Development of a multiphysics solver for complex coupled problems involving thin shells: fluid-structure-electrophysiology interaction

Original Citation:
Development of a multiphysics solver for complex coupled problems involving thin shells: fluid-structure-electrophysiology interaction / Nitti, Alessandro. - ELETTRONICO. - (2021). [10.60576/poliba/iris/nitti-alessandro_phd2021]

Published version
Politecnico di Bari
DOI:10.60576/poliba/iris/nitti-alessandro_phd2021

POLITECNICO DI BARI

Department of Mechanics, Mathematics and Management
Mechanical and Management Engineering Ph.D. Progam

Development of a multiphysics solver for complex coupled problems involving thin shells: fluid-structure-electrophysiology interaction

---

Author:
Alessandro Nitti

Supervisor:
Prof. Marco D. de Tullio

Final Dissertation

Course XXXIII, 01/11/2017-31/10/2020

Coordinator of Ph.D. Program: Prof. G. P. Demelio

POLITECNICO DI BARI

Department of Mechanics, Mathematics and Management
Mechanical and Management Engineering Ph.D. Progam

SSD: ING/IND-06: Fluid Dynamics

Development of a multiphysics solver for complex coupled problems involving thin shells: fluid-structure-electrophysiology interaction

Author:
Alessandro Nitti

Alessandro Nitti

Supervisor:
Prof. Marco D. de Tullio

Marco Donato de Tullio

Referees:

Prof. Markus Uhlmann, Karlsruhe Institute of Technology, Germany

Prof. Ricardo Ruiz Baier, Monash University at Melbourne, Australia

Final Dissertation

Course XXXIII, 01/11/2017-31/10/2020

Coordinator of Ph.D. Program: Prof. G. P. Demelio

Abstract

Development of a multiphysics solver for complex coupled problems involving thin shells: fluid-structure-electrophysiology interaction

by Alessandro Nitti

The present work is devoted to the development of a multiphysics solver for simulating two classes of coupled problems.

A computational framework is designed to accurately predict the elastic response of thin shells undergoing large displacements induced by local hydrodynamic forces, as well as to resolve the complex fluid pattern arising from its interaction with an incompressible fluid. Within the context of partitioned algorithms, two different approaches are employed for the fluid and structural domain. The fluid motion is resolved with a pressure projection method on a Cartesian structured grid. The immersed shell is modeled by means of a NURBS surface, and the elastic response is obtained from a displacement-based Isogeometric Analysis relying on the Kirchhoff-Love theory. The two solvers exchange data through a direct-forcing Immersed Boundary approach, where the interpolation/spreading of the variables between Lagrangian and Eulerian grids is implemented with a Moving Least Squares approximation, which has proven to be very effective for moving boundaries. In this scenario, the isoparametric paradigm is exploited to perform an adaptive collocation of the Lagrangian markers, decoupling the local grid density of fluid and shell domains and reducing the computational expense. The convergence rate of the method is verified by refinement analyses, segregating the Eulerian/Lagrangian refinement, which confirms the expected scheme accuracy in space and time. The effectiveness of the method is then verified against different test–cases of engineering and biologic inspiration, involving fundamentally different physical and numerical conditions, namely: i) a flapping flag, ii) an inverted flag, iii) a clamped plate, iv) a buoyant seaweed in a free stream. Both strong and loose coupling approaches are implemented to handle different fluid-to-structure density ratios, providing accurate results.

In second instance, we propose an IGA approximation of the system of equations describing the propagation of an electrophysiologic stimulus over a thin cardiac tissue with the subsequent muscle contraction. The underlying method relies on the monodomain model for the electrical sub-problem. This requires the solution of a reaction-diffusion equation over a surface in the three-dimensional space. Exploiting the benefits of the high-order NURBS basis functions within a curvilinear framework, the method is found to reproduce complex excitation patterns with a limited number of degrees of freedom. Furthermore, the curvilinear description of the diffusion term provides a flexible and easy-to-implement approach for general surfaces.

The electrophysiological stimulus is converted into a mechanical load by means of the well-established active strain approach. The multiplicative decomposition of the deformation gradient tensor is grafted into the classical finite elasticity weak formulation, providing the necessary tensor expressions in curvilinear coordinates. The expressions derived provides what is needed to implement the active strain approach in standard finite-element solvers without resorting to dedicated formulations. Such a formulation is valid for general three-dimensional geometries and isotropic hyperelastic materials. The formulation is then restricted to Kirchhoff-Love shells by means of the static condensation of the material tensor. The purely elastic response of the structure is investigated with simple static test-cases of thin shells undergoing different active strain patterns. Eventually, various numerical tests performed with a staggered scheme illustrate that the coupled electromechanical model can capture the excitation-contraction mechanism over thin tissue and reproduce complex curvature variations.

Keywords: Multiphysics problems, Partitioned solvers, Fluid-structure interaction, Kirchhoff-Love shell, Incompressible flows, Electromechanical activation, Monodomain model, Active strain

Contents

List of Figures

List of Tables

1 Introduction

1.1 Coupled multiphysic solvers

The increase in computational capability for the purpose of scientific prediction or engineering design, makes the coupling of multiple physical phenomena very popular in the scientific community. The report by Brown et al. [1], *Scientific grand challenges: Crosscutting technologies for computing at the exascale*, emphasizes that:

“The issue of coupling models of different events at different scales and governed by different physical laws is largely wide open and represents an enormously challenging area for future research.”

Following the semantic definition provided in [2], a multiphysic system consists of more than one component governed by its own principle(s) for equilibrium, conservation or constitutive laws. The coupling may occur in the bulk, i.e. through source terms or constitutive relations active in the overlapping domains of the individual sub-problems, otherwise it may occur over an idealized interface that is lower dimensional, or a narrow buffer zone. Such a distinction represents a fundamental issue for analysts because each of the two coupling provide different numerical challenges to be addressed.

The leading motivations for investing computational resources on multiphysic computing are to relax assumptions of decoupling and to provide insights on the coupling mechanisms themselves [2]. However, the claim for a coupled multiphysics might not correspond to actual mathematically coupled solvers. Operator splitting or segregated solvers are often the choice to overcome computational and algorithmic difficulties. However, coupling individual simulations may introduce limitations on stability, accuracy, or robustness that are more severe than the limitations imposed by the individual components. Hence, care must be taken in the coupling of segregated solvers.

On the other hand, many reasons push analysts to pull back on monolithic solvers. These often rely on algebraic paradigms whose linearization brings a discrete equation in which individual components are represented by diagonal blocks and the coupling between them, as off-diagonal blocks [2]. Such formulations often let the analysts face serious numerical issues such as ill-conditioned matrices, unpractical numerical stability limits, complications in the parallel implementation, problematic handling of the coupling variables.

In the present work we conceptualize two coupled solvers in separate frameworks. First a Fluid-Structure Interaction (FSI) solver is presented and analyzed, then a Electromechanical activation solver is proposed. Both coupled problems are resolved in a segregated arrangement. The mathematical models and the algorithmic formulations are presented, with focus on numerical aspects, including coupling issues.

1.2 Fluid-structure interaction: objectives and contribution

The development of accurate numerical techniques to investigate the interaction of a shell structure and an incompressible flow is still a challenging task in computational mechanics. In this connection, the fluid-structure interaction issue consists of resolving: i) the elastic response of a shell undergoing finite displacements induced by local hydrodynamic forces, ii) the simultaneous flow–field evolution arising from the interaction with the body interface. The inherently nonlinear nature of these applications requires robustness, minimal numerical dissipation and accuracy in time and space, to genuinely capture the desired FSI mechanism.

A partitioned approach is presented in this work, with the aim of providing the most suitable solution technique for the fluid and structural sub–problems. The elastic problem is resolved by means of a finite element formulation in the Isogeometric Analysis (IGA) context [3]. IGA has shown superior convergence properties compared to classical finite elements on a per degree-of-freedom basis [4, 5], resulting in a computationally attractive option within a partitioned–FSI framework. Moreover, the NURBS-based IGA [6, 7], employed in the present work, has proven to be perfectly suitable for C^1 conforming thin shells [8, 9]. The flow field is instead resolved in a Finite Difference (FD) environment. Beyond the numerical simplicity of this approach, a low–order FD approximation of spatial derivatives on a staggered grid does preserve kinetic energy and circulation in absence of time-differencing errors and viscosity [10, 11]. This makes the FD approximation convenient for the resolution of separated flow regions and complex vortex dynamics at moderate Reynolds numbers [12]. A reasonable computational expense would be required to perform Direct Numerical Simulations (DNS) of such flows with an adequate grid scaling. To the knowledge of the authors, this represents the first implementation of an IGA tool into a FD environment for FSI applications. Successful partitioned algorithms for FSI applications can be found in [13, 14, 15, 16]. With minimal interweaving of solvers data structure, the coupling operations might be simplified, but the ensuing time-staggered procedure can lead to numerical instabilities in case of low phase–density ratios [17, 18, 19].

The FSI tool presented here is devoted to the simulation of FSI at moderate and low Reynolds number regimes, which represent a broad range of real cases occurring in many biologic and engineering areas. Different numerical features of the present approach are revealed by investigating the interface conditions and the variable transfer. Then, the numerical framework is verified against several three-dimensional cases, involving different numerical and physical challenges: the flapping of a hinged flag, the dynamic regimes of an inverted flag, the flow-induced vibrations of a plate, the reconfiguration of a buoyant seaweed. Moreover, an accuracy analysis of the overall procedure is provided with respect to the pinned flag case.

The author would like to clarify that limited analyses are presented on the unplugged fluid dynamic- and shell-solver, since they have been developed and extensively validated by other authors. To this extent the corresponding references are provided.

1.3 Electrophysiologic stimulation: objectives and contribution

The second part of this works is devoted to the development of a Galerkin solver which aims at simulating the coupled electromechanical excitation of a thin cardiac muscle or a tissue slice. It potentially represents an attractive tool for computational biologists, to assist and optimize the expensive experimental practice. A numerical framework for the electromechanical activation of cardiac tissues can be exploited for variegated purposes, from the replication of pathological states [20] to the possibility to perform clinical studies with patient specific simulations of isolated muscular films [21]. On the other hand, numerical simulations can help engineers in the realization of actuators and soft robotic devices for bio-inspired locomotion built with cultured animal myocardial tissue [22].

In this scenario we propose a segregated approach, in which both the electrophysiologic and mechanical problem is resolved by means of IGA. The propagation of an electrophysiological stimulus is described by the monodomain model, consisting of a reaction-diffusion partial differential equation accompanied by a set of nonlinear ordinary differential equation describing the evolution of the ionic current fluxes. A numerically effective discretization approach is proposed, both in time and space. A novel contribution lays in the IGA implementation of the diffusion term where a local curvilinear description of the differential operators is exploited to offer a flexible approach for arbitrary surfaces in the three-dimensional space. Moreover, the effectiveness of the IGA approximation is tested with different ionic current integration strategies. Eventually, the physical accuracy of the method is validated against benchmark tests from recent literature, whereas the numerical implementation is verified by means of convergence analyses and numerical experiments.

The determination of the loading condition arising from the electrophysiologic excitation requires the knowledge of the mechanical effect deriving from the contraction of the sub-cellular components in the muscle tissue. A common practice in computational electrophysiology, is the approximation of the mechanical response of the tissue with a continuum model, based on the concept of active strain or active stress. In the present work the active strain approach [23] is embedded in the weak formulation for Kirchhoff-Love shells. This imply that the fiber contraction is rewritten in the mechanical balance of forces as a prescribed deformation, rather than as an additive stress contribution. This approach allows a direct incorporation of the micro-level information on the fiber contraction in the kinematics, without the intermediate transcription of their role in terms of the stress [24]. To the knowledge of the author this represents the first implementation of the active strain approach in a shell model. Such and approach entails the multiplicative decomposition of the tensor deformation gradient, where the active part is provided by a phenomenological relation with the transmembrane potential. In our numerical framework the active strain treatment is embedded in a classical total Lagrangian weak formulation for finite elasticity, whereas the majority of the analysts make use of an Euler-Lagrange formulation. The necessary tensor calculations are provided in order to facilitate the implementation of the active strain treatment in a common finite elasticity solver. Furthermore, all quantities are presented in local curvilinear coordinates for general three-dimensional solids. In second instance the shell hypotheses are enforced by the static condensation of the material tensor. The passive elastic response of the unplugged active strain solver is investigated by numerical experiments and convergence analyses. Eventually, we tested the coupled electromechanical method with two simple cases providing in-plane and out-of-plane deformations.

1.4 Outlines

The thesis is outlined as follows. Chapters 3-6 are dedicated to the fluid-structure interaction solver, while chapters 7-9 are for the electromechanical solver:

> *Chapter 2:* The basics of NURBS-based Isogeometric Analysis are reviewed. Brief remarks on the NURBS surface representation are provided. The IGA paradigm is presented with emphasis on the differences to traditional FEA;
>
> *Chapter 3:* The formulation of the isogeometric Kirchhoff-Love (KL) shell is presented.

First, the geometric approximation of the KL model are provided. Subsequently, the compatibility and constitutive equations are drawn. Eventually, the spatial and temporal discretization is briefly discussed.

Chapter 4: The governing equations and the numerical method used for the Direct Numerical Solution (DNS) of the incompressible Navier-Stokes equations are described;

Chapter 5: A brief overview of most widespread Immersed Boundary methods for incompressible flows is provided. The Immersed Boundary procedure employed in the present work is described in detail;

Chapter 6: The fluid- and solid-phase solvers are verified against simple benchmark cases. The FSI tool is then tested with numerical experiments to verify the accuracy of the methodology. Several numerical results are provided, covering a wide range of numerical and physical challenges;

Chapter 7: The physiologic excitation of cardiac muscle is described from the macroscopic perspective, and the main issues concerning the continuum-level modeling are addressed. The bidomain and monodomain models, are derived from electrostatic arguments:

Chapter 8: The implementation of the monodomain model over a surface, within an IGA framework is described. Such an implementation is assessed by means of convergence analyses, as well as it is tested against physical accuracy based on the depolarization front velocity. The curvilinear description of the diffusion effects is checked by comparing the results against a classical two-dimensional finite element method.

Chapter 9: An active strain model for the activation is presented and efficiently embedded in a total Lagrangian IGA formulation. The enforcement of the KL shell approximation is provided by static condensation of the material tensor. The unplugged active strain solver is assessed by convergence analyses. Eventually, two simple coupled electromechanical tests are presented and analyzed.

2 NURBS-based Isogeometric Analysis

2.1 The concept of Isogeometric Analysis

The Isogeometric Analysis (IGA) was developed by Hughes et al. [6, 3] as an exact-geometry, cost-effective alternative to classical Finite Element Analysis (FEA). The initial motivation behind the IGA was the desire to reduce the engineering time taken for the generation of a suitable computational mesh, by using a single mathematical representation for both Computer Aided Design (CAD) and analysis.

In traditional FEA, low-order Lagrange polynomials (mostly linear) are used as basis functions for the analysis, whereas CAD technologies make use of splines-like objects since they provide enhanced control on local derivatives. Thus, the geometric model arising from the CAD stage needs to be converted into an object suitable for the analysis. The CAD model is approximated into a piecewise-linear mesh, whose quality of the approximation depends on the mesh density. The meshing phase is a time-consuming process which introduces a geometric approximation, and it presents several challenges for analysts, who have to take care of the mesh regularity and smoothness. The isogeometric analysis allows to bypass the meshing stage by exploiting the basis functions used to define the geometry as test functions for the solution field. In classical FEA instead, the basis functions which define the polygonal mesh are exploited as basis functions for analysis (isoparametric paradigm). Furthermore, the higher regularity of IGA basis functions with respect to FEA provides a superior convergence rate in all the fields in which high continuity plays a crucial role, e.g., the study of structural vibrations [4, 25], the analysis of nearly incompressible solids [26], biologic structural elements [27, 28], turbulent flows and fluid–structure interaction [29]. A remarkable example has been presented in [27], and it is reported here to provide a quantitative analogy between IGA and FEA. IGA was tested against classic FEA in a bending analysis on a endovascular stent, and it provided more accurate results with at least one order of magnitude fewer degrees of freedom than classical FEA, both on local and global quantities (see Fig.2.1).

