed in those figures, where a rectangular grid is printed on the contours to allow the visualization of the movement of the seed in the different instants of time. Transient response in the flow past the seed is obtained in all cases: great vortex detachments are present for the (45^o) and (90^o) initial positions. Hence, the beginning of rotation is related to the generation of downstream vortices. Especially, to the skew-symmetric condition of the vortex production in the (45^o) and (90^o) initial positions. On the contrary, only the seed displacement (with the smallest rotation) is obtained for the (0^o) initial position. Also, a slight deformation of the seed leaves can be appreciated as the seed is dragged downstream. But a spurious effect is observed when the seed rotates: a volume loss in the seed happens gradually through the rotation phenomena. That has been especially the case for the (45^o) position, for which relevant rotation of the seed is obtained. This volume loss is not physical, but it is related to the discrete information transfer between consecutive deformation solutions of the overlapping solid inside the fluid flow. This issue has been encountered for the discrete numerical approach, and it is expected to be corrected soon. The effect of rotation



(a) 0.01 s **(b)** 0.10 s **(c)** 0.20 s **(d)** 0.50 s



(e) 0.01 s **(f)** 0.10 s **(g)** 0.20 s **(h)** 0.50 s

Figure 4.12: Flow results of the root cross-section fixed rotation: (Top) velocity magnitude contours, and (Bottom) pressure contours at different time instants.

on the seed volume can be corroborated in Figures 4.16(a)-(b), where the displacement and rotation of the three configurations are plotted, showing that the more rotation the greater the seed volume loss. Furthermore, it can be seen that the greater the rotation the smaller the displacement. That is explained by the solid rotation theory, where the available kinetic energy in the fluid flow is mostly transformed into rotational energy in the seed. Displacements through time in x and y directions are shown in Figures 4.16(c)-(d). Besides, Figures 4.16(e)-(f) show the forces on the seed in x and y directions, where it is noted the 0^o configuration barely rotates because its initial position generates symmetrical forces in the y direction in the first two seconds. The other two configurations show more unbalanced forces in this direction caused by the vortex generation past the cross-flow blades.

4.4 Conclusions

In this chapter, the computational FSI framework that has been presented through this thesis has been applied in the description of the auto-rotation of the *Triplaris Americana* seed. The methodology is composed of a stabilized FEM approximation for both the incompressible Navier-Stokes equations and the non-linear hyperelastic solid equations. It solves implicitly the fluid-solid coupling condition at the interface through a Picard's scheme

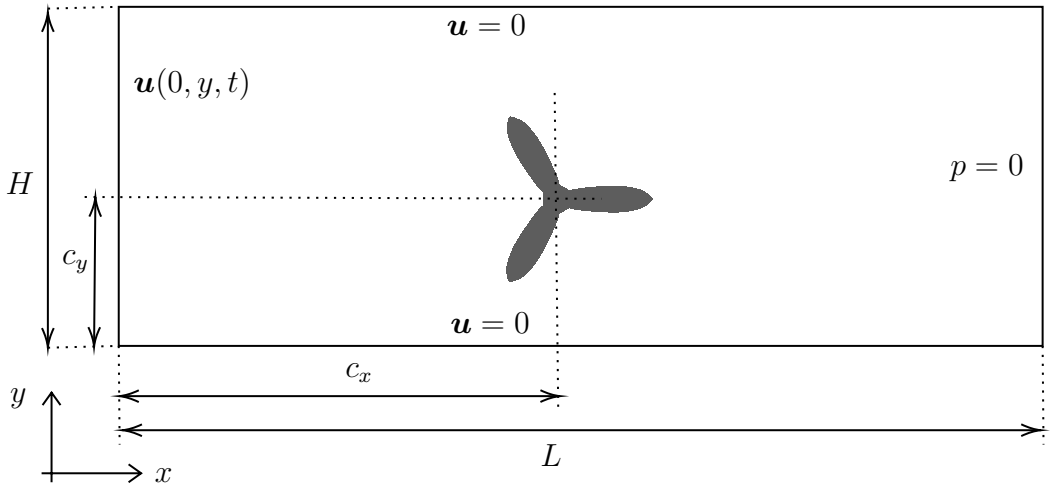
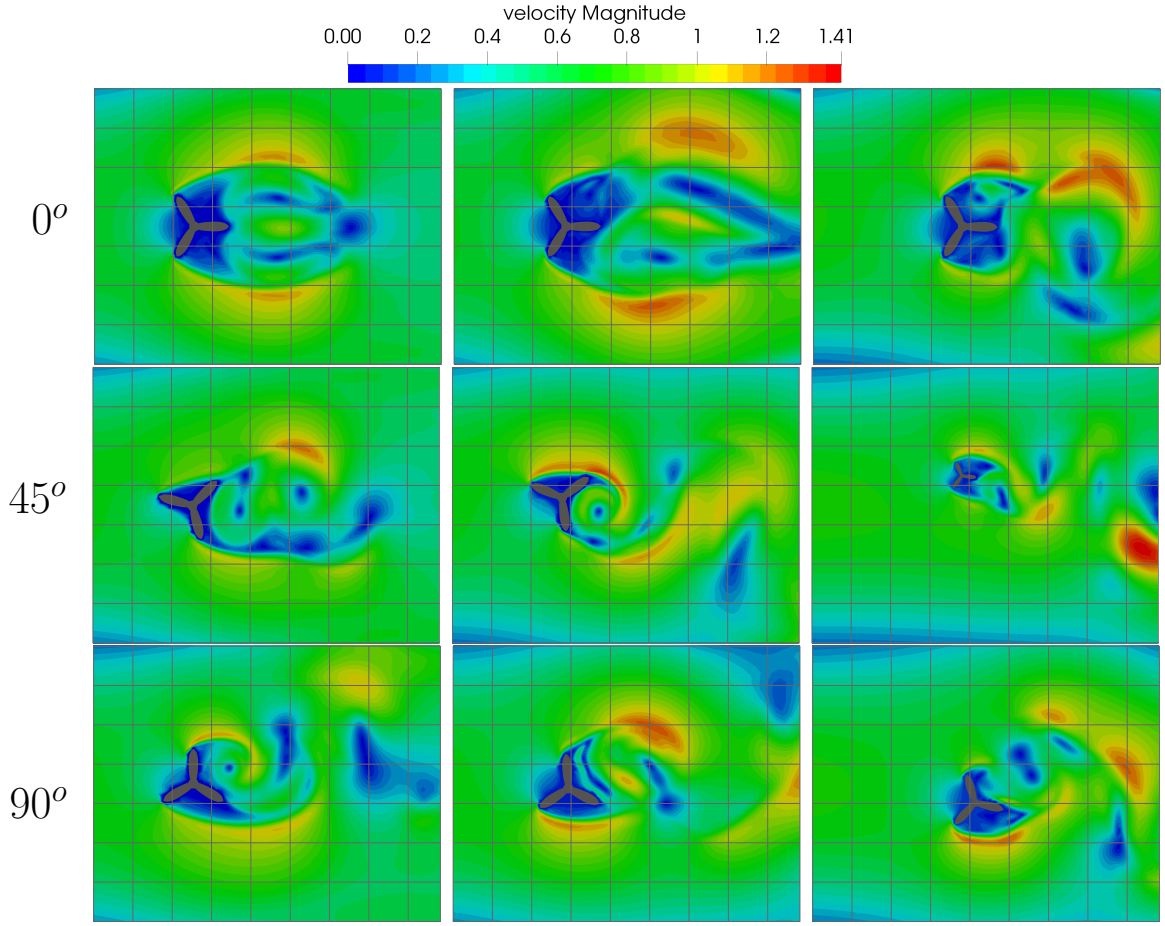


Figure 4.13: Scheme for the fully-coupled FSI simulation of the top view of the *Triplaris Americana* seed.

over multi-mesh methods. The work has exploited the two-dimensional capabilities of the numerical strategy since the full three-dimensional framework is still under algorithmic development. In that sense, several conclusions of the fluid flow interaction with the seed during its flight have been achieved using purely two-dimensional cases.