All functions employed for a CAD model could be used as basis functions for IGA, provided that they fulfill the necessary conditions for basis functions, which are listed in the following section. NURBS (Non-Uniform Rational B-Splines) are the most widespread technology in today's CAD programs and their basis functions fulfill the mathematical requirements of test functions, therefore NURBS represent a straightforward choice for analysis. Further arguments which led the author to the choice of a NURBS-based IGA solver are illustrated in the next sections.

2.2 NURBS surface representation

NURBS are a generalization of B-Splines and most of the features of the B-Spline technology are extended to NURBS, therefore some introductory considerations about B-Splines are provided.

A B-Spline is a non-interpolating, piecewise polynomial curve. It is defined by a set of control points $\mathbf{P}_i$, $i = 1...n$, a polynomial degree p and a so called knot vector $\Xi = [\xi_1, \xi_2, ..., \xi_{n+p+1}]$.

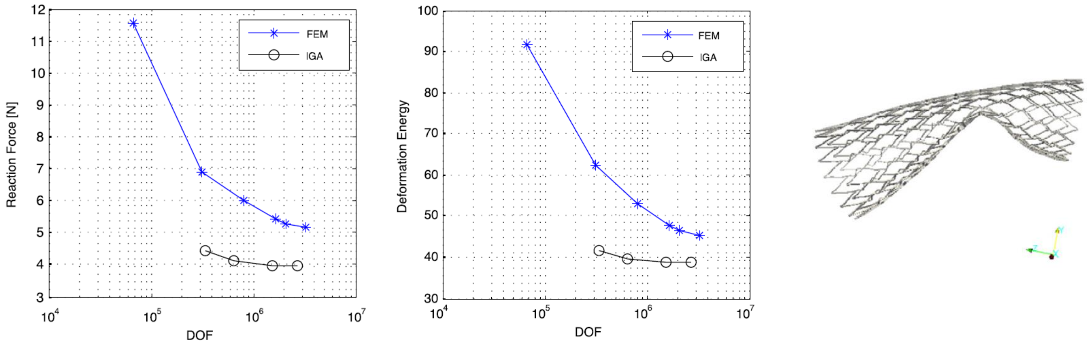


FIGURE 2.1: Convergence plots for a nonlinear bending analysis of a carotid artery stent, presented in [27]. The deformed configuration is reported on the right panel, whilst in the other panels are reported the convergence plot of a local and an integral quantity.

The knot vector is a set of parametric coordinates ξ_i in non-descending order which divide the B-Spline into sections. A B-Spline basis function is C^∞ continuous inside a knot span, i.e. between two distinct knots, and C^{p-1} continuous at a single knot. A knot value can have multiplicity greater than one. At a knot of multiplicity k the continuity is C^{p-k}. If the first and the last knot has multiplicity $p+1$, the knot vector is called open. In a B-Spline with an open knot vector the first and the last control point are interpolated and the curve is tangential to the control polygon at the start and the end of the curve. Open knot vectors are standard in CAD applications and are assumed for the remainder of this work.

B-Spline basis functions are computed by the Cox-deBoor recursion formula [30]. It starts for $p = 0$ with:

$$N_{i,0}(\xi) = \begin{cases} 0 & \xi_i \leq \xi < \xi_{i+1}\,, \\ 1 & \text{otherwise}\,. \end{cases} \tag{2.1}$$

For $p \geq 1$:

$$N_{i,p}(\xi) = \frac{\xi - \xi_i}{\xi_{i+p} - \xi_i} N_{i,p-1}(\xi) + \frac{\xi_{i+p+1} - \xi}{\xi_{i+p+1} - \xi_{i+1}} N_{i,p-1}(\xi)\,. \tag{2.2}$$

From this formulation some important properties of B-Spline basis functions can be deduced:

- Local support. A basis function $N_{i,p}(\xi)$ is non-zero only in the interval $[\xi_i, \xi_{i+p+1}]$;
- Partition of unity: $\sum_{i=1}^{n} N_{i,p}(\xi) = 1$;
- Non-negativity: $N_{i,p}(\xi) \geq 0$;
- Linear independence: $\sum_{i=1}^{n} \alpha_i N_{i,p}(\xi) = 0$, with $alpha_i$ real coefficients;

A B-Spline surface is computed by the tensor product of B-Spline curves in two parametric dimensions ξ and η. It is defined by a net of $n \times m$ control points, two knot vectors Ξ and $\mathcal{H}$, two polynomial degrees p and q (which do not need to be equal), and the corresponding basis functions $N_{i,p}(\xi)$ and $M_{j,q}(\eta)$. It is described by the formula:

$$\mathbf{S}(\xi, \eta) = \sum_{i=1}^{n} \sum_{j=1}^{m} N_{i,p} M_{j,q}(\xi) \mathbf{P}_{ij}\,. \tag{2.3}$$

NURBS are non uniform rational B-Splines. For NURBS each control point has additionally to its coordinates an individual weight w_i. Such a point $\mathbf{P}_i(x_i, y_i, z_i, w_i)$ can be represented with homogeneous coordinates $\mathbf{P}_i^w(w_i x_i, w_i y_i, w_i z_i, w_i)$ in a projective $\mathbb{R}^4$ space if one consider a NURBS curve $\mathbf{C}(\xi)$ as the projection of a B-Spline in $\mathbb{R}^4$ with homogeneous control points onto $\mathbb{R}^4$ [30]:

$$\mathbf{C}(\xi) = \frac{\sum_{i=1}^{n} N_{i,p}(\xi) w_i \mathbf{P}_i}{\sum_{i=1}^{n} N_{i,p}(\xi) w_i} \tag{2.4}$$

Then, a NURBS surface is defined as:

$$\mathbf{S}(\xi, \eta) = \frac{\sum_{i=1}^{n} \sum_{j=1}^{m} N_{i,p}(\xi) M_{j,q}(\eta) w_{i,j} \mathbf{P}_{i,j}}{\sum_{k=1}^{n} \sum_{l=1}^{m} N_{k,p}(\xi) M_{l,q}(\eta) w_{k,l}}. \tag{2.5}$$

NURBS are able to exactly represent some important geometric entities, like e.g. conic sections (i.e. circles, cylinders, spheres etc.). Moreover, a B-Spline is a special case of a NURBS where all weights are equal and is therefore automatically contained in all the derivations for NURBS-based elements. Fig. 2.3 shows an example of a NURBS patch defined with cubic basis functions and open knot vectors, also illustrated within the index space. The patch control net is identified by red dots. Due to the open knot vectors the control points at the vertices of the surface are interpolatory. The black lines laying on the surface mark the knots which divide the surface into elements.

The intrinsic nature of NURBS patches entails interpolatory control points at the patch edges, therefore only C^0 continuity is obtained in those edges. The C^0 continuity means that only the primitive field variables are coupled at the patch interface, but not their derivative. For instance, in a mechanical sense, the C^0 continuity represents a hinge connection, where the respective bending moments cannot be transferred. It represents an obstacle in the original intent of exploiting the same geometry for CAD representation and analysis. This is often verified in the context of NURBS-based IGA, because the tensor-product nature of NURBS patches makes the representation of even simple object impossible without inner interpolatory control points. For instance, any revolution surface have to be represented with C^0 edges. However, the enforcement of the local suitable continuity has been subject of intensive research since the first IGA applications, and several solutions have been presented in literature [31, 32, 33].

2.3 The IGA element

The NURBS representation, with the features illustated above, provides remarkable enhancements in the analysis. Following the most widespread nomenclature, a NURBS element is defined by the knot span. Each element is thus limited by knots of arbitrary multiplicity. However, differently from FEA, the knots do not constitute the nodes of the analysis, which are defined instead at the control points. They carry the degrees of freedom for the analysis and boundary conditions are applied to them.

The non-interpolatory feature of NURBS control points provide superior accuracy in the representation of thin layers, discontinuities and steep gradients in the solution of linear and nonlinear problems, whereas it is well known that typical finite element shape functions, such as Lagrange polynomials, oscillate in attempting to fit nearly-discontinuous data. In this respect, an example proposed by [6] is reported in Fig. 2.2 for a clarity purpose. In the left panel Lagrange polynomials of orders three, five, and seven interpolate a discontinuity represented by eight data points in a two-dimensional space. As the order is increased, the amplitude of the oscillations increases. NURBS behave very differently when the data are viewed as control points. In the right panel the

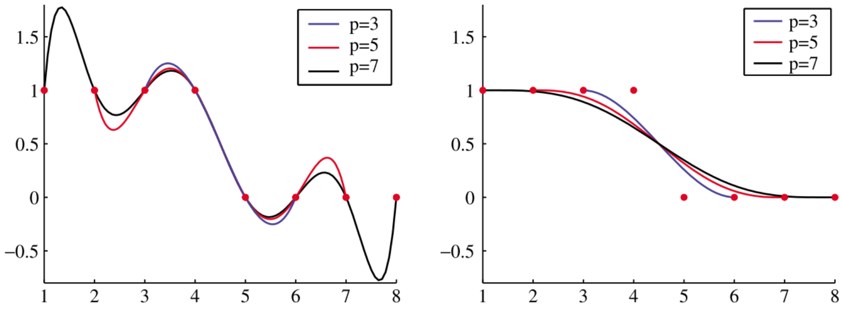


FIGURE 2.2: Lagrange interpolation oscillates when faced with discontinuous data (left panel). NURBS functions exhibit the variation diminishing property for the same data (right panel). Figure adapted from [6].

NURBS curves exhibit a monotone trend, illustrating the so called *variation diminishing* property of NURBS.
In IGA, each element has its corresponding basis functions, which are not confined to one element but extend over the neighborhood elements. This is a very important difference to classical finite elements because it allows high continuity of shape functions over the element boundaries. The high-order nature of the basis functions generally results in higher accuracy compared to low-order elements. Furthermore, the high continuity in the element basis functions allows for a smoother representation of the solution derivatives. This features was found to provide more accurate derived physical quantities, such as strains and stresses. On the other hand one must consider that the elements are interconnected and not independent of each other, then an unconventional implementation strategy might be necessary.

2.4 Mesh refinement

Both h-refinement and p-refinement strategies assume a different connotation in NURBS-based IGA with respect to FEA. The h-refinement is performed by a knot insertion procedure, and the p-refinement is performed by degree-elevation of the basis functions [30]. It is worth noting that these refinement strategies do not change the topology of the NURBS patch. This means that in each refinement step, the geometry is represented exactly and therefore a refined mesh can be further refined without the necessity of going back to the original CAD model.
The knot insertion procedure allows the arbitrary subdivision of the elements in the computational domain. Nevertheless, the tensor-product definition of a NURBS patch lets any knot inserted in the ξ-direction extend over the whole patch in the η direction and vie-versa. Hence, pure local refinement is not possible for NURBS patches. Such a drawback has been addressed by using T-Splines as alternative representation, which are not confined to a tensor product structure [34].
The p-refinement provides a superior convergence rate in case of Gaussian quadrature rule at the cost of a minimal increase in degrees of freedom, since it produces a minimal increase in the number of control points with respect to h-refinement. p-refinement and h-refinement do not commute because while order elevation preserves the local continuities, the insertion of a knot decreases the continuity at its location. As a matter of fact, in common practice order elevation is always performed before knot insertion.

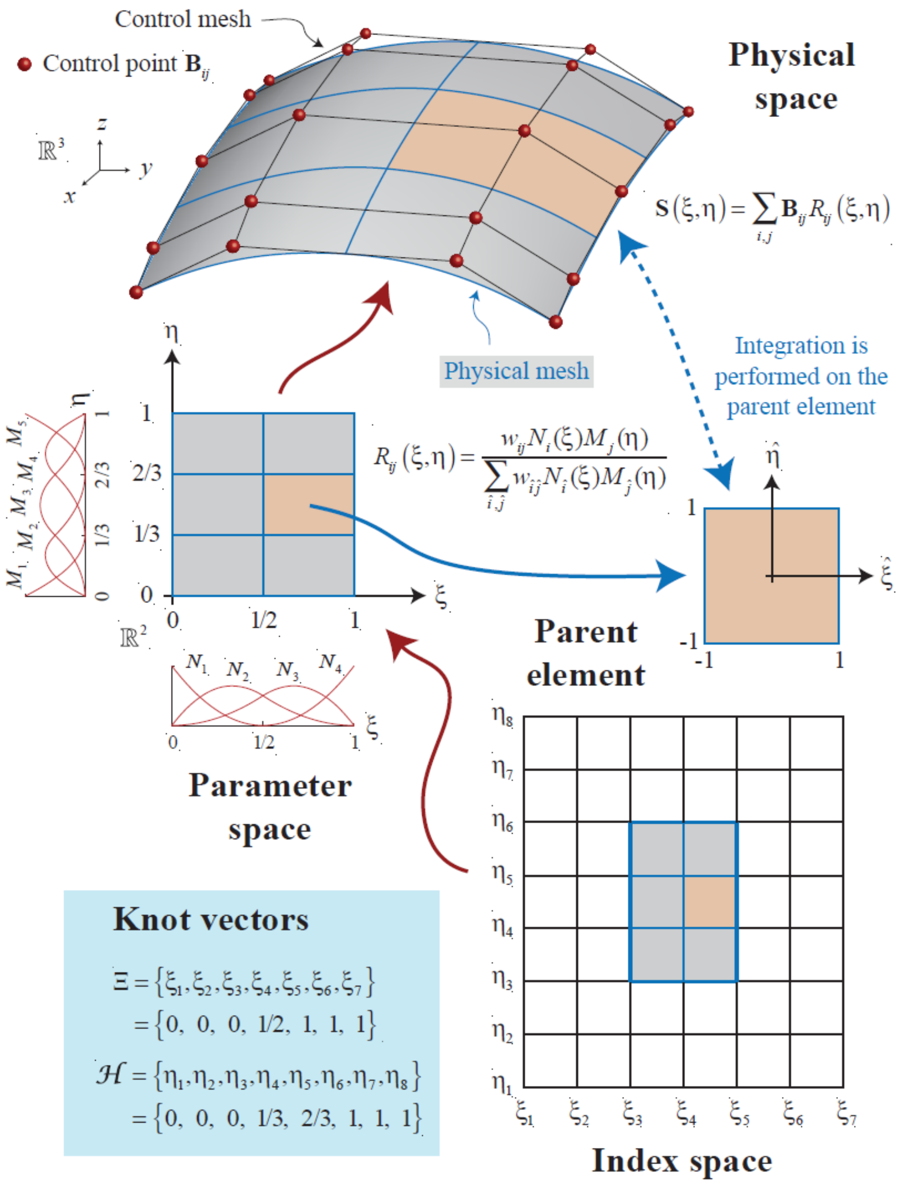


FIGURE 2.3: Schematic illustration of a NURBS surface with relevant spaces. Open knot vectors and quadratic C^1-continuous basis functions are used. Also depicted are C^1-quadratic basis functions determined by the knot vectors. Figure taken from [3].

2.5 Motivation

Most of the features of the NURBS-based IGA approach make this tool computationally efficient for the problems addressed in this work. In first instance, IGA provided a new effective framework for the discretization of high-order derivatives over the computational domain. Hence, the high continuity of the geometry matches the necessity of discretizing the second derivative of the curvature, which is required by the Kirchhoff-Love weak formulation. Since a minimum C^1 continuity can be easily achieved at the internal surface points, no specific treatment are needed to implement a Kirchhoff-Love shell model.

In second instance, the IGA approach is expected to provide superior performance in the approximation of the electropysiological excitation mechanisms. The excitation of an active tissue take place through the propagation of a depolarization front over a reactive domain. Such a phenomenon requires the solution of a reaction-diffusion equation with nearly-discontinuous scalar fields. The high continuity of NURBS basis function an the *variation diminishing* property of the IGA approach can lead to a superior accuracy in the front velocity, and reduced oscillations in the representation of the discontinuities, on a peer-degrees-of-freedom basis with respect to classic FEA.

According to these arguments, NURBS-based IGA is employed as analysis tool in the present work.

3 A Kirchhoff-Love shell model

3.1 Geometry of a Kirchhoff-Love shell

The Kirchhoff-Love (KL) shell theory belongs to the category of direct approaches, where the kinematics of the body is formulated from proper assumptions, and not from a degeneration of the three-dimensional continuum mechanics. Hence, the shell is treated as two-dimensional manifold in the three-dimensional space *ab initio*. All direct shell theories do neglect the normal strains and stresses in the thickness direction. Furthermore, the mechanics of a Kirchhoff-Love shell is founded on the additional assumptions:

- Any cross section is assumed to remain straight during deformation.
- Any cross section that is normal to the middle surface, remains normal to the middle surface in the deformed configuration.