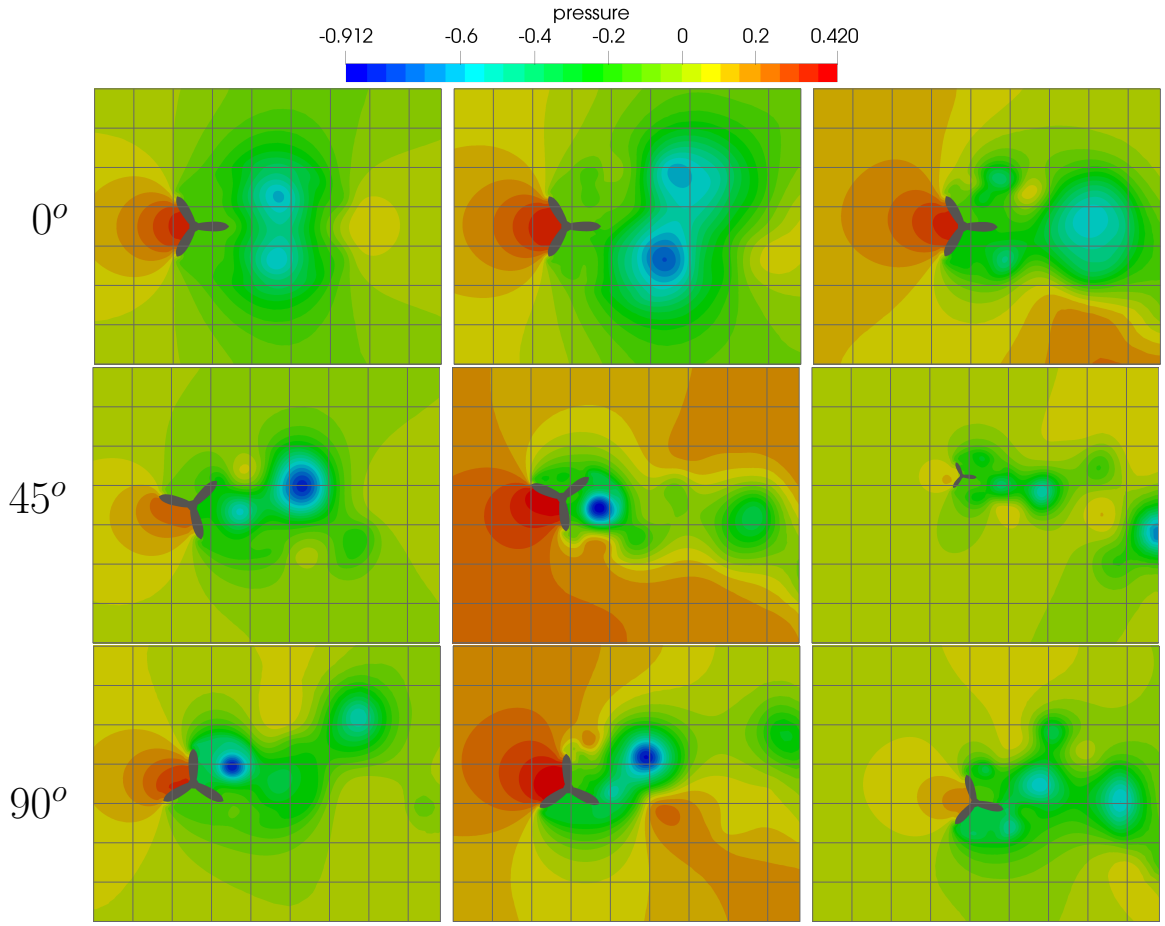
At first, a two-dimensional characterization of the cross profile of the *Triplaris Americana* leaf has demonstrated improved aerodynamic performance in terms of the lift and drag coefficients for the low *Reynolds* numbers involved in the seed flight. The good performance of the cross-sectional airfoil of the *Triplaris* leaves allows it to be used for industrial applications such as Micro Aerial Vehicles (MAV). Additionally, the results of an improved BEMT analysis for non-planar actuator disks concludes that the aerodynamics of the seed are consistent with experimental tests. However, three-dimensional transient phenomena cannot be capture by this analytical model or by two-dimensional profile characterization analysis.

Furthermore, when applying the full FSI methodology to the two-dimensional interaction of the seed cross-sections, results demonstrate that the methodology is suitable for describing the active rotation of the seed and the flow around it. Large displacements and deformations are described as a result of the overlapping meshes strategy. Specifically, the auto-rotation of the *Triplaris Americana* seed has been tested using two different cross-sections of the seed, where it has been demonstrated the complex behavior of the flow around its shape during flight. Additionally, a fully-coupled FSI simulation is used to represent the displacement and rotation of the seed in the fluid flow. The methodology has shown the capacity to represent these phenomena, but it becomes mandatory to achieve three-dimensional analysis to make a complete description of the seed flight mechanisms. Therefore, given the current limitations of the code to work with overlapping meshes in three dimensions, future work is proposed to perform a full coupling FSI simulation in three dimensions of the seed with conforming mesh methods and dynamic meshes using non-inertial frames.



(a) $t = 1$ s **(b)** $t = 2$ s **(c)** $t = 3$ s

Figure 4.14: Contours of velocity magnitude results at different time instants ($1s$, $2s$, and $3s$) for the different initial positions (0^o, 45^o, and 90^o) of the seed.



(a) $t = 1$ s **(b)** $t = 2$ s **(c)** $t = 3$ s

Figure 4.15: Contours of pressure results at different time instants ($1s$, $2s$, and $3s$) for the different initial positions (0^o, 45^o, and 90^o) of the seed.

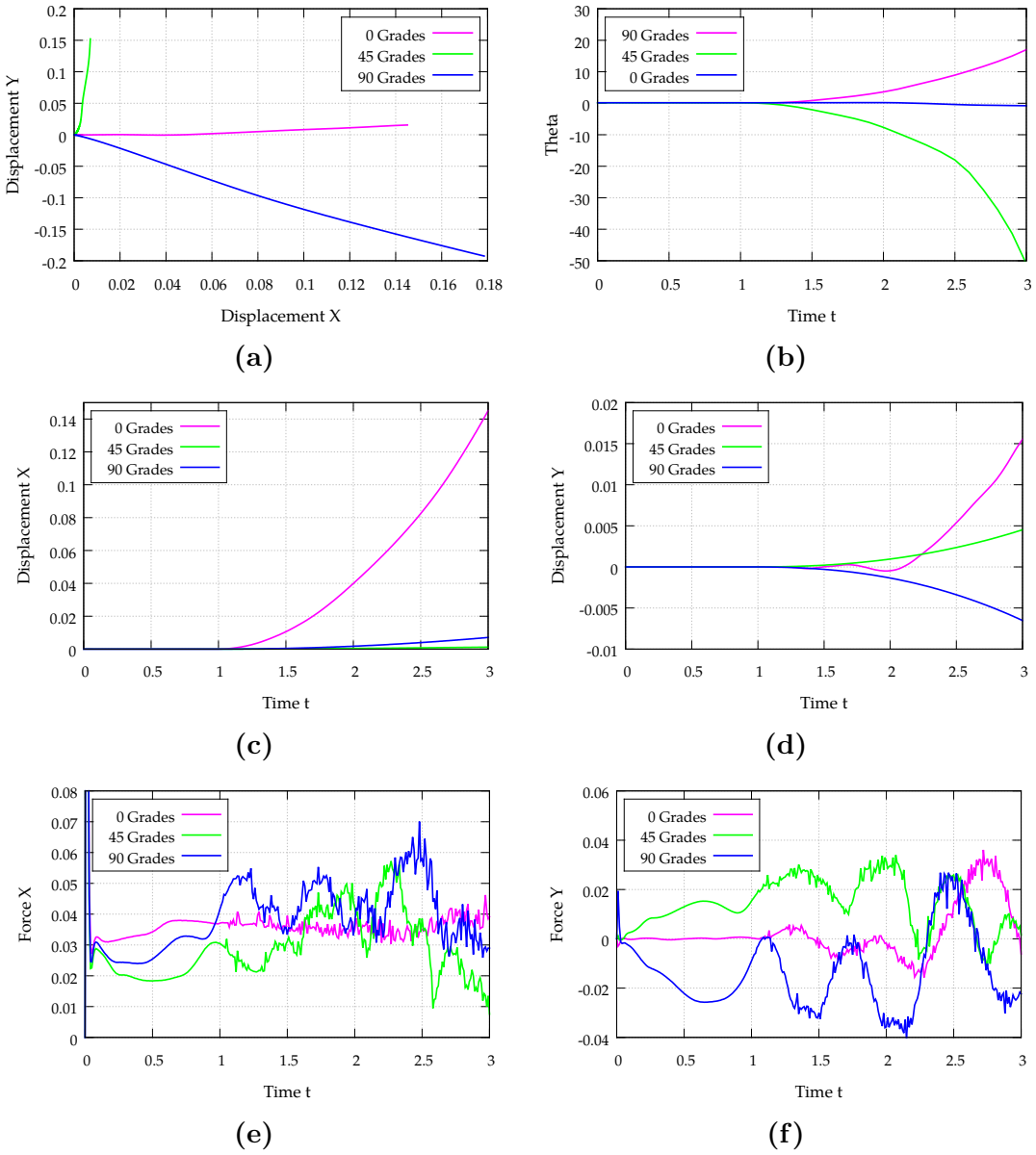


Figure 4.16: Integral results of the fully-coupled FSI simulation of the top view of the seed: (a) $x - y$ displacements phase diagram, (b) rotation angle through time t, (c) x and (d) y displacements, and, (e) x and (f) y forces on the seed for the different initial configurations..

5 Conclusion

5.1 Achievements

The main objective of this thesis has been to develop an approach to solving Fluid-Structure Interaction problems. Specifically, it has been used in the solution of aerodynamic cases involving Newtonian and incompressible fluid flows interacting with largely displaced and/or deformed solids.

In the first part of this thesis, the instability problems associated with the *Galerkin* formulation of the fluid flow equations have been addressed. The numerical approximation of the incompressible Navier-Stokes has been achieved by adopting the Variational Multi-Scale (VMS) framework and defining the sub-scales as both Algebraic (ASGS) and Orthogonal (OSGS). This development has been presented in Chapter 2, together with some other necessary numerical ingredients. The first has been the weak prescription of essential conditions in the immersed boundary of the fluid flow employing the Nitsche's method. Also, the tracking of the immersed interface has been ruled by the Level-Set method for simply-shaped objects, while a mesh that overlaps the fluid mesh has been used for complex geometries. In this regard, a sub-triangle technique has been implemented to perform the numerical integration of the cut elements near the moving interface. But the instabilities generated by the bad cuts need to be locally stabilized. Hence, the local control of the fluid flow solution near the interface has been granted by including the Ghost Penalty method with Orthogonal Projections (GPOP) into the fluid discrete formulation. The inclusion of alternative definitions for the algorithmic parameters of the Nitsche and GPOP methods has been a fundamental contribution of this first part of the thesis: the use of $\alpha\beta^{-1}$ and $\gamma_1\beta^{-1}$ parameters work as a cut cell sensor that demonstrated improved performance of the numerical accuracy provided by the method. This novel definition of the algorithmic parameters has demonstrated optimal behavior for manufactured solution problems when the weak imposition of the Dirichlet boundary conditions has been tested. Also, the ability to simulate real engineering problems of the present formulation has been granted: from the widely tested transient flow past a cylinder to the complex cases of continuously moving obstacles in a channel flow.

Chapter 3 has been devoted to the solution of the fully-coupled 2-way Fluid-Structure Interaction (FSI) problem by applying the stabilized numerical methodology described before. The main objective has been to describe the interaction between an immersed hyperelastic solid object and the encompassing fluid flow. Hence, the discrete coupled formulation of the incompressible flow and the hyperelastic solid has been developed in that chapter. The numerical ingredients of the FSI strategy have included the computational methodology to deal with the multiple finite element meshes representing the fluid and solid domains, the projection of the fluid tractions onto the solid boundary, and the algorithmic approach for the solid displacements in the fluid flow. Several applications are simulated, some of which make use of a simplified rigid body motion scheme for the simulation of embedded bodies. In this sense, the third chapter has been mainly dedicated

to the simulation of applied engineering cases, including the 2-way FSI Turek and the flow past moving cylinders. It has been also devoted to the analysis of the aerodynamic performance of flapping foils.

Chapter 4 has presented the aerodynamic simulation and characterization of the auto-rotating *Triplaris Americana* seed applying the complete methodology that was presented in the previous chapters. The work has exploited the two-dimensional capabilities of the numerical strategy since the full three-dimensional framework is still under algorithmic development. In that sense, several conclusions of the fluid flow interaction with the seed during its flight have been achieved using purely two-dimensional cases. First, the two-dimensional characterization of the cross profile of the *Triplaris Americana* and the improved BEMT analysis for non-planar actuator disks representing the leaves have been used to evaluate its aerodynamic performance. In this sense, the transient analysis of the aerodynamic performance of the cross-section of the *Triplaris* leaf has been carried out and compared with other low *Reynolds* airfoils to demonstrate the improved efficiency. Indeed, numerical simulations demonstrated the superior performance of this geometry, which allows it to be used in industrial applications such as Micro Aerial Vehicles (MAV). Two-dimensional simulations have also been performed to visualize the flow around the seed when it rotates at a constant speed. Two cases have been evaluated: a top view of the seed and a cross-sectional view of the root of the seed. Finally, the fully-coupled 2-way FSI methodology is applied to the interesting case of the two-dimensional displacement, deformation, and rotation of the top view of the seed. This case emulates the drag of the seed by the airflow, showing the capacity of the proposed methodology to solve the main motivation of the present thesis.

5.2 Future works

The development of the thesis has brought several ideas that can be addressed in the future.

- The most important issue to be tackled is the three-dimensional extension of the algorithmic approach that has been presented in the thesis. Especially, regarding the colliding schemes between multiple meshes composed by tetrahedra. That will become the major tool development to analyze the complex aerodynamics of immersed elastic bodies producing lift.

- One foregoing work in the mathematical formulation of the method is a deeper study of the added algorithmic parameters of the Nitsche and Ghost Penalty with Orthogonal Projection (GPOP), $\alpha\beta^{-1}$ and $\gamma_1\beta^{-1}$. These parameters can still be improved.

  This is explained by the asymptotic behavior of the error convergence against the mesh size. A detailed mathematical analysis is needed to understand the boundedness and symmetry of these new terms in the well-known global stabilized formulation arising from the VMS framework.

- Another task is the evaluation of alternative methods to the Nitsche's Method for the weak imposition that may not need user-defined parameters, especially the Linked Lagrange method or the Nitsche non-symmetric method without penalty parameters. That will simplify the proposed methodology and will allow the adjustment of GPOP parameters only making the developed code to be a front-end software for computational engineers.