A straight cross section in the deformed configuration entails a linear strain distribution through the thickness. In second instance, the assumption of normal cross section in the deformed configuration means that the shell can be completely represented by its middle surface. A geometric description of the middle surface with the relevant base vectors used in this work is presented in Fig. 3.1. From a mechanical perspective, the assumption of cross sections remaining normal to the midsurface means that transverse shear strains are neglected. Commonly this is considered a reasonable assumption for a shell slenderness ratio $r/h > 20$, with r being the local radius of curvature and h being the shell's thickness [35][36].

In the present chapter, these assumptions are applied to the kinematic, constitutive, and equilibrium equations of solid mechanics in total Lagrangian form. Italic letters a, A indicate scalars, lower case bold letters $\mathbf{a}$ indicate vectors, upper case bold letters $\mathbf{A}$ indicate second order tensors and upper case blackboard-bold letters $\mathbb{A}$ indicate fourth order tensors. The operators "$\cdot$", "$\times$" and "$\otimes$" denote the scalar product, the vector product ant the tensor product of general vectors, respectively, whereas the operator ":" denotes the double contraction between tensors. All geometric variables denoted by $\mathring{(\cdot)}$ refer to the reference configuration. The compact (matrix) notation is used when convenient for the presentation of general equations, while detailed derivations are presented in index notation. Latin indices take on values $\{1, 2, 3\}$, while Greek indices take on values $\{1, 2\}$. Convective curvilinear coordinates θ^i are used, where θ^α are the surface coordinates of the shell's midsurface and θ^3 is the thickness coordinate. The letter $\mathbf{a}$ indicates a base vector on the shell's midsurface while $\mathbf{g}$ indicates a base vector in the shell continuum. Eventually, indices denote contravariant entities and pedices refer to covariant entities. Given a point $\mathbf{r}$ on the midsurface, the tangent covariant base vectors of the midsurface, and the normal base vector are obtained by:

$$\mathbf{a}_\alpha = \frac{\partial \mathbf{r}}{\partial \theta^\alpha}, \tag{3.1}$$

$$\mathbf{a}_3 = \frac{\mathbf{a}_1 \wedge \mathbf{a}_2}{|\mathbf{a}_1 \wedge \mathbf{a}_2|}. \tag{3.2}$$

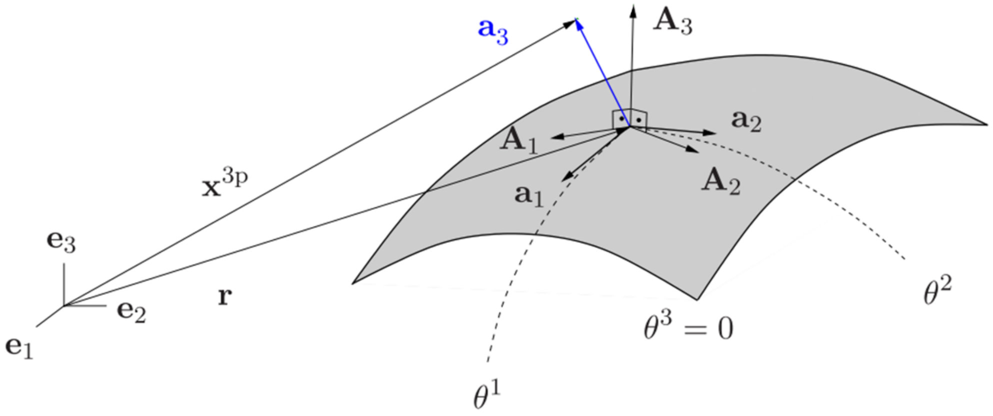


FIGURE 3.1: Geometry of a surface in the three-dimensional space. Figure adapted from [37]

A representation of the local and global frames of reference is provided in figure 3.1. The global Cartesian base vectors are denoted by the $\mathbf{e}_i$, the local Cartesian base vectors by $\mathbf{A}_i$, the local curvilinear base vectors by $\mathbf{a}_\alpha$. The metric coefficients of the midsurface are obtained by the first fundamental form of surfaces [36]:

$$A_{\alpha\beta} = \mathbf{a}_\alpha \cdot \mathbf{a}_\beta\,. \tag{3.3}$$

Curvature coefficients of the midsurface are obtained by the second fundamental form:

$$B_{\alpha\beta} = -\mathbf{a}_\alpha \cdot \frac{\partial \mathbf{a}_3}{\partial \theta^\beta}\,. \tag{3.4}$$

The position vector $\mathbf{x}$ in the shell continuum is defined as:

$$\mathbf{x} = \theta^\alpha \mathbf{a}_\alpha + \theta^3 \mathbf{a}_3\,, \tag{3.5}$$

whose derivative with respect to a curvilinear coordinate yields the base vectors $\mathbf{g}_\alpha$:

$$\mathbf{g}_\alpha = \frac{\partial \mathbf{x}}{\partial \theta^\alpha} = \mathbf{a}_\alpha + \theta^3 \frac{\partial \mathbf{a}_3}{\partial \theta^\alpha}\,, \tag{3.6}$$

$$\mathbf{g}_3 = \mathbf{a}_3\,. \tag{3.7}$$

The base vectors can be used to define the metric tensor $\mathbf{G}$, which can be expressed in terms of covariant and contravariant basis:

$$\mathbf{G} = G^{\alpha\beta}\,\mathbf{g}_\alpha \otimes \mathbf{g}_\beta = G_{\alpha\beta}\,\mathbf{g}^\alpha \otimes \mathbf{g}^\beta\,. \tag{3.8}$$

According to the definition (3.5), the metric coefficients in the shell continuum are defined by:

$$G_{\alpha\beta} = A_{\alpha\beta} - 2\theta^3 B_{\alpha\beta} + \left(\theta^3\right)^2 \frac{\partial \mathbf{a}_3}{\partial \theta^\alpha}\frac{\partial \mathbf{a}_3}{\partial \theta^\beta}\,. \tag{3.9}$$

For thin and moderately thick shells, the quadratic term is neglected [36]:

$$G_{\alpha\beta} = A_{\alpha\beta} - 2\theta^3 B_{\alpha\beta} \, . \tag{3.10}$$

3.2 Kinematic measures on surfaces

The deformation gradient **F** is defined by the base vectors in the reference and current configurations:

$$\mathbf{F} = \mathbf{g}_i \otimes \mathring{\mathbf{g}}^i \, , \tag{3.11}$$

$$\mathbf{F}^T = \mathring{\mathbf{g}}^i \otimes \mathbf{g}_i \, , \tag{3.12}$$

$$\mathbf{F}^{-1} = \mathring{\mathbf{g}}_i \otimes \mathbf{g}^i \, , \tag{3.13}$$

$$\mathbf{F}^{-T} = \mathbf{g}^i \otimes \mathring{\mathbf{g}}_i \, . \tag{3.14}$$

A suitable strain measure used for finite strain applications is given by the Green-Lagrange strain tensor [38]:

$$\mathbf{E} = \frac{1}{2}\left(\mathbf{F}^T\mathbf{F} - \mathbf{I}\right) . \tag{3.15}$$

Recalling that the identity tensor **I** is identical to the metric tensor in the reference configuration, it results:

$$\begin{aligned} \mathbf{E} &= \frac{1}{2}\left[\left(\mathring{\mathbf{g}}^i \otimes \mathbf{g}_i\right)\left(\mathbf{g}_j \otimes \mathring{\mathbf{g}}^j\right) - G_{ij}\left(\mathring{\mathbf{g}}^i \otimes \mathring{\mathbf{g}}^j\right)\right] \\ &= \frac{1}{2}\left(G_{ij} - \mathring{G}_{ij}\right)\left(\mathring{\mathbf{g}}^i \otimes \mathring{\mathbf{g}}^j\right) . \end{aligned} \tag{3.16}$$

In a Kirchhoff-Love shell only the in-plane coefficients of the strain tensor are considered:

$$E_{\alpha\beta} = \frac{1}{2}\left(G^{\alpha\beta} - \mathring{G}^{\alpha\beta}\right) . \tag{3.17}$$

Following the definition of metric coefficients in the shell continuum (3.10), the previous equation yields the split of the deformation tensor into membrane and bending (out-of-plane) effects:

$$E_{\alpha\beta} = \frac{1}{2}\left(A_{\alpha\beta} - \mathring{A}_{\alpha\beta}\right) + \theta^3\left(\mathring{B}_{\alpha\beta} - B_{\alpha\beta}\right) \tag{3.18}$$

It is worth noting that the strain coefficients are constant through the thickness.
The right Cauchy-Green deformation tensor is defined in the shell continuum instead, as it must be integrated over the thickness to provide the stress resultant:

$$\mathbf{C} = \mathbf{F}^T\mathbf{F} = G_{ij}\left(\mathring{\mathbf{g}}^i \otimes \mathring{\mathbf{g}}^j\right) . \tag{3.19}$$

According to Eq. (3.19), which is valid for a general 3D continuum, the covariant coefficients of the deformation tensor are identical to the metric coefficients of the deformed configuration, i.e., $C_{ij} = G_{ij}$. However, following the assumptions of the thin shell model, $C_{i3} = C_{3i} = 0$, while in the transverse normal direction $C_{33} = 1$. Accordingly, the covariant deformation tensor coefficients are:

$$C_{ij} = \begin{pmatrix} G_{11} & G_{12} & 0 \\ G_{21} & G_{22} & 0 \\ 0 & 0 & 1 \end{pmatrix}. \tag{3.20}$$

3.3 Constitutive equations

The relation between stress and strain is provided by the constitutive equation. Similarly to the strain tensor, there exist different definitions of stress tensor. The energetically conjugate quantity to the Green-Lagrange strain tensor **E** is the second Piola- Kirchhoff (PK2) stress tensor **S**. This can be formulated as the derivative of the strain energy function Ψ with respect to the tensors **E** or **C**, depending on the functional dependence of Ψ:

$$\mathbf{S} = \frac{\partial \Psi}{\partial \mathbf{E}} = 2\,\frac{\partial \Psi}{\partial \mathbf{C}}. \tag{3.21}$$

In linear elasticity stress and strain tensor are related by the fourth-order material tensor $\mathbb{C}$:

$$\mathbb{C} = \frac{\partial^2 \Psi}{\partial \mathbf{E} \partial \mathbf{E}} = 4\,\frac{\partial^2 \Psi}{\partial \mathbf{C} \partial \mathbf{C}}. \tag{3.22}$$

though the relation:

$$\mathbf{S} = \mathbb{C} : \mathbf{E}. \tag{3.23}$$

In the present work, we use both a St.Venant-Kirchhoff material model and a Hyperelastic material model. The constitutive law treatment, in the context of the shell model, is presented in the following sub-sections.

3.3.1 Linear material treatment

The St.Venant-Kirchhoff material model provides a linear relation between strains and stresses. The constitutive equation (3.23) have to be integrated in the material continuum. In the KL shell model, the stress distribution through the thickness is assumed to be linear, thus, the material law can be pre-integrated through the thickness analytically. The separation of the strain variable into membrane and bending contribution (3.39) is repeated for the stress variable, yielding the stress resultant **N** for normal forces and **M** for bending moments:

$$\mathbf{N} = \int_{-h/2}^{h/2} \mathbf{S}\, d\theta^3 = h\,\mathbb{C} : \varepsilon, \tag{3.24}$$

$$\mathbf{M} = \int_{-h/2}^{h/2} \mathbf{S}\,\theta^3\, d\theta^3 = \frac{h^3}{12}\,\mathbb{C} : \kappa. \tag{3.25}$$

Thus, the constitutive equations in the local Cartesian frame of reference, reads, in Voigt notation:

$$\begin{bmatrix} n^{11} \\ n^{22} \\ n^{12} \end{bmatrix} = h\,\mathbf{D} \begin{bmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ 2\varepsilon_{12} \end{bmatrix}, \quad \begin{bmatrix} m^{11} \\ m^{22} \\ m^{12} \end{bmatrix} = \frac{h^3}{12}\,\mathbf{D} \begin{bmatrix} \kappa_{11} \\ \kappa_{22} \\ 2\kappa_{12} \end{bmatrix}. \tag{3.26}$$

being $\mathbf{D}$ the material matrix, which for isotropic materials is written in terms of the Young's modulus E and the Poisson's ratio ν:

$$\mathbf{D} = \frac{E}{1-\nu}\begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix}. \tag{3.27}$$

3.3.2 Nonlinear material treatment

If large strains are considered, the transverse normal strain and the corresponding thickness stretch cannot be neglected. With the aim of keeping the present kinematic description of the shell, the transverse normal strain is statically condensed using the "plane stress" condition (which refers to the state of zero transverse normal stress). As a consequence, the shell kinematics can still be thoroughly described by the midsurface.

Under these assumptions, the additional unknown in the formulation is represented by the thickness deformation C_{33}. If $C_{33} = G_{33} = 1$ is used, as stated for linear materials, the plane stress conditions is violated, since:

$$S^{33} = 2\frac{\partial \Psi}{\partial C_{33}} \neq 0\,. \tag{3.28}$$

Accordingly, the transverse normal deformation C_{33} needs to be determined such that $S^{33} = 0$ is satisfied. Since no boundary conditions can be applied in the thickness direction, this can be done iteratively. An analytical method is presented instead in [28] for incompressible materials. Once the plane stress condition is enforced, it can be used to eliminate the transverse normal strain E_{33} by static condensation of the material tensor:

$$S^{33} = \mathbb{C}^{33\alpha\beta}E_{\alpha\beta} + \mathbb{C}^{3333}E_{33} = 0\,, \tag{3.29}$$

implying:

$$E_{33} = -\frac{\mathbb{C}^{33\alpha\beta}}{\mathbb{C}^{3333}}E_{\alpha\beta}\,. \tag{3.30}$$

The coefficients of the statically condensed material tensor are indicated by $\tilde{\mathbb{C}}^{\alpha\beta\gamma\delta}$ and are obtained as:

$$\tilde{\mathbb{C}}^{\alpha\beta\gamma\delta} = \mathbb{C}^{\alpha\beta\gamma\delta} - \frac{\mathbb{C}^{\alpha\beta 33}\mathbb{C}^{33\gamma\delta}}{\mathbb{C}^{3333}}\,. \tag{3.31}$$

The iterative procedure to enforce the plane stress condition has been presented in [8], and will be described in Section 9.5 in the context of the Active Strain approach. In the present work, only hyperelastic compressible materials are treated.