- To improve accuracy at the interface, another task would be the inclusion of adaptive mesh methods. Some automatic and dynamic mesh refinement strategies or the local inclusion of higher interpolation order near the interface can greatly improve the computational methodology. Additionally, linear elements have been a valid approximation to build the present method. However, the precision at the interface must be upgraded to simulate real applied cases.

- In the same line, the variational subscales can be used as error estimators for the adaptive mesh refinement. Including the subscales at the edges of the element in the variational formulation may be the way to accurately estimate the finite element error at each fluid element.

- The volume loss encountered in the rotation of the solid bodies must be corrected. Since this loss is related to the discrete information transfer between consecutive deformation solutions of the overlapping solid inside the fluid flow, the work must be focused on debugging the discrete numerical approach of the solid temporal integration.

- For a better analysis of the results, more post-processing tools need to be implemented in the algorithm to address the calculation of variables such as the torque, which is necessary for the aeroelastic applications. For instance, in the calculation of the wind turbine power coefficient or the flapping foil pitching moment.

- The computational study of several flapping foil systems arise as possible direct applications and technology developments from the present thesis. For instance, the energy harvesting systems with flexible trailing edge flapping airfoils need to be detailed. Another problem set can be the three-dimensional analysis of the flapping foil system, for which less simplifying conditions may lead to better descriptions of this technology. The flapping foil systems in propulsion applications are technologies that can be studied straightforwardly with the present developments. Lastly, the performance analysis of the flapping foils when these are allowed to move freely in the channel flow must also be quickly addressed.

- Indeed, scaling the proposed methodology to three-dimensions is mandatory for evaluating more complex seed rotation problems. As commented before, the Triplaris Samara flight can not be represented by simplified analytical models or the two-dimensional analysis and must be deeper investigated by using the three-dimensional setting. In this sense it is also proposed to perform a full three-dimensional FSI simulation of the seed with conforming mesh methods and dynamic meshes using non-inertial frames, which are technologies that are available and can be relatively

easy to implement.

- Once the Triplaris Samara flight is represented, the same developments can be used to different rotational geometries that increase the description complexity of the solid material deformation: the description of complex structures such as composite materials (of which wind turbine blades are usually made) becomes mandatory in the technological transfer of this thesis to the aeronautic industry.

References

[1] Gene Hou, Jin Wang, and Anita Layton. Numerical Methods for Fluid-Structure Interaction — A Review. *Communications in Computational Physics*, 12(2):337–377, aug 2012.

[2] Xiaocui Wu, Yiwei Wang, Chenguang Huang, Yubiao Liu, and Lingling Lu. Experiment and numerical simulation on the characteristics of fluid-structure interactions of non-rigid airships. *Theoretical and Applied Mechanics Letters*, 5(6):258–261, 2015.

[3] Fotis Sotiropoulos and Xiaolei Yang. Immersed boundary methods for simulating fluid–structure interaction. *Progress in Aerospace Sciences*, 65(4):1–21, feb 2014.

[4] Thomas J.R. Hughes. Multiscale phenomena: Green's functions, the Dirichlet-to-Neumann formulation, subgrid scale models, bubbles and the origins of stabilized methods. *Computer Methods in Applied Mechanics and Engineering*, 127(1-4):387–401, nov 1995.

[5] J.G. Marshall and M. Imregun. a Review of Aeroelasticity Methods With Emphasis on Turbomachinery Applications. *Journal of Fluids and Structures*, 10(3):237–267, 1996.

[6] Ramji Kamakoti and Wei Shyy. Fluid-structure interaction for aeroelastic applications. *Progress in Aerospace Sciences*, 40(8):535–558, 2004.

[7] M. Roura, A. Cuerva, A. Sanz-Andrés, and A. Barrero-Gil. A panel method free-wake code for aeroelastic rotor predictions. *Wind Energy*, 13(4):357–371, may 2010.

[8] Y. L. Young. Fluid-structure interaction analysis of flexible composite marine propellers. *Journal of Fluids and Structures*, 24(6):799–818, 2008.

[9] R. L. Campbell and E. G. Paterson. Fluid-structure interaction analysis of flexible turbomachinery. *Journal of Fluids and Structures*, 27(8):1376–1391, 2011.

[10] Iman Borazjani. Fluid-structure interaction, immersed boundary-finite element method simulations of bio-prosthetic heart valves. *Computer Methods in Applied Mechanics and Engineering*, 257:103–116, 2013.

[11] T.M. M van Opstal, E.H. H van Brummelen, and G.J. J van Zwieten. A finite-element/boundary-element method for three-dimensional, large-displacement fluid-structure-interaction. *Computer Methods in Applied Mechanics and Engineering*, 284:637–663, feb 2015.

[12] T M van Opstal and E H van Brummelen. A finite-element/boundary-element method for large-displacement fluid-structure interaction with potential flow. *Computer Methods in Applied Mechanics and Engineering*, 266:57–69, nov 2013.

[13] J. Pereira Gomes, S. Yigit, H. Lienhart, and M. Schäfer. Experimental and numerical study on a laminar fluid-structure interaction reference test case. *Journal of Fluids and Structures*, 27(1):43–61, jan 2011.

[14] Kenji Takizawa. Computational engineering analysis with the new-generation space-time methods. *Computational Mechanics*, 54(2):193–211, 2014.

[15] Kenji Takizawa, Tayfun E. Tezduyar, and Ryan Kolesar. FSI modeling of the Orion spacecraft drogue parachutes. *Computational Mechanics*, 55(6):1167–1179, jun 2015.

[16] Xiaohui Su, Yuanwei Cao, and Yong Zhao. An unstructured mesh arbitrary Lagrangian-Eulerian unsteady incompressible flow solver and its application to insect flight aerodynamics. *Physics of Fluids*, 28(6):061901, jun 2016.

[17] Zati Hakim Azizul Hasan and N. Selvanathan. 2D Arbitrary Lagrangian-Eulerian (ALE) Model of Blood Flow in the Left Ventricle (LV) of the Heart. In *Computer Methods and Programs in Biomedicine*, pages 554–557. 2007.

[18] Jean Donea, Antonio Huerta, J.-Ph. Ponthot, A. Rodrï¿½ï¿½guez-Ferran, and A. Rodriguez-Ferran. Arbitrary Lagrangian-Eulerian Methods. In *Encyclopedia of Computational Mechanics*. John Wiley & Sons, Ltd, Chichester, UK, nov 2004.

[19] R. van Loon, P.D. Anderson, F.N. van de Vosse, and S.J. Sherwin. Comparison of various fluid–structure interaction methods for deformable bodies. *Computers & Structures*, 85(11-14):833–843, jun 2007.

[20] Raphaël Loubère, Pierre-Henri Maire, Mikhail Shashkov, Jérôme Breil, and Stéphane Galera. ReALE: A reconnection-based arbitrary-Lagrangian-Eulerian method. *Journal of Computational Physics*, 229(12):4724–4761, jun 2010.

[21] Scott T. Miller, R. L. Campbell, C. W. Elsworth, J. S. Pitt, and D. a. Boger. An Overset Grid Method for Fluid-Structure Interaction. *World Journal of Mechanics*, 04(07):217–237, 2014.

[22] G Houzeaux, B Eguzkitza, Romain Aubry, H Owen, and M Vázquez. A Chimera method for the incompressible Navier-Stokes equations. *International Journal for Numerical Methods in Fluids*, 75(3):155–183, may 2014.

[23] Steffen Basting, Annalisa Quaini, Sunčica Čanić, and Roland Glowinski. Extended ALE Method for fluid-structure interaction problems with large structural displacements. *Journal of Computational Physics*, 331:312–336, 2017.

[24] Joan Baiges, Ramon Codina, Arnau Pont, and Ernesto Castillo. An adaptive Fixed-Mesh ALE method for free surface flows. *Computer Methods in Applied Mechanics and Engineering*, 313:159–188, jan 2017.

[25] Tayfun E. Tezduyar. Finite element methods for flow problems with moving boundaries and interfaces. *Archives of Computational Methods in Engineering*, 8(2):83–130, jun 2001.

[26] a. Legay, J. Chessa, and T. Belytschko. An Eulerian-Lagrangian method for fluid-structure interaction based on level sets. *Computer Methods in Applied Mechanics and Engineering*, 195(17-18):2070–2087, 2006.

[27] Thomas J R Hughes, Gonzalo R Feijóo, Luca Mazzei, and Jean-Baptiste Quincy. The variational multiscale method—a paradigm for computational mechanics. *Computer Methods in Applied Mechanics and Engineering*, 166(1):3–24, 1998.

[28] Ramon Codina. Stabilization of incompressibility and convection through orthogonal sub-scales in finite element methods. *Computer Methods in Applied Mechanics and Engineering*, 190(13-14):1579–1599, dec 2000.

[29] Rajat Mittal and Gianluca Iaccarino. Immersed boundary methods. *Annual Review of Fluid Mechanics*, 37(1):239–261, jan 2005.

[30] Iman Borazjani, Liang Ge, Fotis Sotiropoulos, and F. Borazjani, I., Ge, L., Sotiropoulos. Curvilinear immersed boundary method for simulating fluid structure interaction with complex 3D rigid bodies. *Journal of Computational Physics*, 227(16):7587–7620, aug 2008.

[31] Luoding Zhu and Charles 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.

[32] Fang-Bao Tian, Haoxiang Luo, Jialei Song, and Xi-Yun Lu. Force production and asymmetric deformation of a flexible flapping wing in forward flight. *Journal of Fluids and Structures*, 36:149–161, jan 2013.

[33] Charles S. Peskin. Numerical analysis of blood flow in the heart. *Journal of Computational Physics*, 1977.

[34] Enkeleida Lushi and Charles S. Peskin. Modeling and simulation of active suspensions containing large numbers of interacting micro-swimmers. *Computers & Structures*, 122:239–248, jun 2013.

[35] Markus Uhlmann. An immersed boundary method with direct forcing for the simulation of particulate flows. *Journal of Computational Physics*, 209(2):448–476, nov 2005.

[36] Yongsam Kim and Charles S Peskin. A penalty immersed boundary method for a rigid body in fluid. *Physics of Fluids*, 28(3):033603, mar 2016.

[37] Qianlong Liu and Oleg V. Vasilyev. A Brinkman penalization method for compressible flows in complex geometries. *Journal of Computational Physics*, 2007.

[38] C. Bost, G.-H. Cottet, and E. Maitre. Convergence Analysis of a Penalization Method for the Three-Dimensional Motion of a Rigid Body in an Incompressible Viscous Fluid. *SIAM Journal on Numerical Analysis*, 2010.

[39] M. Bergmann and A. Iollo. Modeling and simulation of fish-like swimming. *Journal of Computational Physics*, 2011.

[40] Antoine Morente, Jérôme Laviéville, and Dominique Legendre. A penalization method for the simulation of bubbly flows. *Journal of Computational Physics*, 374:563–590, 2018.

[41] T. Ye, R. Mittal, H.S. Udaykumar, and W. Shyy. An Accurate Cartesian Grid Method for Viscous Incompressible Flows with Complex Immersed Boundaries. *Journal of Computational Physics*, 156(2):209–240, dec 1999.

[42] Nicolas Moës, John Dolbow, and Ted Belytschko. A finite element method for crack growth without remeshing. *International Journal for Numerical Methods in Engineering*, 46(1):131–150, sep 1999.

[43] P. Díez, R. Cottereau, and S. Zlotnik. A stable extended fem formulation for multi-phase problems enforcing the accuracy of the fluxes through lagrange multipliers. *International Journal for Numerical Methods in Engineering*, 96(5):303–322, 2013.

[44] Frédéric Alauzet, Benoit Fabrèges, Miguel A. Fernández, and Mikel Landajuela. Nitsche-XFEM for the coupling of an incompressible fluid with immersed thin-walled structures. *Computer Methods in Applied Mechanics and Engineering*, 301:300–335, apr 2016.

[45] Zhijun Tan, D.V. Le, K.M. Lim, and B.C. Khoo. An Immersed Interface Method for the Incompressible Navier–Stokes Equations with Discontinuous Viscosity Across the Interface. *SIAM Journal on Scientific Computing*, 31(3):1798–1819, jan 2009.

[46] Erik Burman, Susanne Claus, Peter Hansbo, Mats G. Larson, and André Massing. CutFEM: Discretizing geometry and partial differential equations. *International Journal for Numerical Methods in Engineering*, 104(7):472–501, nov 2015.

[47] W.G. Dettmer, C. Kadapa, and D. Perić. A stabilised immersed boundary method on hierarchical b-spline grids. *Computer Methods in Applied Mechanics and Engineering*, 311:415–437, nov 2016.

[48] Susanne Claus and Pierre Kerfriden. A CutFEM method for two-phase flow problems. *Computer Methods in Applied Mechanics and Engineering*, 348:185–206, may 2019.

[49] Erik Burman and Miguel A. Fernández. An unfitted Nitsche method for incompressible fluid–structure interaction using overlapping meshes. *Computer Methods in Applied Mechanics and Engineering*, 279:497–514, sep 2014.

[50] C. Kadapa, W.G. Dettmer, and D. Perić. A stabilised immersed boundary method on hierarchical b-spline grids for fluid–rigid body interaction with solid–solid contact. *Computer Methods in Applied Mechanics and Engineering*, 318:242–269, may 2017.

[51] Ursula M. Mayer, Axel Gerstenberger, and Wolfgang A. Wall. Interface handling for three-dimensional higher-order XFEM-computations in fluid-structure interaction. *International Journal for Numerical Methods in Engineering*, 79(7):846–869, aug 2009.

[52] Dominik Schillinger, Isaac Harari, Ming Chen Hsu, David Kamensky, Stein K.F. Stoter, Yue Yu, and Ying Zhao. The non-symmetric Nitsche method for the parameter-free imposition of weak boundary and coupling conditions in immersed finite elements. *Computer Methods in Applied Mechanics and Engineering*, 309:625–652, 2016.

[53] Long Chen, Huayi Wei, and Min Wen. An interface-fitted mesh generator and virtual element methods for elliptic interface problems. *Journal of Computational Physics*, 334(20134301120003):327–348, apr 2017.

[54] László Kudela, Nils Zander, Tino Bog, Stefan Kollmannsberger, and Ernst Rank. Efficient and accurate numerical quadrature for immersed boundary methods. *Advanced Modeling and Simulation in Engineering Sciences*, 2(1):10, dec 2015.

[55] F. de Prenter, C.V. Verhoosel, G.J. van Zwieten, and E.H. van Brummelen. Condition number analysis and preconditioning of the finite cell method. *Computer Methods in Applied Mechanics and Engineering*, 316:297–327, apr 2017.

[56] Dominik Schillinger and Martin Ruess. The Finite Cell Method: A Review in the Context of Higher-Order Structural Analysis of CAD and Image-Based Geometric

Models. *Archives of Computational Methods in Engineering*, 22(3):391–455, jul 2015.

[57] Vasco Varduhn, Ming-Chen Hsu, Martin Ruess, and Dominik Schillinger. The tetrahedral finite cell method: Higher-order immersogeometric analysis on adaptive non-boundary-fitted meshes. *International Journal for Numerical Methods in Engineering*, 107(12):1054–1079, sep 2016.

[58] Frank P. T. Baaijens. A fictitious domain/mortar element method for fluid-structure interaction. *International Journal for Numerical Methods in Fluids*, 35(7):743–761, apr 2001.