3.4 Weak formulation for Kirchhoff-Love shells

A total potential energy functional defined with respect to an equilibrium field $\mathbf{u}$, whose directional derivative yields the principle of virtual work, is, in the material configuration:

$$\Pi(\mathbf{u}) = \int_{\Omega_0} \left[\Psi(\mathbf{C}) + \rho\,\ddot{\mathbf{u}}\cdot\mathbf{u} + \mathbf{b}_0\cdot\mathbf{u}\right]\,dV + \int_{\partial\Omega_0} \mathbf{t}_0\cdot\mathbf{u}\,dS\,, \tag{3.32}$$

where Ω_0 represents the initial configuration, $\rho\,\ddot{\mathbf{u}}$ is the body inertial effect, and $\mathbf{b}_0$ and $\mathbf{t}_0$ are the vectors of body forces and boundary forces respectively. The strain energy density function Ψ is assumed to have a functional dependence on the right Cauchy-Green deformation tensor $\Psi = \Psi(\mathbf{C})$. In the following derivation the inertial effect is neglected for conciseness of notation. Under assumptions of external forces independent of the displacement fields, the stationary position of the above functional is obtained by equating to zero its derivative in an arbitrary variation $\delta\mathbf{u}$:

$$D_{\delta\mathbf{u}}\Pi(\mathbf{u}) = \int_{\Omega_0} \frac{\partial\Psi}{\partial\mathbf{C}} : D_{\delta\mathbf{u}}\mathbf{C}\, dV + \int_{\Omega_0} \mathbf{b}_0 \cdot \delta\mathbf{u}\, dV + \int_{\partial\Omega_0} \mathbf{t}_0 \cdot \delta\mathbf{u}\, dS\,, \tag{3.33}$$

$$D_{\delta\mathbf{u}}\Pi(\mathbf{u}) = \int_{\Omega_0} \mathbf{S} : D_{\delta\mathbf{u}}\mathbf{E}\, dV + \int_{\Omega_0} \mathbf{b}_0 \cdot \delta\mathbf{u}\, dV + \int_{\partial\Omega_0} \mathbf{t}_0 \cdot \delta\mathbf{u}\, dS\,, \tag{3.34}$$

This is a nonlinear system of equation which needs to be linearized in order to solve it. By linearizing in the direction of the arbitrary incremental displacement $\Delta\mathbf{u}$, the corresponding Newton equation reads:

$$D^2_{\delta\mathbf{u},\Delta\mathbf{u}}\Pi(\mathbf{u}) = -D_{\delta\mathbf{u}}\Pi(\mathbf{u})\,, \tag{3.35}$$

where the right hand side represents the residual vector in the corresponding discrete formulation. The second derivative of the potential functional only contains the derivative of the strain energy due to the assumption of displacement-independent external loads:

$$\begin{aligned} D_{\delta\mathbf{u},\Delta\mathbf{u}}\Pi(\mathbf{u}) &= \int_{\Omega_0} D_{\Delta\mathbf{u}}\left(\mathbf{S} : D_{\delta\mathbf{u}}\mathbf{E}\right)\, d\Omega_0 \\ &= \int_{\Omega_0} D_{\Delta\mathbf{u}}\mathbf{S} : D_{\delta\mathbf{u}}\mathbf{E} + \mathbf{S} : D^2_{\delta\mathbf{u},\Delta\mathbf{u}}\mathbf{E}\, d\Omega_0 \\ &= \int_{\Omega_0} D_{\Delta\mathbf{u}}\mathbf{E} : \mathbb{C} : D_{\Delta\mathbf{u}}\mathbf{E} + \mathbf{S} : D^2_{\delta\mathbf{u},\Delta\mathbf{u}}\mathbf{E}\, d\Omega_0\,, \end{aligned} \tag{3.36}$$

provided that the directional derivative of the Kirchhoff strain tensor is specified by applying the chain rule:

$$D_{\Delta\mathbf{u}}\mathbf{S}(\mathbf{E}) = \frac{\partial\mathbf{S}(\mathbf{E})}{\partial\mathbf{E}} : D_{\Delta\mathbf{u}}\mathbf{E} = \mathbb{C} : D_{\Delta\mathbf{u}}\mathbf{E}\,. \tag{3.37}$$

Exploiting the separation of membrane and bending action, it yields:

$$D_{\delta\mathbf{u}}\Pi_i = \int_{\Omega_0} \mathbf{S} : \delta\boldsymbol{\varepsilon} + \mathbf{S} : \theta^3\delta\boldsymbol{\kappa}\, d\Omega_0\,, \tag{3.38}$$

where:

$$\boldsymbol{\varepsilon} = \frac{1}{2}\left(A_{\alpha\beta} - \mathring{A}_{\alpha\beta}\right)\left(\mathring{\mathbf{a}}^{\alpha} \otimes \mathring{\mathbf{a}}^{\beta}\right)\,, \quad \boldsymbol{\kappa} = \left(\mathring{B}_{\alpha\beta} - B_{\alpha\beta}\right)\left(\mathring{\mathbf{a}}^{\alpha} \otimes \mathring{\mathbf{a}}^{\beta}\right)\,. \tag{3.39}$$

Provided that the strain variables are not dependent on the thickness (3.18), and considering the body and traction loads constant through the thickness, it yields:

$$D_{\delta\mathbf{u}}\Pi(\mathbf{u}) = \int_{A_0} \mathbf{N} : \delta\boldsymbol{\varepsilon} + \mathbf{M} : \delta\boldsymbol{\kappa}\, dA_0 + h\int_{A_0} \mathbf{b}_0 \cdot \delta\mathbf{u}\, dA_0 + h\int_{l_0} \mathbf{t}_0 \cdot \delta\mathbf{u}\, dl_0 \tag{3.40}$$

with:

$$\mathbf{N} = \int_{-h/2}^{h/2} \mathbf{S}\, d\theta^3\,, \tag{3.41}$$

$$\mathbf{M} = \int_{-h/2}^{h/2} \mathbf{S}\,\theta^3\, d\theta^3\,, \tag{3.42}$$

and **T** and **B** representing the vector of boundary and body forces respectively. A_0 and l_0 represent the area of the NURBS patch and the edge along which the traction load is applied, respectively. The same treatment is applied to the left hand side of Eq. (3.35):

$$\begin{aligned}
D^2_{\delta\mathbf{u},\Delta\mathbf{u}}\Pi(\mathbf{u}) &= \int_{A_0}\int_{-h/2}^{h/2} D_{\Delta\mathbf{u}}\left(\boldsymbol{\varepsilon} + \theta^3\boldsymbol{\kappa}\right) : \mathbb{C} : \delta\left(\boldsymbol{\varepsilon} + \theta^3\boldsymbol{\kappa}\right) + \mathbf{S} : D_{\Delta\mathbf{u}}\delta\left(\boldsymbol{\varepsilon} + \theta^3\boldsymbol{\kappa}\right)\, dh\, dA_0 \\
&= \int_{A_0}\int_{-h/2}^{h/2} D_{\Delta\mathbf{u}}\boldsymbol{\varepsilon} : \mathbb{C} : \delta\boldsymbol{\varepsilon} + D_{\Delta\mathbf{u}}\boldsymbol{\varepsilon} : \mathbb{C} : \theta^3\delta\boldsymbol{\kappa} + \theta^3 D_{\Delta\mathbf{u}}\boldsymbol{\kappa} : \mathbb{C} : \delta\boldsymbol{\varepsilon} + \\
&\left(\theta^3\right)^2 D_{\Delta\mathbf{u}}\boldsymbol{\kappa} : \mathbb{C} : \delta\boldsymbol{\kappa} + \mathbf{S} : D_{\Delta\mathbf{u}}\delta\boldsymbol{\varepsilon} + \mathbf{S} : \theta^3 D_{\Delta\mathbf{u}}\delta\boldsymbol{\kappa} \quad dh\, dA_0 \\
&= \int_{A_0} D_{\Delta\mathbf{u}}\boldsymbol{\varepsilon} : \mathbb{D}_1 : \delta\boldsymbol{\varepsilon} + D_{\Delta\mathbf{u}}\boldsymbol{\varepsilon} : \mathbb{D}_2 : \delta\boldsymbol{\kappa} + D_{\Delta\mathbf{u}}\boldsymbol{\kappa} : \mathbb{D}_2 : \delta\boldsymbol{\varepsilon} + D_{\Delta\mathbf{u}}\boldsymbol{\kappa} : \mathbb{D}_3 : \delta\boldsymbol{\kappa} + \\
&\mathbf{N} : D_{\Delta\mathbf{u}}\delta\boldsymbol{\varepsilon} + \mathbf{M} : D_{\Delta\mathbf{u}}\delta\boldsymbol{\kappa} \quad dA_0\,.
\end{aligned} \tag{3.43}$$

where:

$$\mathbb{D}_1 = \int_{-h/2}^{h/2} \mathbb{C}\, d\theta^3\,, \tag{3.44}$$

$$\mathbb{D}_2 = \int_{-h/2}^{h/2} \mathbb{C}\,\theta^3\, d\theta^3\,, \tag{3.45}$$

$$\mathbb{D}_3 = \int_{-h/2}^{h/2} \mathbb{C}\,\left(\theta^3\right)^2\, d\theta^3\,. \tag{3.46}$$

Prior to numerical approximation of the midsurface integrals (3.35), the stress resultants (3.41),(3.42) and the material matrices (3.44)-(3.46) must be integrated over the thickness. This can be performed analytically for linear materials, and numerically for hyperelastic materials, as shown in the section ahead.

3.5 Finite element approximation

The variations in Eq. (3.35) are considered as suitable trial functions fulfilling the topological requirement to guarantee the existence and uniqueness of the weak solution. According to the IGA approach, the space spanned by the NURBS basis functions is considered as the finite dimensional space of the approximate solution of the elastic problem. Thus, the displacement can be approximated as a linear combination of basis functions and discrete nodal values:

$$\mathbf{u} \approx \hat{\mathbf{u}} = \sum_{i}^{n_{sh}} N^i \mathbf{u}^i\,, \tag{3.47}$$

where N^a are the NURBS shape functions. n_{sh} is the total number of shape functions, and $\mathbf{u}^i$ is the nodal displacement vector. A detailed derivation of the discrete strain and curvature derivatives can be found in [9, 28]. After substitution of the discrete nodal displacement in Eq. (3.35), and application of the suitable test functions, it yields the Newton-Raphson equation:

$$\frac{\partial \mathbf{R}}{\partial \mathbf{u}_{n+1}} \Delta \mathbf{u}_{n+1} = -\mathbf{R}, \tag{3.48}$$

or, for the acceleration variation:

$$\frac{\partial \mathbf{R}}{\partial \ddot{\mathbf{u}}_{n+1}} \Delta \ddot{\mathbf{u}}_{n+1} = -\mathbf{R}, \tag{3.49}$$

where **R** represents the residual vector and its derivatives provide the tangent stiffness matrix (it accounts for the mass matrix too in time-dependent problems). In other words **R** represents the discrete form of the right hand side of Eq. (3.35). It is worth noting that the tangent stiffness matrix contains the second derivative of the displacement, as a consequence of the Kirchhoff-Love shell formulation. Hence, a C^1 parametric continuity is needed to fulfill the compatibility condition. In this scenario, where a minimum C^1 continuity can be easily achieved at the internal surface points, the NURBS-based IGA allows to exploit such continuity to implement the Kirchhoff-Love model without a specific treatment.

In the present implementation, strain and stresses are integrated over the surface by a classic Gauss-Legendre quadrature rule with $(p + 1) \times (q + 1)$ quadrature points within one element, being p and q the polynomial orders of the shape functions. However, more efficient quadrature rules, which account for the precise smoothness of the shape functions across the element boundaries, have been derived [39].

3.6 Time discretization

The equations of motion are integrated in time by the generalized-α scheme [40]. This algorithm allows to introduce controlled algorithmic damping in the unresolved high frequency modes. This is certainly a desirable feature when solving highly nonlinear FSI problems. The generalized–α scheme has had successful implementations in different FSI frameworks [41, 42, 43]. Hereafter, the scheme is presented in a *predictor–multicorrector* form [3].

Assuming a constant velocity predictor, the kinematic variables are initialized at the time t_{n+1} as:

$$\begin{aligned}
\dot{\mathbf{u}}^0_{n+1} &= \dot{\mathbf{u}}_n\,, \\
\ddot{\mathbf{u}}^0_{n+1} &= \frac{\gamma - 1}{\gamma}\,\ddot{\mathbf{u}}_n\,, \\
\mathbf{u}^0_{n+1} &= \mathbf{u}_n + \Delta t\,\dot{\mathbf{u}}_n + \frac{(\Delta t)^2}{2}\left[(1-2\beta)\,\ddot{\mathbf{u}}_n + 2\beta\,\ddot{\mathbf{u}}^0_{n+1}\right],
\end{aligned} \tag{3.50}$$

where the superscript indicates the iteration index, the subscript denotes the time step counter. Expressions (3.50) are obtained from the classic Newmark formulas [44], where β and γ are the Newmark coefficients. In the corrector stage, kinematic variables and external load are interpolated at a time instant in between two discrete time steps t_n and t_{n+1} by the interpolation coefficients α_f and α_m:

$$\begin{aligned}
\mathbf{u}_i &= \alpha_f \mathbf{u}_{n+1} + (1-\alpha_f)\mathbf{u}_n\,, \\
\dot{\mathbf{u}}_i &= \alpha_f \dot{\mathbf{u}}_{n+1} + (1-\alpha_f)\dot{\mathbf{u}}_n\,, \\
\ddot{\mathbf{u}}_i &= \alpha_m \ddot{\mathbf{u}}_{n+1} + (1-\alpha_m)\ddot{\mathbf{u}}_n\,, \\
\mathbf{F}^{ext}_i &= \alpha_f \mathbf{F}^{ext}_{n+1} + (1-\alpha_f)\mathbf{F}^{ext}_n\,.
\end{aligned} \tag{3.51}$$

In this section the subscript i denotes the time–interpolated quantities. $\mathbf{F}^{ext}_{n+1}$ is the load array assembled by integrating the stresses computed in the last fluid dynamic solution. A detailed discussion about the solver coupling in time is provided in Section 5.5. The intermediate solution is used to assemble the linearized problem, which is solved for the acceleration increment:

$$\frac{\partial \mathbf{R}_i}{\partial \ddot{\mathbf{u}}_{n+1}} \Delta \ddot{\mathbf{u}}_{n+1} = -\mathbf{R}_i \,. \tag{3.52}$$

After repeated application of the chain rule, this yields:

$$\left(\alpha_m \mathbf{M} + \alpha_f \beta (\Delta t)^2 \mathbf{K}(\mathbf{u}_i)\right) \Delta \ddot{\mathbf{u}}_{n+1} = -\mathbf{M}\, \ddot{\mathbf{u}}_i - \mathbf{F}^{int}_i + \mathbf{F}^{ext}_i \,. \tag{3.53}$$

The linear system stated in Eq. (3.53) is solved by a LU decomposition with partial pivoting. Given the FSI procedure presented in chapter 5, the proportion in the computational effort between the linear systems in (3.53) and (4.12) is biased towards the fluid side, even for the simplest FSI case. Then, the solution of Eq. (3.53) does not require more sophisticated solution algorithms. Once $\Delta \ddot{\mathbf{u}}_{n+1}$ is obtained, the variables for the next iteration loop are computed by a Newmark update:

$$\begin{aligned}
\ddot{\mathbf{u}}^{k+1}_{n+1} &= \ddot{\mathbf{u}}^{k}_{n+1} + \Delta \ddot{\mathbf{u}}_{n+1} \,, \\
\dot{\mathbf{u}}^{k+1}_{n+1} &= \dot{\mathbf{u}}_{n} + \Delta t \left((1-\gamma)\, \ddot{\mathbf{u}}_n + \gamma\, \ddot{\mathbf{u}}^{k+1}_{n+1}\right) , \\
\mathbf{u}^{k+1}_{n+1} &= \mathbf{u}_{n} + \Delta t \dot{\mathbf{u}}_n + \frac{1}{2}(\Delta t)^2 \left((1-2\beta)\, \ddot{\mathbf{u}}_n + 2\, \ddot{\mathbf{u}}^{k+1}_{n+1}\right) .
\end{aligned} \tag{3.54}$$

The corrector procedure, from Eq. (3.51) to Eq. (3.54), is repeated until convergence to a predefined tolerance on the L^2 norm of the residual vector [44]. The convergence (relative) tolerance is set to a value within the interval $\varepsilon \in \left[10^{-6}, 10^{-8}\right]$, depending on the numerical stiffness of the problem.

According to [40], the interpolation coefficients are parametrized by ρ_∞, the spectral radius of the iteration matrix, in order to achieve an optimal numerical damping. Following this, the Newmark parameters are build for second order accuracy and unconditional stability [3]:

$$\begin{aligned}
\alpha_m &= \frac{2-\rho_\infty}{1+\rho_\infty} \,, \\
\alpha_f &= \frac{1}{1+\rho_\infty} \,, \\
\beta &= \frac{1}{2}\left(1-\alpha_f+\alpha_m\right)^2 , \\
\gamma &= \frac{1}{2} - \alpha_f + \alpha_m \,.
\end{aligned} \tag{3.55}$$

Values within the interval $\rho_\infty \in [0.4, 0.6]$ are adopted in the present work. This allows to damp out the unresolved high frequencies introduced by the numerical discretization while preserving most of the natural ones.

4 An incompressible Navier-Stokes solver

Immersed Boundary (IB) methods [45] represent a flexible and computationally attractive tool for the simulation of flows around complex-shaped bodies on structured grids. Hence, a straightforward reduction in the computational cost of the FSI method can be obtained by employing efficient numerical schemes for the solution of the fluid phase, such as Fast Fourier transform algorithms, alternating direction implicit methods, and geometric multigrid methods. In this scenario, a numerical approach for the discretization of the Navier-Stokes equations is presented hereafter.