[59] Anita Hansbo and Peter Hansbo. An unfitted finite element method, based on Nitsche's method, for elliptic interface problems. *Computer Methods in Applied Mechanics and Engineering*, 191(47-48):5537–5552, nov 2002.

[60] Janos Benk, Michael Ulbrich, and Miriam Mehl. The Nitsche's method of the Navier-Stokes equation for immersed and moving boundaries. 2011.

[61] Thomas Boiveau and Erik Burman. A penalty-free Nitsche method for the weak imposition of boundary conditions in compressible and incompressible elasticity. *IMA Journal of Numerical Analysis*, 36(2):770–795, apr 2016.

[62] Joan Baiges, Ramon Codina, Florian Henke, Shadan Shahmiri, and Wolfgang A. Wall. A symmetric method for weakly imposing Dirichlet boundary conditions in embedded finite element meshes. *International Journal for Numerical Methods in Engineering*, 90(5):636–658, may 2012.

[63] Christoph Lehrenfeld and Arnold Reusken. Optimal preconditioners for Nitsche-XFEM discretizations of interface problems. *Numerische Mathematik*, 135(2):313–332, feb 2017.

[64] Erik Burman. Ghost penalty. *Comptes Rendus Mathematique*, 348(21-22):1217–1220, nov 2010.

[65] Erik Burman and Peter Hansbo. Fictitious domain methods using cut elements: III. a stabilized nitsche method for stokes' problem. *ESAIM: Mathematical Modelling and Numerical Analysis - Modélisation Mathématique et Analyse Numérique*, 48(3):859–874, 2014.

[66] I. R. Dehkordi, H. Shahverdi, A. S. Nobari, and A. Khalili. Aeroelastic numerical approach of a wing based on the finite element and boundary element methods. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 227(5):882–894, may 2013.

[67] Sang Truong Ha, Long Cu Ngo, Muhammad Saeed, Byoung Jin Jeon, and Hyounggwon Choi. A comparative study between partitioned and monolithic methods for the problems with 3D fluid-structure interaction of blood vessels. *Journal of Mechanical Science and Technology*, 31(1):281–287, 2017.

[68] Santiago Badia, Fabio Nobile, and Christian Vergara. Fluid-structure partitioned procedures based on Robin transmission conditions. *Journal of Computational Physics*, 227(14):7027–7051, 2008.

[69] Joris Degroote and Jan Vierendeels. Multi-solver algorithms for the partitioned simulation of fluid-structure interaction. *Computer Methods in Applied Mechanics and Engineering*, 200(25-28):2195–2210, 2011.

[70] W. McKinney and J. Delaurier. The wingmill - An oscillating-wing windmill. In *Wind Energy Conference*, volume 5, Reston, Virigina, apr 1980. American Institute of Aeronautics and Astronautics.

[71] Keon Lindsey. *A Feasibility Study of Oscillating-Wing Power Generators*. PhD thesis, Naval Postgraduate School, 2002.

[72] Bing Zhu, Yun Huang, and Yongming Zhang. Energy harvesting properties of a flapping wing with an adaptive Gurney flap. *Energy*, 152:119–128, 2018.

[73] Firas Siala and James A. Liburdy. Energy harvesting of a heaving and forward pitching wing with a passively actuated trailing edge. *Journal of Fluids and Structures*, 57:1–14, 2015.

[74] J. Wu, C. Shu, N. Zhao, and F. B. Tian. Numerical study on the power extraction performance of a flapping foil with a flexible tail. *Physics of Fluids*, 27(1), 2015.

[75] Zhengliang Liu, Fang Bao Tian, John Young, and Joseph C.S. Lai. Flapping foil power generator performance enhanced with a spring-connected tail. *Physics of Fluids*, 29(12), 2017.

[76] Yang Cao, Wei Han, Bing Zhu, Xiaojing Sun, Ying Wang, Zhongquan Charlie Zheng, Diangui Huang, and Guoqing Wu. Research on energy extraction characteristics of an adaptive deformation oscillating-wing. *Journal of Renewable and Sustainable Energy*, 7(2):023101, 2015.

[77] E. J. Falagkaris, D. M. Ingram, K. Markakis, and I. M. Viola. PROTEUS: A coupled iterative force-correction immersed-boundary cascaded lattice Boltzmann solver for moving and deformable boundary applications. *Computers and Mathematics with Applications*, 75(4):1330–1354, 2018.

[78] Jie Wu and Chang Shu. A coupled immersed boundary-lattice Boltzmann method and its simulation for biomimetic problems. *Theoretical and Applied Mechanics Letters*, 5(1):16–19, 2015.

[79] Fang Bao Tian, John Young, and Joseph C.S. Lai. Improving power-extraction efficiency of a flapping plate: From passive deformation to active control. *Journal of Fluids and Structures*, 51:384–392, 2014.

[80] Byungkwon Jung Rezgui, Djamel, Byungkwon Jung, and Djamel Rezgui. Investigating the Autorotational Performance of Scaled Samara Rotor in Vertical and Forward Flight. In *42nd European Rotorcraft Forum 2016*, number September 2016, pages 1252–1266. Association Aeronautique et Astronautique de France, 2017.

[81] Adnan M. El Makdah, Laura Sanders, Kai Zhang, and David E. Rival. The stability of leading-edge vortices to perturbations on samara-inspired rotors: a novel solution for gust resistance. *Bioinspiration & Biomimetics*, 15(1):016006, dec 2019.

[82] Mohamed Y. Zakaria, Carlos R. dos Santos, Abdallah Dayhoum, Flávio D. Marques, and Muhammad R. Hajj. Modeling and prediction of aerodynamic characteristics

of free fall rotating wing based on experiments. *IOP Conference Series: Materials Science and Engineering*, 610(1):012098, oct 2019.

[83] E. Salcedo, C. Trevino, R. O. Vargas, and L. Martinez-Suastegui. Stereoscopic particle image velocimetry measurements of the three-dimensional flow field of a descending autorotating mahogany seed (Swietenia macrophylla). *Journal of Experimental Biology*, 216(11):2017–2030, jun 2013.

[84] Kapil Varshney, Song Chang, and Z. Jane Wang. The kinematics of falling maple seeds and the initial transition to a helical motion. *Nonlinearity*, 25(1), 2012.

[85] Kirthi Tennakone. Aerodynamics and right-left symmetry in wind dispersal of maple, dipterocarps, conifers and some genera of apocyanaceae and magnoliaceae. *Journal of the National Science Foundation of Sri Lanka*, 45(3):201, sep 2017.

[86] G Arranz, M Moriche, M Uhlmann, O Flores, and M García-Villalba. Kinematics and dynamics of the auto-rotation of a model winged seed. *Bioinspiration & Biomimetics*, 13(3):036011, apr 2018.

[87] Injae Lee and Haecheon Choi. Flight of a falling maple seed. *Physical Review Fluids*, 2(9):090511, sep 2017.

[88] Bruno A Roccia, Marcos L Verstraete, and Manuel Valdano. Study of the dynamics and unsteady aerodynamics of rotating flying seeds. *Mecánica Computacional (AMCA)*, XXXVII:5–7, 2019.

[89] Myong H Sohn, Han S Yoo, and Joun G Kwak. Numerical simulations of the autorotative flight of the real and artificial maple seeds. volume 1, pages 1–17, 2019.

[90] Camilo Herrera, Mariana Correa, Valentina Villada, Juan D Vanegas, Juan G García, César Nieto-Londoño, and Julián Sierra-Pérez. Structural design and manufacturing process of a low scale bio-inspired wind turbine blades. *Composite Structures*, 208:1–12, 2019.

[91] Mariana Correa-Álvarez, Valentina Villada-Quiceno, Julián Sierra-Pérez, Juan Guillermo J.G. García-Navarro, and César Nieto-Londoño. Structural design of carbon/epoxy bio-inspired wind turbine blade using fluid/structure simulation. *International Journal of Energy Research*, 40(13):1832–1845, oct 2016.

[92] Gunnary León Finalé. Flight performance of Triplaris americana (Polygonacea) fruits accordingto their morphology. *Revista del Jardín Botánico Nacional*, 37:47–51, mar 2016.

[93] Thammarong Eadkong, Pimchanok Pimton, Punsiri Dam-O, and Phongpichit Channuie. Unraveling the vertical motion of Dipterocarpus alatus seed using Tracker. *Physica Scripta*, 95(5):055003, may 2020.

[94] Celso Ladera and Pedro Pineda. The physics of the spectacular flight of the Triplaris samaras. *Latin-American Journal of Physics Education*, 3(3):11, 2009.

[95] Stanley Osher and Ronald P Fedkiw. Level Set Methods: An Overview and Some Recent Results. *Journal of Computational Physics*, 169(2):463–502, may 2001.

[96] André Massing, Mats G. Larson, and Anders Logg. Efficient Implementation of Finite

Element Methods on Nonmatching and Overlapping Meshes in Three Dimensions. *SIAM Journal on Scientific Computing*, 35(1):C23–C47, jan 2013.

[97] Qin Yang, Zhiguo Zhang, Mingyue Liu, and Jing Hu. Numerical Simulation of Fluid Flow inside the Valve. *Procedia Engineering*, 23(February):543–550, 2011.

[98] Yared Alemu and Danny Bluestein. Flow-induced Platelet Activation and Damage Accumulation in a Mechanical Heart Valve: Numerical Studies. *Artificial Organs*, 31(9):677–688, sep 2007.

[99] M. D. De Tullio, A. Cristallo, E. Balaras, and R. Verzicco. Direct numerical simulation of the pulsatile flow through an aortic bileaflet mechanical heart valve. *Journal of Fluid Mechanics*, 622:259–290, mar 2009.

[100] Li-ning Liu, Xing-shen Wang, Guang-sheng Du, Zheng-gang Liu, and Li Lei. Transient aerodynamic characteristics of vans during the accelerated overtaking process. *Journal of Hydrodynamics*, 30(2):357–364, apr 2018.

[101] B Basara, S Jakirlić, F Aldudak, and C Tropea. Truck Interference Effects on a Car during an Overtaking Manoeuvre: A Computational Study. *Notes on Numerical Fluid Mechanics and Multidisciplinary Design*, pages 611–619, 2010.

[102] Tran Ngoc Tu. Numerical simulation of propeller open water characteristics using RANSE method. *Alexandria Engineering Journal*, 58(2):531–537, jun 2019.

[103] Saswati Roy, Luca Heltai, and Francesco Costanzo. Benchmarking the immersed finite element method for fluid–structure interaction problems. *Computers & Mathematics with Applications*, 69(10):1167–1188, may 2015.

[104] Alexander N Brooks and Thomas J.R. Hughes. Streamline upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations. *Computer Methods in Applied Mechanics and Engineering*, 32(1-3):199–259, sep 1982.

[105] Thomas J.R. Hughes, Leopoldo P Franca, and Gregory M Hulbert. A new finite element formulation for computational fluid dynamics: VIII. The galerkin/least-squares method for advective-diffusive equations. *Computer Methods in Applied Mechanics and Engineering*, 73(2):173–189, may 1989.

[106] Ramon Codina, Javier Principe, Oriol Guasch, and Santiago Badia. Time dependent subscales in the stabilized finite element approximation of incompressible flow problems. *Computer Methods in Applied Mechanics and Engineering*, 196(21-24):2413–2430, apr 2007.

[107] Matias Avila, Javier Principe, and Ramon Codina. A finite element dynamical nonlinear subscale approximation for the low Mach number flow equations. *Journal of Computational Physics*, 230(22):7988–8009, sep 2011.

[108] Camilo Andrés Bayona Roa, Joan Baiges, and Ramon Codina. Variational multi-scale finite element approximation of the compressible Navier-Stokes equations. *International Journal of Numerical Methods for Heat & Fluid Flow*, 26(3/4):1240–1271, may 2016.

[109] Charles S. Peskin. Numerical analysis of blood flow in the heart. *Journal of Computational Physics*, 25(3):220–252, nov 1977.

[110] Vito Pasquariello, Georg Hammerl, Felix Örley, Stefan Hickel, Caroline Danowski, Alexander Popp, Wolfgang A. Wall, and Nikolaus A. Adams. A cut-cell finite volume – finite element coupling approach for fluid–structure interaction in compressible flow. *Journal of Computational Physics*, 307:670–695, feb 2016.

[111] Fei Xu, Dominik Schillinger, David Kamensky, Vasco Varduhn, Chenglong Wang, and Ming-Chen Hsu. The tetrahedral finite cell method for fluids: Immersogeometric analysis of turbulent flow around complex geometries. *Computers & Fluids*, 141:135–154, dec 2016.

[112] F. Auricchio, F. Brezzi, A. Lefieux, and A. Reali. An "immersed" finite element method based on a locally anisotropic remeshing for the incompressible Stokes problem. *Computer Methods in Applied Mechanics and Engineering*, 294:428–448, sep 2015.

[113] Ramon Codina. Comparison of some finite element methods for solving the diffusion-convection-reaction equation. *Computer Methods in Applied Mechanics and Engineering*, 156(1-4):185–210, apr 1998.

[114] E. Castillo and R. Codina. Dynamic term-by-term stabilized finite element formulation using orthogonal subgrid-scales for the incompressible Navier–Stokes problem. *Computer Methods in Applied Mechanics and Engineering*, 349:701–721, jun 2019.

[115] A. Villota and Ramon Codina. Approximation of the scalar convection-diffusion-reaction equation with stabilized finite element formulations of high order. *Revista Internacional de Métodos Numéricos para Cálculo y Diseño en Ingeniería*, 35(1), feb 2019.

[116] Tom Andreas Nærland. *Geometry decomposition algorithms for the Nitsche method on unfitted geometries*. PhD thesis, University of Oslo, 2014.

[117] John W. Barrett and Charles M. Elliott. Finite element approximation of the Dirichlet problem using the boundary penalty method. *Numerische Mathematik*, 49(4):343–366, jul 1986.

[118] J. Nitsche. Über ein Variationsprinzip zur Lösung von Dirichlet-Problemen bei Verwendung von Teilräumen, die keinen Randbedingungen unterworfen sind. *Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg*, 1971.

[119] B. Schott and W.A. Wall. A new face-oriented stabilized XFEM approach for 2D and 3D incompressible Navier–Stokes equations. *Computer Methods in Applied Mechanics and Engineering*, 276:233–265, jul 2014.

[120] Patrick J. Roache. Code Verification by the Method of Manufactured Solutions. *Journal of Fluids Engineering*, 124(1):4–10, mar 2002.

[121] M. Schäfer, S. Turek, F. Durst, E. Krause, and R. Rannacher. Benchmark Computations of Laminar Flow Around a Cylinder. In *Flow Simulation with High-Performance Computers II, Volume 52 of Notes on Numerical Fluid Mechanics, Vieweg*, volume 52, pages 547–566. 1996.

[122] Toshiyuki Nakata and Hao Liu. A fluid–structure interaction model of insect flight with flexible wings. *Journal of Computational Physics*, 231(4):1822–1847, feb 2012.

[123] Jie Liu. A second-order stable explicit interface advancing scheme for FSI with both rigid and elastic structures and its application to fish swimming simulations. *Computers & Fluids*, 118:274–292, sep 2015.