4.1 Governing Equations

A system of four scalar equations with parabolic and elliptic terms must be resolved. Their nondimensional and non-conservative form reads:

$$\begin{aligned} &\frac{\partial \mathbf{q}}{\partial t} + (\mathbf{q} \cdot \nabla)\,\mathbf{q} = -\nabla p + \frac{1}{\mathrm{Re}}\nabla^2 \mathbf{q} + \mathbf{f}\,, \\ &\nabla \cdot \mathbf{q} = 0\,, \end{aligned} \tag{4.1}$$

where the nondimensional variables $\mathbf{q}$, p, and $\mathbf{f}$ denote the flow velocity, the pressure, and the external volume force respectively. t indicates the time variable. The governing parameter is the Reynolds number $\mathrm{Re} = \rho_f Q L/\mu$, defined with respect to the scale parameter array $(\rho_f, Q, L, \mu)^T$, which contains fluid density, bulk velocity, characteristic length, and dynamic viscosity.

4.2 Discrete approach

The present finite-difference implementation follows a staggered approach [46], where scalar quantities are located at the cell center, and velocity components at the face center (see Fig. 4.1). The staggered arrangement, coupled with a projection method, makes differential operators as compact as possible and does not require *ad hoc* boundary conditions for pressure and pressure-like variables within the majority of time-schemes [12]. Furthermore, it can be proved that with a low-order approximation of spatial derivatives on a staggered mesh, and in the absence of time discretization errors and viscosity, global conservation of momentum, kinetic energy, and circulation are preserved. Eventually, the staggered arrangement, in conjunction with a direct forcing IB approach, is recognized to provide a smaller error on the interface condition with respect to collocated-grid schemes [47].

All derivatives are discretized in space with centered finite differences.

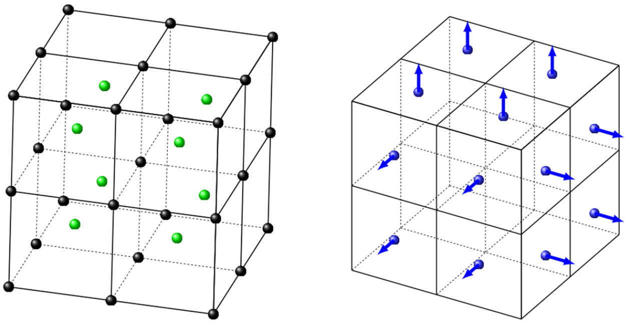


Figure 4.1: Staggered grids on which discrete grid functions are defined. Grid nodes (black dots) and cell-centered (green dots) locations are defined on the left figure. Face-centered grid functions (blue dots) are defined on the right. Figure adapted from [47]

4.2.1 Time discretization

The implicit Crank-Nicholson scheme is employed for the viscous terms, whereas the convective terms are discretized by the explicit three-steps Runge-Kutta scheme. This yields the discrete momentum equation for the k-th Runge-Kutta substep:

$$\frac{\tilde{\mathbf{q}} - \mathbf{q}^k}{\Delta t} = -\alpha^k \nabla p^k + \beta^k \left[(\mathbf{q}\cdot\nabla)\,\mathbf{q}\right]^k + \gamma^k \left[(\mathbf{q}\cdot\nabla)\,\mathbf{q}\right]^{k-1} + \frac{\alpha^k}{2Re}\nabla^2\left(\tilde{\mathbf{q}}+\mathbf{q}^k\right) + \tilde{\mathbf{f}}, \tag{4.2}$$

where $\tilde{\mathbf{f}}$ is the external body force term arising from the immersed boundary treatment. The time advancement coefficients for three substep levels [48] are $\alpha = [8/15,\ 2/15,\ 1/3]$, $\beta = [8/15,\ 5/12,\ 3/4]$, $\gamma = [0,\ -17/60,\ -5/12]$. This yields a self-starting scheme with second order accuracy in time. The third order Runge-Kutta method is chosen for the convective terms because it allows for a larger stability limit on the CFL. It has been proved with numerical investigations that the immersed boundary treatment does not affect the theoretical limit on the CFL number. This means that stable integration is possible in proximity of the limit prescribed for the Runge-Kutta scheme.

The presence of the pressure in this equation, defined at the cell center, requires the evaluation of the pressure gradient at the old time step, thus a pressure interpolation is needed to compute the right-hand-side (RHS) of equation (8) at the face centers. Second-order accurate interpolation operators are used to get mixed convective derivatives in points were velocities are not defined.

4.2.2 Helmholtz solver

Each substep of the time scheme is resolved by means of a classical fractional step scheme [49], with the pressure gradient introduced in the first stage [50]. Firstly, a non-solenoidal intermediate velocity field is computed using a blend of old and new variables, as stated in Eq. (4.2); then, a pressure-correction equation is solved to project the provisional field onto a solenoidal one [50, 10].

Eq. (4.2) must be solved for the provisional velocity $\tilde{\mathbf{q}}$. Since a semi-implicit scheme is implemented, the computation of the forcing field $\tilde{\mathbf{f}}$ would require the use of an implicit solution for the immersed boundary interpolation procedure, due to the dependence of $\tilde{\mathbf{f}}$ on $\tilde{\mathbf{q}}$. A straightforward way to address this issue with a minimal increase in the computational expense is to take a preliminary explicit step for the purpose of evaluating the forcing function, as proposed by Kim et al. [51]. Thus, a preliminary field $\hat{\mathbf{q}}$ can be computed as:

$$\hat{\mathbf{q}} = \mathbf{q}^k + \Delta t \left[-\alpha^k \nabla p^k + \beta^k \left[(\mathbf{q} \cdot \nabla) \mathbf{q} \right]^k + \gamma^k \left[(\mathbf{q} \cdot \nabla) \mathbf{q} \right]^{k-1} + \frac{\alpha^k}{Re} \nabla^2 \mathbf{q}^k \right] . \tag{4.3}$$

The evaluated field $\hat{\mathbf{q}}$ is used to compute the forcing function $\tilde{\mathbf{f}}$ by means of the interpolation technique described in Appendix A. Once the forcing function has been computed, Eq. (4.2) can be retrieved to get the provisional velocity field $\tilde{\mathbf{q}}$ in incremental form. This leads to the Helmholtz equation:

$$(1 - \xi \nabla^2) \Delta \tilde{\mathbf{q}} = \mathbf{S}^k + \Delta t \, \tilde{\mathbf{f}} , \tag{4.4}$$

with $\xi = (\alpha^k \Delta t)/(2Re)$, and

$$\Delta \tilde{\mathbf{q}} = \tilde{\mathbf{q}} - \mathbf{q}^k , \tag{4.5}$$

$$\mathbf{S}^k = \Delta t \left[-\alpha^k \nabla p^k + \beta^k \left[(\mathbf{q} \cdot \nabla) \mathbf{q} \right]^k + \gamma^k \left[(\mathbf{q} \cdot \nabla) \mathbf{q} \right]^{k-1} + \frac{\alpha^k}{Re} \nabla^2 \mathbf{q}^k \right] . \tag{4.6}$$

Note that the preliminary evaluation of $\hat{\mathbf{q}}$ does not require any additional computational step, since the terms in square brackets in Eq. (4.3) are then used as part of the right-hand-side in the Helmholtz equation. It should be pointed out that modeling a no-slip condition with an external force field within a pressure projection scheme introduces an error of the same order of magnitude of the standard scheme truncation error [52].

Given the implicit treatment of the viscous components, the computation of $\tilde{\mathbf{q}}$ requires the inversion of large sparse matrices. An approximate factorization method [53] is employed to reduce the large band of the coefficient matrices to a set of tridiagonal matrices, while preserving the scheme accuracy in time. The use of iterative solvers is avoided in the present work, because in a partitioned FSI framework the convergence procedure would make the global time-advancement loop inefficient. Furthermore, in presence of highly stretched and refined grids, matrix preconditioning or multigrid methods would be mandatory to increase the convergence rate. Although very efficient preconditioning techniques have been developed for similar purposes [54], these approaches are avoided to limit the overall complexity of the method.

The approximate factorization [53] for discrete parabolic and hyperbolic problems belongs to the class of Alternating Direction Implicit (ADI) methods. The spatial operator is split in three one-dimensional components, each corresponding to an implicit problem over a single coordinate. As a result of the splitting a set of tridiagonal system must be solved sequentially. The discrete Laplace operator corresponding to a five-points centered finite differences is denoted with a simplified notation as:

$$\begin{aligned} \nabla^2 q \approx \; & \frac{q_{i-1,j,k} - 2q_{i,j,k} + q_{i+1,j,k}}{\Delta x^2} + \frac{q_{i,j-1,k} - 2q_{i,j,k} + q_{i,j+1,k}}{\Delta y^2} + \\ & \frac{q_{i,j,k-1} - 2q_{i,j,k} + q_{i,j,k+1}}{\Delta z^2} = \delta_x^2 q + \delta_y^2 q + \delta_z^2 q \end{aligned} \tag{4.7}$$

If this notation is applied to equation (4.4), and the right-hand-side is indicated for simplicity by $RHS_{i,j,k}$, one can write in index notation:

$$\left[(1 - \beta \left(\delta_x^2 + \delta_y^2 + \delta_z^2 \right) \right] \Delta q_{i,j,k} = RHS_{i,j,k} \tag{4.8}$$

The Laplace operator is split as:

$$\left(1-\beta\delta_x^2\right)\left(1-\beta\delta_y^2\right)\left(1-\beta\delta_z^2\right)\Delta q_{i,j,k} = RHS_{i,j,k} \tag{4.9}$$

Thus the linear system is solved in three sequential steps:

$$\begin{aligned}
&\text{STEP 1:} \qquad \left(1-\beta\delta_x^2\right)\Delta q^{**}_{i,j,k} = RHS_{i,j,k}\\
&\text{STEP 2:} \qquad \left(1-\beta\delta_y^2\right)\Delta q^{*}_{i,j,k} = \Delta q^{**}_{i,j,k}\\
&\text{STEP 3:} \qquad \left(1-\beta\delta_z^2\right)\Delta q_{i,j,k} = \Delta q^{*}_{i,j,k}
\end{aligned} \tag{4.10}$$

In step 1 a tridiagonal system must be resolved for each grid node $j,\ k$ of the plane i, because only the finite difference in direction x is implicit. The procedure is repeated in the other directions exploiting the intermediate quantities $\Delta q^{*}_{i,j,k}$ and $\Delta q^{**}_{i,j,k}$. Tridiagonal systems are inverted by means of the Thomas algorithm with Sherman-Morrison perturbation if the solution direction has periodic boundary conditions [55].

The order in which calculations for each one-coordinate problem (4.10) are performed is indifferent for three-periodic problems. In the present framework, where wall-bounded flows are subject of study, the inhomogeneos direction is resolved last [12] since explicit boundary conditions must be imposed just after the provisional field calculation.

Unlike original ADI methods, the approximate factorization leads to an unconditionally stable scheme which preserves the order of accuracy in time even in three-dimensional problems, since a factorization error $O(\Delta t^3)$ is introduced [56].

4.2.3 Pressure correction

It can be proved that the intermediate velocity $\tilde{\mathbf{q}}$ is globally divergence-free, but not locally [12]. Thus, a scalar variable ϕ is introduced to project the provisional field onto a divergence-free field. This operation allows to evaluate the velocity at the time step $n+1$ via the formula [50]:

$$\mathbf{q}^{n+1} = \tilde{\mathbf{q}} - \alpha^k\,\Delta t\,\nabla\phi\,. \tag{4.11}$$

In order to get the scalar ϕ, the discrete divergence operator is applied to Eq. (4.11), then the left-hand-side is set to zero for the local mass conservation. By enforcing the continuity condition, the following Poisson's equation for the scalar ϕ is obtained:

$$\nabla^2\phi = \frac{\nabla\cdot\tilde{\mathbf{q}}}{\alpha\,\Delta t}\,. \tag{4.12}$$

The velocity field at the time-step $n+1$ can finally be computed from Eq. (4.11), whereas the new pressure field is updated by [50]:

$$p^{n+1} = p^n + \phi - \xi\,\nabla^2\phi\,. \tag{4.13}$$

The solution of the elliptic equation (4.12) would require a large computational effort, similarly to Eq. (4.4). In the simulation of unsteady flows the pressure-correction equation must be solved to a tight tolerance to ensure mass conservation at each time step. Hence, no iterative methods are employed at this stage. The large band of the coefficient matrix is reduced here relying on one-dimensional Fast Fourier Transform (FFT) for equally-spaced grid directions. Being X the uniform grid direction, the pressure correction equation is reduced in the Fourier space to:

$$\left(-\omega_{x,i}^2 + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}\right)\mathcal{F}(\phi) = \mathcal{F}\left(\frac{\nabla \cdot \tilde{\mathbf{q}}}{\alpha^k \, \Delta t}\right), \tag{4.14}$$

where $\mathcal{F}(\cdot)$ denotes the one-dimensional FFT operator, and $\omega_{x,i}^2$ is the m-th modified wavenumber in the X direction. Thus, n_x equations must be solved for each modified wave number. The uniform grid spacing in direction X has been exploited to reformulate the three-dimensional Poisson problem in a more effective way; a 1D-FFT in direction X followed by n_x bi-dimensional Poisson problem. The FFT with periodic boundary conditions requires approximately $n_x \log n_x$ operations to be computed, therefore the computational saving is relevant with regards to direct or iterative solvers on uniform grids.

Afterwards, the solution of the Poisson problem in the Y-Z plane is accelerated by means of the eigendecomposition of the discrete operator matrices. This technique is well suited to problems in which a repeated solution of the Poisson's equation is required [57], as in time-dependent problems over staggered grids, where just the forcing terms change at each time step. The two-dimensional Poisson problem, arising from (4.14) can be rewritten in matrix notation as:

$$\mathbf{D}_y \tilde{\phi}_l + \tilde{\phi}_l \mathbf{D}_z^T = \tilde{\mathbf{S}}_l, \tag{4.15}$$

where $\tilde{\phi}$ is the unknown matrix of dimension $N_y \times N_z$ in the Fourier space, $\mathbf{D}_y$ and $\mathbf{D}_z$ are the penta-diagonal matrices arising from the centered difference operators of dimensions $N_y \times N_y$ and $N_z \times N_z$, respectively; $\tilde{\mathbf{S}}_l$ represents the transformed right–hand side of dimension $N_y \times N_z$. By exploiting the symmetry and the orthogonality of $\mathbf{D}_z$, one can write:

$$\mathbf{D}_z^T = \mathbf{Q}_z \mathbf{\Lambda}_z \mathbf{Q}_z^T, \tag{4.16}$$

where $\mathbf{Q}_z$ represents a real matrix whose column are the eigenvectors associated with $\mathbf{D}_z$, and $\mathbf{\Lambda}_z$ a diagonal matrix containing the corresponding eigenvalues. Following this, one can get the two-dimensional problem:

$$\mathbf{D}_y \phi_l^* + \phi_l^* \mathbf{\Lambda}_z = \mathbf{S}_l^*. \tag{4.17}$$

Therefore, the decomposition of the finite difference operator in direction Z leads to a computationally simplified problem which can be solved as N_z independent problems in direction Y by projecting the unknown matrix on a diagonal operator (4.17). Thus each plane Poisson equation (corresponding to the *m*-th wavenumber) is reduced to a set of tridiagonal problems in the form:

$$\left[\mathbf{D}_y + \left(\lambda_m - k_{1l}^{\prime 2}\right)\mathbf{I}_y\right]\phi_{lm}^* = \mathbf{S}_{lm}^*. \tag{4.18}$$

After equation (4.18) has been solved for each point k, and ϕ_i has been assembled, it is possible to recover the unknown scalar field by $\tilde{\phi} = \phi^* Q_z^T$.

The algorithm for solving the Poisson equation provides for the *a priori* calculation of eigenvector and eigenvalues of the finite difference coefficient matrix in z direction. Then, it proceeds with the following time-dependent operations:

- Perform a real-to-complex Fourier transform in the x direction;
- For each wavenumber compute $\phi^* = \tilde{\phi} Q_z$ and $S^* = \tilde{S} Q_z$;
- For each wavenumber solve $D_y \phi^* + \phi^* \Lambda_z = S^*$;

- For each wavenumber compute $\tilde{\phi} = \phi^* Q_z^T$;
- Perform the complex-to-real inverse Fourier transform in the x direction.

This found various applications tailored for specific boundary conditions or domain configurations [58][59]. In the present case the problem can be reformulated applying the eigendecomposition to the operator D_y as well, as illustrated in [58]; however, the procedure involving the resolution of N_z tridiagonal matrices of order N_y was considered to better perform on domain elongated in z direction, which are often employed herein.

4.3 Boundary conditions treatment

The resolution of the momentum equation by means of the increment:

$$\Delta \mathbf{q}^n = \hat{\mathbf{q}} - \mathbf{q}^n \,, \tag{4.19}$$

yields a provisional velocity field with second order accuracy in time [60]. Thus, the physical boundary conditions for $\mathbf{u}^{n+1}$ can be equivalently assigned to $\hat{\mathbf{u}}$:

$$\hat{\mathbf{q}} = \mathbf{q}^{n+1} + O\left(\Delta t^2\right) . \tag{4.20}$$

As well as natural boundary conditions, the IB forcing term is included at this stage. However, it should be pointed out that the forcing condition imposed at the internal faces is then spoiled within the correction step in order to enforce continuity at local level. A numerically exact enforcement of mass conservation and internal interface conditions cannot be achieved at the same time with the present method, without setting *ad-hoc* boundary conditions on the Poisson equation (4.12).

In the present work the *outflow* boundary condition is reproduced by means of a radiative outlet [61]. A two-dimensional wave equation is solved at the outlet face to allow the propagating structures to exit the computational domain with minimal distortions:

$$\frac{\partial \mathbf{q}}{\partial t} + c\frac{\partial \mathbf{q}}{\partial z} = \mathbf{0} \,, \tag{4.21}$$

where z is the outflow direction. The convective velocity c is *a priori* assumed to be within the range $[0.8Q, Q]$. Numerical and experimental investigation found that it is an acceptable assumption for the simulation of flows across bluff bodies if a suitable downstream domain extent is considered [62][63].

5 An immersed boundary method for moving boundaries

5.1 Role of computational FSI

Fluid–Structure Interaction is a class of problems with mutual dependence between the fluid and solid phases. The fluid-dynamics depends on the configuration of the structure and its kinematics, and the deformation of the structure depends on the hydrodynamic loads acting on the structure. FSI occurs at every biological length scale, as well as in many industrial applications. The first category includes multiscale sophisticated mechanisms such as the projection of lamellipodia and bleb-like protrusions by motile cells, the flow of blood in the heart and throughout the circulation, the locomotion of swimming fishes and flying birds and insects, and the dispersal of seeds and pollen by the wind. Among the more relevant FSI phenomena in industrial applications one can mention the fluttering of lifting wings, the deflection of wind-turbine blades, the inflation of automobile airbags, the dynamics of spacecraft parachutes, the rocking motion of ships.
In engineering applications, FSI plays an important role and influences the decisions that go into the design of systems of contemporary interest. On the other hand, the FSI mechanism can provide insight on complex biological mechanisms. Therefore, truly predictive FSI methods, which help address the problems of interest, are in high demand in industry, research laboratories, medical fields, space exploration, and many other contexts. In the present chapter the FSI framework including the techniques for the information transfer between the segregated solvers described in chapters 3 and 4, is thoroughly described.

5.2 Immersed boundary methods for fluid-structure interaction

In IB methods, the procedures to transfer variables from the fluid domain to the structural domain and vice versa are necessary to satisfy the boundary conditions on the interface because the fluid mesh does not conform to the body interface. For an elastic body, the computational domain is divided into elements. The position, velocity and acceleration of each element are determined from the balance among the inertia force of the element, surface integration of the Cauchy traction vector along the interface of the element, and internal force within the element. A schematic of the field variables involved in the information transfer is provided in Fig. 5.1. The information transfer at the interface constitutes a key feature of every FSI method. Boundary-fitted approaches have the advantage of satisfying kinematic constraints by construction, but, when large displacements are induced on the immersed body, these approaches require the re-meshing of the fluid domain and the field variables interpolation in time. Sophisticated algorithms must be employed to achieve a well–conditioned grid at each time–step with limited computational effort [65, 66]. However, certain problems might lead to severe mesh distortions with ensuing degradation of the spatial accuracy of the fluid solution. The most widespread approaches belonging to the class of deforming domain methods are the Arbitrary Lagrangian–Eulerian (ALE) formulation, and the

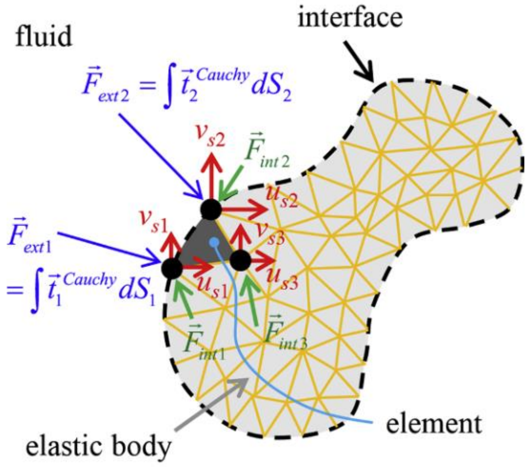


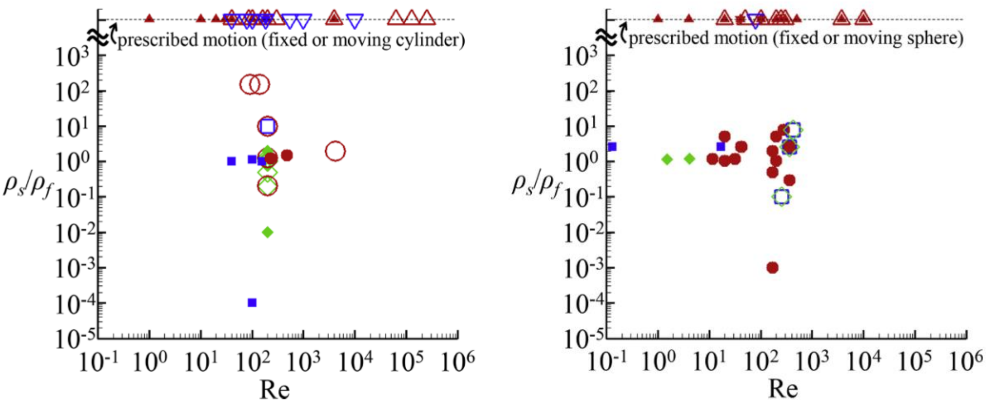
FIGURE 5.1: schematic of the field variables involved in the information transfer between structural and fluid solvers in a general FSI framework. Figure taken from [64]

Deforming–Spatial–Domain/Stabilized Space–Time (DSD/SST) formulation. A comprehensive overview of these methods can be found in [41].

In the original IB method by Peskin [67], the structure is assumed to consist of massless Lagrangian nodes and springs connecting them. The velocity on a Lagrangian node is imposed by interpolating the velocities at surrounding Eulerian grids through a Discrete Delta Function (DDF), and the momentum forcing at the node is transferred to nearby Eulerian grid points through DDF. Such momentum forcing at the node is obtained from the balance of the spring force and momentum forcing. Peskin [68] further developed an IB method to handle the mass of an elastic body. However, it is not possible to apply this IB treatment to FSI of a rigid body, since it represents a limit condition for the lagrangian node stiffness. Kim and Peskin [69] applied an IB method based on a penalty approach to FSI of a rigid body by introducing massless and massive boundaries, in which the momentum forcing at a Lagrangian node on the massless boundary is obtained through a feedback loop using the difference between the positions of massless and massive boundaries. However, this IB method severely restricts the size of computational time step owing to large values of parameters required for the feedback loop. Borazjani et al. [13] and Luo et al. [70] applied IB methods based on velocity reconstruction to FSI of rigid and elastic bodies, respectively, where the no-slip condition is satisfied by transferring the velocity of structure at an interface to fluid mesh, and the Cauchy's stress theorem is satisfied by transferring the velocity gradient and pressure of fluid at an interface to structure. However, spurious force oscillations occur when Eulerian grids are changed from a fluid state to a solid state or vice versa due to the movement of an interface [71]. To this extent, Uhlmann [72] developed an IB method to reduce spurious force oscillations and avoid excessive restrictions on the computational time step size. The momentum forcing is obtained at a Lagrangian node through the Navier-Stokes equations using the interpolated fluid and structure velocities and it is distributed to nearby Eulerian nodes through a transfer function. One of the most widespread transfer functions is that computed by a Moving-Least-Square interpolation [73]. Direct momentum forcing through a feedback approach induces small spurious force oscillations for FSI but smears the interface, which leads to failure of resolving the boundary layer and satisfying the no-slip condition. To reduce the error on the no-slip condition, a few strategies have been suggested. A multidirect forcing technique was applied to enforce the fluid velocity updated from the governing equations of fluid flow equal to the structure velocity at a Lagrangian node through an iterative imposition of additional momentum forcing [74]. Alternatively Kempe et al. [75] imposed additional momentum forcing by repeating the forcing step in a semi-implicit fashion.

A better enforcement of the interface condition is provided by the class of ”sharp-interface” methods [76, 77, 14, 13]. These methods modify the finite-difference stencil near the immersed boundary to ensure an accurate representation of the fluid boundary layer. Typically they consider the immersed boundary to be a boundary of the flow domain and solve the equations of fluid dynamics on only one side of the interface. Although they present severe spurious force oscillations with moving boundaries, efficient solutions to damp out these oscillations have been found [78].

Two fundamental issues regarding FSI with IB methods are the problems associated with low density ratio of solid to fluid phase ρ_s/ρ_f and high Reynolds number. Fig. 5.2 depicts the spectrum of ρ_s/ρ_f and *Re* under which different authors performed simulations of the flows around a circular cylinder and a sphere, with different IB methods. Their effort has been focused on reducing the density ratio up to value lower than unity and increasing the Reynolds number above 10^4.



(a) (b)

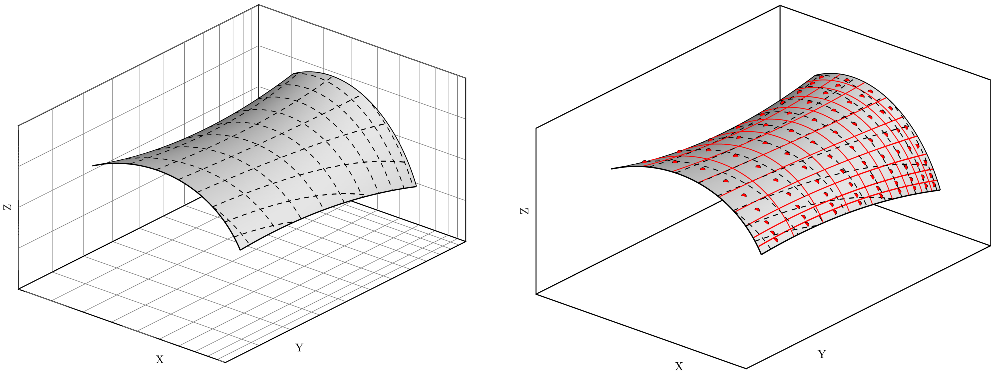
FIGURE 5.2: Flows around a circular cylinder (a) and sphere (b) at various Reynolds numbers and density ratios, obtained with with different IB methods: direct momentum forcing, velocity reconstruction, monolithic and segregated approaches. Figure taken from [64].

A low density ratio, often observed in biological FSI applications, was found to be source of numerical instability when an explicit FSI coupling (staggered) scheme is employed [18]. To achieve numerical stability, several solution have been experimented: the governing equations of fluid and solid phases have been solved in a semi-implicit fashion for time advancement, by means of iterative or predictor-corrector schemes [13, 64]. Alternative solutions are presented in [75], where an indirect way to calculate fluid force on a sphere is proposed, and in [79], where a virtual mass approach to the linear momentum balance is proposed.

At high Reynolds number, it is important to resolve a thin boundary layer to capture the flow transition from laminar to turbulent. So far, most of the simulations using IB methods have been confined to relatively low Reynolds number because of the limited resolution at the boundary layer. Among the mentioned methods, the direct-forcing sharp interface method provided the most accurate reconstruction of the boundary layer, therefore it is expected to provide the most accurate results at high *Re*. Local refinement has been also used in Cartesian mesh to resolve a thin boundary layer. A few examples are provided by the flow around a sphere at $Re = 10,000$ [80], flow around a rotating wind turbine at $Re = 17,000$ based on the turbine diameter [81] and flow around an Eppler 387 airfoil at $Re = 30,000$ based on the chord length [82]. On the other hand, IB methods with velocity reconstruction can save the number of grids near the wall by imposing a velocity profile based on a wall model rather than a linear velocity profile [83].

5.3 Embedded boundary reconstruction

Following the direct forcing scheme proposed by Uhlmann [72], the information is transferred among arbitrary Lagrangian and Eulerian positions. The no-slip condition at the body interface is enforced by applying a consistent body force on a cluster of Lagrangian markers laying over the immersed interface. This approach was proposed as a solution to the spurious force oscillations observed in earlier direct–forcing methods for moving–boundary problems [84, 85]. This matches the computational requirements of FSI applications with deformable bodies, where the load approximation plays a crucial role for the correct resolution of the FSI dynamics [71, 75, 86]. Since the position of the Lagrangian markers and Eulerian nodes almost never coincides, the fluid solution at each marker must be reconstructed by an interpolation technique. A Moving-Least-Squares (MLS) approach was originally proposed by Vanella and Balaras [73], with the aim of building a transfer function able to provide a smooth pressure distribution when applied to moving boundaries. The MLS interpolation allows for a satisfactory compromise between consistent sharp-like distribution of the field variables over the surface and smoothing of the local force discontinuities when crossing the Eulerian cells [80, 87]. Recently the MLS-Lagrangian forcing has been applied to FSI problems involving deformable shell-like structure [16, 88]. On the heels of these works, in the present implementation the MLS procedure is adjusted for the information transfer between a Cartesian grid and a C^1-continuous NURBS surface. The introduction of a discrete forcing function in the time-stepping scheme was already described in Section 4.1.



(a) (b)

FIGURE 5.3: (a) Projection of the Eulerian grid lines (dashed lines) over the NURBS surface. (b) Lagrangian markers (red dots) and consistent surface tessellation (red lines).

The fundamental steps of the forcing procedure are summarized as follows:

(a) An Eulerian grid density distribution is retrieved by detecting the local grid size at the shell interface. A three-dimensional Cartesian grid must be projected over a surface. Hence, the grid spacing at each Cartesian direction is acquired, then the two smallest cell spacing distribution are selected. The discrete distribution is interpolated by a cubic spline law; then, a Cartesian grid of Lagrangian markers is collocated in the NURBS parametric space accordingly to the interpolated distribution. Before collocation, the marker distribution is scaled by a predefined value $\Delta l/\Delta x$, which is needed to avoid a "porous" representation of the surface in terms of body force condition. A sketch of the marker collocation over the surface with respect to the local Eulerian grid spacing is provided in Fig. 5.3.

(b) The Eulerian cell containing each Lagrangian marker is detected. A support domain of n_e cells is created by collecting the nearest neighbor cells, as depicted by a two-dimensional projection in Fig. 5.4. In the three-dimensional space $n_e = 27$. The fluid solution $\hat{\mathbf{q}}$ is then retrieved in the faces belonging to the support domain. Thus, the velocity component $\hat{q}_i^l$, at the Lagrangian point location $\mathbf{X}$, is computed by means of the interpolating surface constructed with the MLS operator $\mathbf{\Phi}$:

$$\hat{q}_i^l(\mathbf{X}) = \mathbf{\Phi}^T(\mathbf{X})\,\hat{\mathbf{q}}_i = \sum_{k=1}^{n_e} \Phi_k^l(\mathbf{X})\,\hat{q}_i^k\,, \tag{5.1}$$

where $\mathbf{\Phi}^T(\mathbf{X})$ is the MLS transfer function built at run time for the velocity component i, and $\hat{\mathbf{q}}_i = [\hat{q}_i^1\;\hat{q}_i^2\;...\;\hat{q}_i^e]^T$ is the array of collected velocity components in the support domain. The index k enumerates the cells contained in the support domain. A detailed description of the transfer operator construction is provided in A.