[124] Matthew Hirschhorn, Vakhtang Tchantchaleishvili, Randy Stevens, Joseph Rossano, and Amy Throckmorton. Fluid–structure interaction modeling in cardiovascular medicine – A systematic review 2017–2019. *Medical Engineering & Physics*, (xxxx):1–13, feb 2020.

[125] K. Yusuf Billah and Robert H. Scanlan. Resonance, Tacoma Narrows bridge failure, and undergraduate physics textbooks. *American Journal of Physics*, 59(2):118–124, feb 1991.

[126] D T K Hoang, S V Pham, K N Tran, C D Nguyen, and K P Nguyen. Aeroelastic Analysis on Wing Structure Using Immersed Boundary Method. In Hung Nguyen-Xuan, Phuc Phung-Van, and Timon Rabczuk, editors, *Proceedings of the International Conference on Advances in Computational Mechanics 2017*, pages 783–792, Singapore, 2018. Springer Singapore.

[127] Ming-Chen Hsu and Yuri Bazilevs. Fluid–structure interaction modeling of wind turbines: simulating the full machine. *Computational Mechanics*, 50(6):821–833, dec 2012.

[128] Li Yu, Han Cheng, Ya'nan Zhan, and Shaoteng Li. Study of parachute inflation process using fluid–structure interaction method. *Chinese Journal of Aeronautics*, 27(2):272–279, apr 2014.

[129] Ramon Codina. Stabilization of incompressibility and convection through orthogonal sub-scales in finite element methods. *Computer methods in applied mechanics and engineering*, 190(13-14):1579–1599, 2000.

[130] Martin S. Alnæs, Jan Blechta, Johan Hake, August Johansson, Benjamin Kehlet, Anders Logg, Chris Richardson, Johannes Ring, Marie E. Rognes, and Garth N. Wells. The fenics project version 1.5. *Archive of Numerical Software*, 3(100), 2015.

[131] Ramon Codina. A stabilized finite element method for generalized stationary incompressible flows. *Computer Methods in Applied Mechanics and Engineering*, 190(20-21):2681–2706, 2001.

[132] Stefan Turek and Jaroslav Hron. Proposal for Numerical Benchmarking of Fluid-Structure Interaction between an Elastic Object and Laminar Incompressible Flow. In *Fluid-Structure Interaction*, volume 53, pages 371–385. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006.

[133] Bruce R Munson, Theodore H Okiishi, Wade W Huebsch, Rothmayer, and Alric P. *Fundamentals of Fluid Mechanics Seventh Edition*. 2013.

[134] M. Rahmani and A. Wachs. Free falling and rising of spherical and angular particles. *Physics of Fluids*, 26(8):083301, aug 2014.

[135] Ulrich Küttler and Wolfgang A. Wall. Fixed-point fluid–structure interaction solvers with dynamic relaxation. *Computational Mechanics*, 43(1):61–72, dec 2008.

[136] Antoine Placzek, Jean-François Sigrist, and Aziz Hamdouni. Numerical simulation of an oscillating cylinder in a cross-flow at low Reynolds number: Forced and free oscillations. *Computers & Fluids*, 38(1):80–100, jan 2009.

[137] Hélène Persillon and Marianna Braza. Physical analysis of the transition to turbulence in the wake of a circular cylinder by three-dimensional Navier-Stokes simulation. *Journal of Fluid Mechanics*, 1998.

[138] S. Mittal and V. Kumar. Flow-induced vibrations of a light circular cylinder at reynolds numbers 103 to 104. *Journal of Sound and Vibration*, 2001.

[139] D Shiels, A Leonard, and A Roshko. Flow-Induced vibration of a circular cylinder at limiting structural parameters. *Journal of Fluids and Structures*, 15(1):3–21, jan 2001.

[140] T. Kinsey and G. Dumas. Parametric study of an oscillating airfoil in a power-extraction regime. *AIAA Journal*, 46(6):1318–1330, 2008.

[141] Madeleine Seale and Naomi Nakayama. From passive to informed: mechanical mechanisms of seed dispersal. *New Phytologist*, 225(2):653–658, jan 2020.

[142] Mukul Rao, Dhruv C. Hoysall, and Jagadeesh Gopalan. Mahogany seed - a step forward in deciphering autorotation. *Current Science*, 2014.

[143] Martin Alnæs, Jan Blechta, Johan Hake, August Johansson, Benjamin Kehlet, Anders Logg, Chris Richardson, Johannes Ring, Marie E Rognes, and Garth N Wells. The fenics project version 1.5. *Archive of Numerical Software*, 3(100), 2015.

[144] Christian Valdemar Hansen, Anders Logg, and Carl Lundholm. Simulation of flow and view with applications in computational design of settlement layouts. *arXiv preprint arXiv:1610.02277*, 2016.

[145] Jean Rabault, Richard A. Fauli, and Andreas Carlson. Curving to Fly: Synthetic Adaptation Unveils Optimal Flight Performance of Whirling Fruits. *Physical Review Letters*, 122(2):1–9, 2019.

[146] Julián Sierra-Pérez. Diseño de aerogenerador basado en formas de la naturaleza. Technical report, Universidad Nacional, 2005.

[147] Natalia Gómez-Velásquez and Sergio Hincapié-Perilla. Bioinspiración para el diseño básico de la hélice de un aerogenerador. Technical report, Universidad Pontificia Bolivariana, 2009.

[148] Anna Ciupak, Agata Dziwulska-Hunek, Bożena Gładyszewska, and Anita Kwaśniewska. The relationship between physiological and mechanical properties of Acer platanoides L. and Tilia cordata Mill. leaves and their seasonal senescence. *Scientific Reports*, 9(1):1–10, 2019.

[149] Jean François Louf, Logan Nelson, Hosung Kang, Pierre Ntoh Song, Tim Zehnbauer, and Sunghwan Jung. How wind drives the correlation between leaf shape and mechanical properties. *Scientific Reports*, 8(1):1–7, 2018.

[150] Ramon Codina. A stabilized finite element method for generalized stationary incompressible flows. *Computer Methods in Applied Mechanics and Engineering*, 190(20-21):2681–2706, 2001.

[151] Erik Burman and Peter Hansbo. Fictitious domain finite element methods using cut elements: II. A stabilized Nitsche method. *Applied Numerical Mathematics*, 62(4):328–341, apr 2012.

[152] Suraj Bansal, Gavin K. Ananda, and Michael S. Selig. Development of an aerodynamic analysis methodology for segmented ultralight morphing rotors. *35th AIAA Applied Aerodynamics Conference, 2017*, (June), 2017.

[153] Martin Hansen. *Aerodynamics of Wind Turbines*. Routledge, may 2015.

[154] Carol K. Augspurger. Morphology and dispersal potential of wind-dispersed diaspores of neotropical trees. *American Journal of Botany*, 73(3):353–363, mar 1986.

[155] D. Lentink, W. B. Dickson, J. L. Van Leeuwen, and M. H. Dickinson. Leading-edge vortices elevate lift of autorotating plant seeds. *Science*, 324(5933):1438–1440, 2009.

[156] Sanjay P. Sane. The aerodynamics of insect flight. *Journal of Experimental Biology*, 206(23):4191–4208, dec 2003.

[157] Florian T. Muijres, L. Christoffer Johansson, and Anders Hedenström. Leading edge vortex in a slow-flying passerine. *Biology Letters*, 8(4):554–557, aug 2012.

[158] Mostafa R. A. Nabawy and William J. Crowther. The role of the leading edge vortex in lift augmentation of steadily revolving wings: a change in perspective. *Journal of The Royal Society Interface*, 14(132):20170159, jul 2017.

[159] Iván Herráez, Buşra Akay, Gerard J. W. van Bussel, Joachim Peinke, and Bernhard Stoevesandt. Detailed analysis of the blade root flow of a horizontal axis wind turbine. *Wind Energy Science*, 1(2):89–100, jul 2016.