(c) A volume force component F_i is computed at each Lagrangian point via:

$$F_i = \frac{\dot{u}_i - \hat{q}_i^l}{\Delta t}\,, \tag{5.2}$$

being $\dot{u}_i$ the shell velocity component at $\mathbf{X}$. F_i^l is the force that one needs to impose on the flow field to get the desired flow velocity at the interface.

(d) At this stage, the computed body force must be spread back over the fluid faces within the support domain. Namely, a Lagrangian body force F_i^l is transformed into a smooth distribution of Eulerian loads f_i^l by means of the same transfer operator defined in Eq. (5.1). Hence, the corresponding volume force at the Eulerian point k is evaluated as:

$$f_i^k(\mathbf{x}^k) = \sum_{l=1}^{n_l} c^l\,\Phi_k^l(\mathbf{x}^k)\,F_i^l\,, \tag{5.3}$$

where n_l denotes the number of Lagrangian points associated with the Eulerian node k, i.e., Lagrangian points whose support domain contains the selected *k-th* Eulerian point, and c^l is the scaling factor needed to preserve the local conservation of momentum in the forcing operation. The forces arising from Eq. (5.3) are collected into a three-dimensional forcing field and included in Eq. (4.4) to finally enforce the desired boundary condition. The scaling coefficient c^l is obtained by imposing the conservation of the total force acting on the fluid within the support domain:

$$\sum_{k=1}^{n_{et}} f_i^k(\mathbf{x}^k)\,\Delta V^k = \sum_{l=1}^{n_{lt}} F_i^l \mathbf{x}^l\,\Delta V^l\,, \tag{5.4}$$

being n_{et} the total number of forced grid points, and n_{lt} the total number of Lagrangian markers. ΔV^k and ΔV^l are the forcing volumes associated with the Eulerian and Lagrangian points involved in the information transfer. By convention, the former corresponds to the Eulerian cell volume and the latter is computed as the product of the local patch area A_l associated to the selected Lagrangian marker, and the average local cell size h_l:

$$\Delta V^l = A^l\,h^l\,, \tag{5.5}$$

$$h^l = \frac{1}{3}\sum_{k=1}^{n_e} \Phi_k^l\,(\Delta x^k + \Delta y^k + \Delta z^k)\,. \tag{5.6}$$

The area A^l associated to each marker is obtained by tessellation of the NURBS surface just after the collocation of the markers themselves. The tessellation of the surface is performed considering the integral conservation of the surface area. By substituting expression (5.3) in (5.4), one obtains the scaling coefficient:

$$c^l = \frac{\Delta V^l}{\sum_{k=1}^{n_e} \Phi_k^l\,\Delta V^k}\,. \tag{5.7}$$

In this scenario, the torque is conserved if the cell volume across the stencil is kept constant for each marker. This is trivially satisfied in case of uniform grids, however, in case of moderate stretching the departure from equivalence is limited to a small fraction of the local torque [73].

The process described above is repeated at each time-step for each Lagrangian marker in an explicit fashion. This results in the superposition of the support domain stencils for adjacent Lagrangian markers; this effect actually corrupts the local conservation of the momentum achieved by (5.7). The explicit treatment of the MLS interpolation allows for a reduced computational cost (see A), but it leads to a local error on the momentum conservation which mitigates the smoothing properties of the approach. This problem might be solved by assembling the MLS operator implicitly [87], or including a further continuity constraint in the transfer function [89]. Nevertheless, both techniques imply a larger computational expense at run-time.

A reduction of the relative Lagrangian spacing $\Delta l/\Delta x$ is found to reduce the error on the enforced condition and reduce the permeability of the surface up to a certain threshold, beyond which a further reduction of $\Delta l/\Delta x$ does not affect the interface resolution. Such a threshold is investigated in Section 6.4.3.

An important advantage of the present implementation arises from the flexibility of the marker collocation procedure. The isoparametric paradigm allows for a simple collocation of the Lagrangian markers through parametric coordinates, and for a full decoupling of the relative node density in the shell and fluid solvers. It should be however highlighted that the isoparametric mapping has been successfully applied also to spring-network solvers to meet this achievement [82]. Secondly, a flexible marker collocation is suitable for a body moving across a stretched

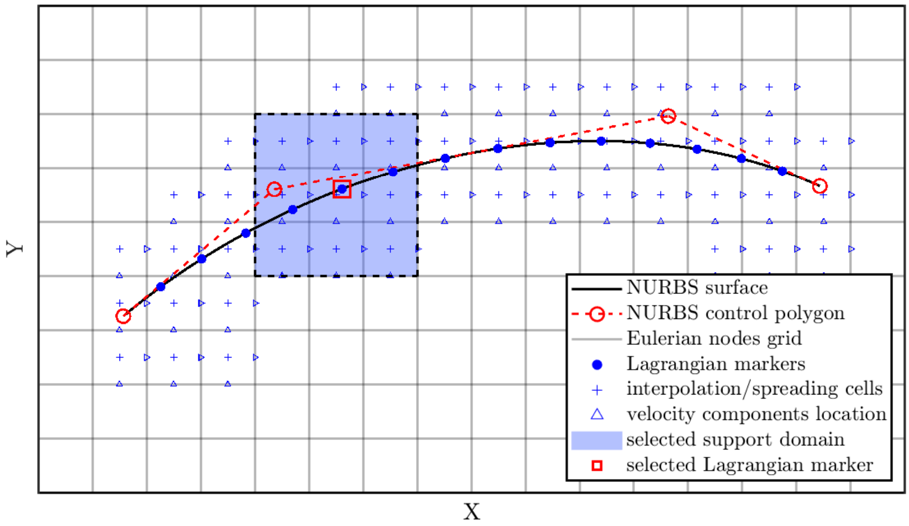


FIGURE 5.4: Schematic of the IBM forcing with MLS reconstruction. The cell centers of all Eulerian cells involved in the forcing procedure are marked; the support domain corresponding to the selected Lagrangian marker is highlighted by a blue stencil.

Eulerian grid. The collocation can be repeated every time step without affecting the shell discretization, with minimal computational effort; this is a remarkable feature for bodies moving within a near-wall region, where a significant Eulerian node stretching is needed for the resolution of the boundary layer gradients [90].

5.4 Hydrodynamic load evaluation

For non-boundary conforming approaches the computation of the local hydrodynamic force on the immersed surface is not a trivial task. Following the procedure introduced by [16, 73], the load distribution is obtained exploiting the MLS approach described in the previous subsection. Given the present immersed boundary treatment, the flow field across the surface presents a smooth transition layer whose thickness takes at most two Eulerian cells, as shown by [16] with numerical experiments. Therefore, the field variables are corrupted by the reconstruction procedure at the surface location. This feature was found to induce an underestimation of the actual hydrodynamic forces [73]. One way to address this issue is the interpolation of the field variables at a probe created along the normal direction at a distance h_p from the surface. For an open surface, pressure and velocity derivatives are evaluated at the outward ($\mathbf{n}^+$) and inward ($\mathbf{n}^-$) probe locations, in correspondence of the shell quadrature points. Thus, at each Gaussian point, two probes and two support domains are created (see Fig. 5.5), then the MLS procedure is repeated twice. The probe distance h_p is computed from (5.6) as the local average grid size. The pressure interpolated at the probe is transported on the surface p^{gp} with a simplified boundary layer equation:

$$p^{gp} = p^p + h_p \frac{D\mathbf{q}^{gp}}{Dt} \cdot \mathbf{n}^{gp} , \tag{5.8}$$

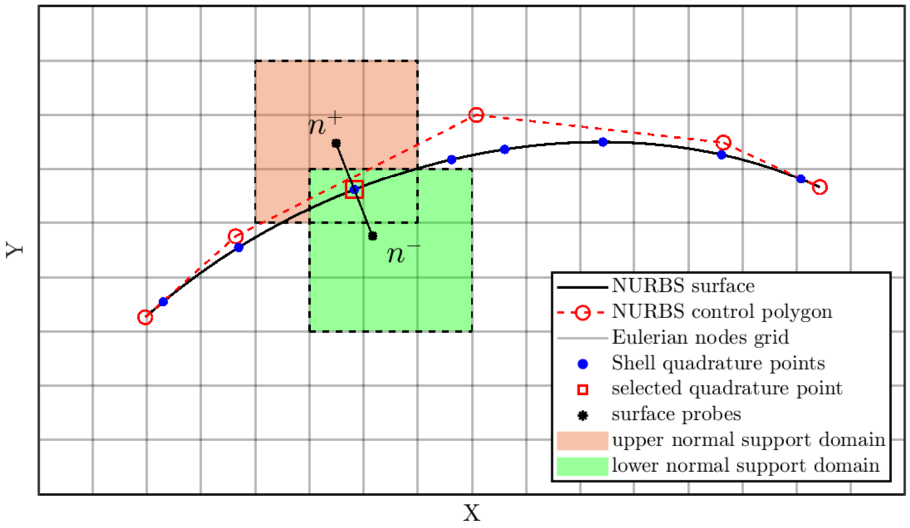


FIGURE 5.5: Schematic of the field variable interpolation for the calculation of the hydrodynamic load over the surface.

being p^p and $\mathbf{q}^p$ the pressure and fluid velocity on the probe, $\mathbf{n}^{gp}$ the outward/inward unit normal vector, and $D\mathbf{q}^{gp}/Dt$ the material derivative of the fluid velocity at the quadrature point location [73], which is approximated by the corresponding Gaussian point acceleration. The velocity derivatives at each Gaussian point $\partial q_i/\partial x_j$, needed for the evaluation of the viscous stresses, are computed by differentiating Eq. (5.1):

$$\left(\frac{\partial q_i}{\partial x_j}\right)^{gp} = \sum_{k=1}^{n_e} \left(\frac{\partial \Phi_k}{\partial x_j}\,\hat{q}_i^k\right)^p . \tag{5.9}$$

The solution of Eq. (5.9) requires an additional computational effort for the evaluation of the derivative of the transfer function. Given the fact that h_p is comparable with the local grid size, and assuming a linear variation of the velocity near the body, the derivatives at the probe location, computed from (5.9) are a good approximation for the corresponding derivatives at the surface [73]. From a purely physical perspective, this corresponds to the assumption that the probe relies inside the linear region of the boundary layer of the flow. This is a reasonable approximation in case of adequately refined grid in proximity of the body. The local stress at the surface is assembled as:

$$\mathbf{t}^{gp} = -(p^{gp+} - p^{gp-})\mathbf{I}\cdot\mathbf{n}^{gp+} + 2\mu(\mathbf{E}^+ - \mathbf{E}^-)\cdot\mathbf{n}^{gp+}\,, \tag{5.10}$$

$$\mathbf{E} = \frac{1}{2}\left(\nabla\mathbf{q}^{gp} + (\nabla\mathbf{q}^{gp})^T\right), \tag{5.11}$$

and integrated over the surface with a Gaussian integration rule. A wide campaign of numerical test (Section 6.4) proves that the force evaluation procedure does provide a sufficiently smooth load distribution for moving boundaries without underestimating the local load peaks. This approach was shown to give accurate results, comparable with those of sharp direct-forcing approaches, as well as manage high pressure differences across the surface [16].

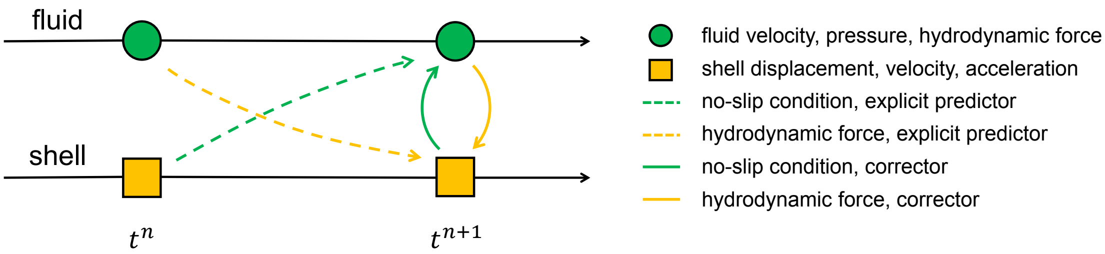


FIGURE 5.6: Schematic of the iterative FSI time coupling.

5.5 Partitioned solvers for time coupling

The present FSI implementation follows a partitioned approach, where fluid and structural solutions are staggered in time. However, a staggered procedure does not reflect the instantaneous mutual interaction between the flow field and the shell. The fluid method needs the actual node displacement and velocity to enforce the no-slip condition at the interface, but the structural method needs the hydrodynamic loads to integrate the equations of motion. It has been found in previous studies that a time–staggered solution degrades the temporal accuracy of the FSI procedure as long as the the phase-density ratio ρ_s/ρ_f reduces [91]. The lower the shell relative density, the more sensitive the shell dynamics to local fluid accelerations. Consequently, for highly scattered and sharp acceleration fields, the structural solver undertakes a stiff system of equations, and possibly loses convergence. In other words, for $\rho_s/\rho_f \sim O(1)$ the fluid acts as an extra mass on the structural degrees of freedom at the coupling interface [18]. Thus, such *added mass* effect restricts the numerical stability of the staggered (weak) coupling [17][92]. Moreover, the added mass destabilizing effect is found in partitioned solvers regardless of the spatial discretization scheme, but it is rather an inherent property of sequentially staggered methods [18]. As shown in section 6.4.3, this feature was recognized in the present implementation. The temporal accuracy in the FSI dynamics could be restored with an iterative transfer of coupling variables at the body interface. Nevertheless, the iterative coupling dramatically increases the computational expense per time step. In this case many works have been focused on the reduction of the computational effort for the strong coupling, as well as the enlargement of the applicability limits of weak coupling schemes [93, 94, 43]. A general iterative procedure is proposed in the present work.

In the following, both weak and strong coupling procedures are summarized. Circled letters stand for the fundamental operators: fluid dynamic solver $\mathcal{F}$, hydrodynamic load $\mathcal{L}$, shell motion solver $\mathcal{S}$. Square and curly brackets contain input (initial solution) and output variables for each operator, respectively; the round brackets contain the information needed for time integration. Subscripts contain the time advancement indices, whereas superscripts denote the iteration indices for the strong coupling procedures.

Staggered (weak) coupling

a. $[\mathbf{q}_n,\, p_n]\,(\mathbf{u}_n,\, \dot{\mathbf{u}}_n) \Rightarrow \mathcal{F} \Rightarrow \{\mathbf{q}_{n+1},\, p_{n+1}\},$

b. $[\mathbf{q}_{n+1},\, p_{n+1}]\,(\mathbf{u}_n,\, \ddot{\mathbf{u}}_n) \Rightarrow \mathcal{L} \Rightarrow \{\mathbf{F}^{FSI}_{n+1}\},$

c. $[\mathbf{F}^{FSI}_{n+1}]\,(\mathbf{u}_n,\, \dot{\mathbf{u}}_n,\, \ddot{\mathbf{u}}_n) \Rightarrow \mathcal{S} \Rightarrow \{\mathbf{u}_{n+1},\, \dot{\mathbf{u}}_{n+1},\, \ddot{\mathbf{u}}_{n+1}\}.$

The iterative coupling is presented as a predictor-multicorrector scheme, where convergence is checked on the Root Mean Square Error (RMSE) of the structural displacement of each degree of freedom. A schematic of the iterative coupling is provided in Fig. 5.6 for sake of clarity. The predictor stage corresponds to the staggered coupling described above, with the superscript p denoting the predicted solution variable. The convergence tolerance was set to $\epsilon = 10^{-8}$. Within this framework a low order time stepping of the FSI operators could result in a slow convergence rate. Thus, the strong coupling procedure is combined with the Aitken adaptive under-relaxation technique [95][96]; this was shown to effectively accelerate the convergence procedure in iterative FSI methods [13]. In the following procedural scheme the symbol ← indicates a variable assignment.

Iterative (strong) coupling
Predictor stage

a. $[\mathbf{q}_{n+1}, p_{n+1}]\,(\mathbf{u}_n, \dot{\mathbf{u}}_n) \Rightarrow \mathcal{F} \Rightarrow \{\mathbf{q}^p_{n+1}, p^p_{n+1}\}$,

b. $[\mathbf{q}^p_{n+1},\, p^p_{n+1}]\,(\mathbf{u}_n,\, \ddot{\mathbf{u}}_n) \Rightarrow \mathcal{L} \Rightarrow \{\mathbf{F}^p_{n+1}\}$,

c. $[\mathbf{F}^p_{n+1}]\,(\mathbf{u}_n,\, \dot{\mathbf{u}}_n,\, \ddot{\mathbf{u}}_n) \Rightarrow \mathcal{S} \Rightarrow \{\mathbf{u}^p_{n+1},\, \dot{\mathbf{u}}^p_{n+1},\, \ddot{\mathbf{u}}^p_{n+1}\}$,

d. $\dot{\mathbf{u}}^p_{n+1} \leftarrow \dot{\mathbf{u}}^k_{n+1},\;\; \dot{\mathbf{u}}^p_{n+1} \leftarrow \dot{\mathbf{u}}^k_{n+1},\;\; \ddot{\mathbf{u}}^p_{n+1} \leftarrow \ddot{\mathbf{u}}^k_{n+1},\;\; \mathbf{F}^p_{n+1} \leftarrow \mathbf{F}^k_{n+1},\;\; \mathbf{q}^p_{n+1} \leftarrow \mathbf{q}^k_{n+1},\;\; p^p_{n+1} \leftarrow p^k_{n+1}$.

Corrector stage

e. $[\mathbf{q}^k_{n+1},\, p^k_{n+1}]\,(\mathbf{u}^k_{n+1},\, \dot{\mathbf{u}}^k_{n+1}) \Rightarrow \mathcal{F} \Rightarrow \{\mathbf{q}^{k+1}_{n+1},\, p^{k+1}_{n+1}\}$,

f. $[\mathbf{q}^{k+1}_{n+1},\, p^{k+1}_{n+1}]\,(\mathbf{u}^k_{n+1},\, \ddot{\mathbf{u}}^k_{n+1}) \Rightarrow \mathcal{L} \Rightarrow \{\mathbf{F}^{k+1}_{n+1}\}$,

g. $[\mathbf{F}^{k+1}_{n+1}]\,(\mathbf{u}^k_{n+1},\, \dot{\mathbf{u}}^k_{n+1},\, \ddot{\mathbf{u}}^k_{n+1}) \Rightarrow \mathcal{S} \Rightarrow \{\tilde{\mathbf{u}}^{k+1}_{n+1},\, \tilde{\dot{\mathbf{u}}}^{k+1}_{n+1},\, \tilde{\ddot{\mathbf{u}}}^{k+1}_{n+1}\}$,

h. Compute the under-relaxation coefficient and correct the computed solution $\tilde{\mathbf{u}}^{k+1}_{n+1}$,

i. Check for convergence on the shell displacement: RMSE $(\tilde{\mathbf{u}}^{k+1}_{n+1} - \mathbf{u}^k_{n+1}) < \epsilon Q$,

j. if not converged, increment k by one and return to step *e.*,

k. $\dot{\mathbf{u}}^k_{n+1} \leftarrow \dot{\mathbf{u}}^{k+1}_{n+1},\, \dot{\mathbf{u}}^k_{n+1} \leftarrow \dot{\mathbf{u}}^{k+1}_{n+1},\, \ddot{\mathbf{u}}^k_{n+1} \leftarrow \ddot{\mathbf{u}}^{k+1}_{n+1},\, \mathbf{F}^k_{n+1} \leftarrow \mathbf{F}^{k+1}_{n+1},\, \mathbf{q}^k_{n+1} \leftarrow \mathbf{q}^{k+1}_{n+1},\, p^k_{n+1} \leftarrow p^{k+1}_{n+1}$.

l. if converged, update solution variables in time: $\dot{\mathbf{u}}_{n+1} \leftarrow \dot{\mathbf{u}}^{k+1}_{n+1},\, \dot{\mathbf{u}}_{n+1} \leftarrow \dot{\mathbf{u}}^{k+1}_{n+1},\, \ddot{\mathbf{u}}_{n+1} \leftarrow \ddot{\mathbf{u}}^{k+1}_{n+1},\, \mathbf{F}_{n+1} \leftarrow \mathbf{F}^{k+1}_{n+1},\, \mathbf{q}_{n+1} \leftarrow \mathbf{q}^{k+1}_{n+1},\, p_{n+1} \leftarrow p^{k+1}_{n+1}$.

The under-relaxation is applied to the acceleration array as:

$$\ddot{\mathbf{u}}^{k+1}_{n+1} = \ddot{\mathbf{u}}^k_{n+1} + \lambda^{k+1}\left(\tilde{\ddot{\mathbf{u}}}^{k+1}_{n+1} - \ddot{\mathbf{u}}^k_{n+1}\right), \tag{5.12}$$

being $\tilde{\ddot{\mathbf{u}}}^{k+1}_{n+1}$ the solution returned by the structural solver, before the under-relaxation. Then, velocity and displacement are retrieved by the Newmark update formulas (3.54). The under-relaxation factor is dynamically determined from the Aitken method [95] by exploiting information from the last two FSI iterations:

$$\lambda^{k+1} = -\lambda^k \frac{\left(\tilde{\ddot{\mathbf{u}}}^k_{n+1} - \ddot{\mathbf{u}}^{k-1}_{n+1}\right)^T \left\{\left(\tilde{\ddot{\mathbf{u}}}^k_{n+1} - \ddot{\mathbf{u}}^k_{n+1}\right) - \left(\tilde{\ddot{\mathbf{u}}}^k_{n+1} - \ddot{\mathbf{u}}^{k-1}_{n+1}\right)\right\}}{\left\|\left(\tilde{\ddot{\mathbf{u}}}^k_{n+1} - \ddot{\mathbf{u}}^k_{n+1}\right) - \left(\tilde{\ddot{\mathbf{u}}}^k_{n+1} - \ddot{\mathbf{u}}^{k-1}_{n+1}\right)\right\|^2} \tag{5.13}$$

For the first iteration a predefined guess value for λ is set to 0.5; in the proceeding time-steps the initial value is taken from the last FSI iteration $\lambda^1_{n+1} = \lambda^m_n$, where m is the last FSI iteration step at the previous time index n.

6 Fluid-Structure Interaction: benchmarking and numerical experiments

6.1 Verification and validation

The reliability of multiphysic models as predictive tools for engineering scenarios is an issue of fundamental concern in computational science, especially when dealing with multi-degree-of-freedom models. In the present work the level of confidence assigned to a computer prediction of a complex event relies on the collection of procedures described in [97] named ”verification and validation”. The verification process addresses the quality of the numerical treatment of the model used in the prediction and the validation process addresses the quality of the model itself. The verification of the numerical solution, which concerns the accuracy and the order of convergence of the numerical implementation, is provided for the simplest test-cases considered, and the relative results are judged by means of tolerances supplied by the vast majority of the analysts in the corresponding field. The validation process addresses the analysis of meaningful physical indicators or the comparison against reproducible and widely accepted experimental measurements.

In the present chapter, the implementation of the unplugged solvers is firstly verified, by simple examples. Subsequently, the reliability of the coupled FSI solver is investigated against different test-cases of engineering and biologic inspiration, involving fundamentally different physical and numerical conditions.

In the numerical experiments, a combination of the following boundary conditions are applied on the boundary faces of the fluid domain:

- Periodic: Dirichlet conditions are applied to achieve domain periodicity;
- Wall: Dirichlet conditions are employed to get zero normal flux at the domain face;
- Inflow: a uniform streamwise velocity Q_∞ and zero cross-stream velocity are specified. Furthermore the normal pressure gradient is set to zero;
- Outflow: a convective boundary condition is applied by resolving the Eq. (4.21) for minimal distortions and minimal reflections.

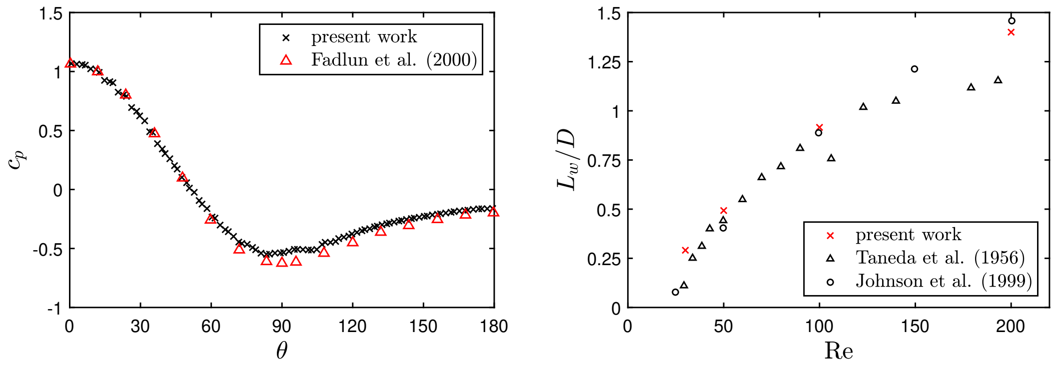
Similarly, the shell boundary conditions are defined as follows:

- Hinged edge: Dirichlet conditions are specified at the boundary by setting the displacement to zero for the specified degrees of freedom;
- Clamped edge: displacements and normal rotations of the clamped edge are prescribed by fixing the first two rows of control points.

Simulations are carried out with a constant time-step size. It has been proven with numerical investigations that the present immersed boundary implementation does not affect the theoretical limit on the CFL number of 0.6 [60]. This means that stable integration is possible in proximity of the limit prescribed for the Runge-Kutta scheme. However, this limit is impractical for FSI simulations involving mutual interaction between segregated solvers; a much smaller time integration step is required to capture the actual FSI dynamics. Hence, simulations are carried out within the CFL range [0.05 – 0.2].

6.2 Verification of the immersed boundary method

Most of the IBM analysts introduced their verification studies with the investigation of the flow past an isolated sphere. Such a test relies on extensive numerical and experimental results providing local and integral measurements of the quantities of interest. Moreover, the topology of the sphere presents all possible intersections with a Cartesian grid, thus the approximation of the flow field in the near-wall region can be investigated to assess the accuracy of the results. We investigate the flow past a sphere at different values of the Reynolds number Re = {30, 50, 100, 200}, in which the flow field is steady and axisymmetric. A computational domain of extension $[-5D, 5D] \times [-5D, 5D] \times [-4D, 16D]$ is considered, being Z the streamwise direction. We use a non-uniform Cartesian grid with cubic spline coarsening far from the body-wake region. We adopt a minimum grid spacing of $\Delta x = 0.02D$ in the body region and a time step size of $\Delta t = 0.005\, U/D$. By using an Eulerian-Lagrangian refinement ratio of 0.6, as discussed in section 5.3, we end up with 22326 Lagrangian markers. Our implementation of the immersed boundary method, along with the fluid solver, is compared with the experimental results by Taneda et al. [98], with the data obtained by Johnson et al. [99] and Fornberg at al. [100] with a body-fitted method, and with the results obtained by Fadlun with a direct forcing IB method [76]. The comparison is based on several local and integral factors: (i) the distribution of the pressure coefficients over the surface (Fig. 6.1), defined as $c_p = 2(p - p_\infty)/\rho U^2$, where p is the pressure at the sphere wall and p_∞ is pressure in the undisturbed flow field, (ii) the length of the separation bubble L_w/D, intended as the axial distance between the sphere surface and the central stagnation point in the sphere wake (Fig. 6.1), (iii) the drag coefficient of the sphere



(a) (b)

FIGURE 6.1: Steady flow past a sphere. Pressure coefficient distribution over the sphere surface at $Re = 100$ (a). Length of the separation bubble ad different Reynolds numbers (b).

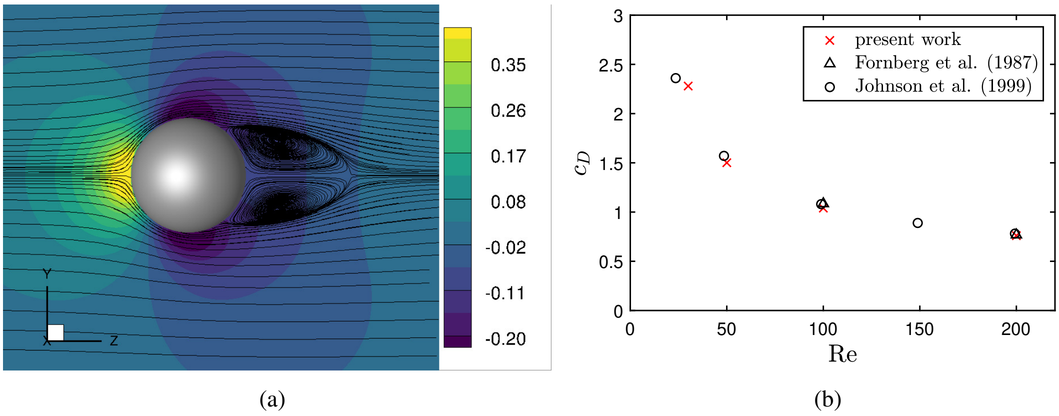
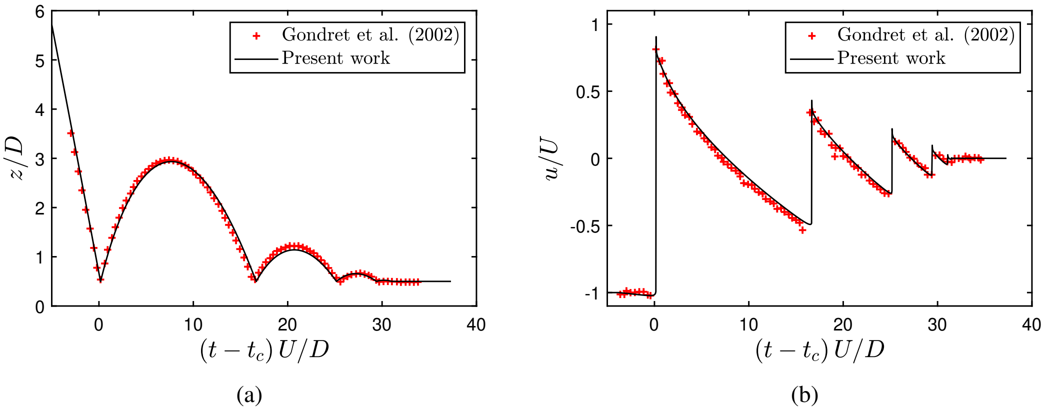


FIGURE 6.2: Steady flow past a sphere. Visualization of the flow streamlines superposed to the pressure contours over a section of the flow field at Re = 100 (a). Evolution of the drag coefficient with the Reynolds number (b).

$c_D = 8\,F_z/\rho U^2 \pi D^2$ (Fig. 6.2). A satisfactory agreement is found between the present calculations and the reference results, showing good accuracy both in the flow field and in the local forces over the surface. Specifically, we obtain a value of drag coefficient for the $Re = 100$ case equal to $c_D = 1.041$, which is slightly underestimated with respect to the reference values, $c_D = 1.085$ for Fornberg [100] and $c_D = 1.079$ for Fadlun [76].

In second instance we test the effectiveness of the solver in handling moving bodies. A falling sphere in a quiescent fluid is simulated. This test cases is performed in a domain of extension $[-3.5D, 3.5D] \times [-2D, 8D] \times [-3.5D, 3.5D]$ with a periodic boundary condition in the directions X, Z, and viscous wall conditions in the Y direction. The rigid body motion is resolved by integrating the Newton equations of motion with an Adams-Bashforth explicit scheme. The gravity force act along the Y direction. A further scope of this test is the assessment of the interaction between the boundary condition of the fluid domain and the immersed boundary treatment. To this extent we implement the collision/lubrication model presented in [101] for the simulation of 2-ways coupled particle leaden flows. Such a model, which is not presented here for brevity, provides a penetration penalty factor to estimate the normal collision force, thus allowing a minimal superposition of the immersed boundary with the fluid domain edge. We employ a uniform Cartesian grid with spacing $\Delta x = 0.035D$, and a time step size of $\Delta t = 0.002D/U$. We run the simulation from the release of the sphere up to the restoration of the rest condition after four rebounds. The bouncing dynamics makes the body undergo very large accelerations in the impact, forcing the IB treatment to deal with a temporal discontinuity in the forcing function. The sphere is released at $y/D = 8$ with zero initial velocity.

The kinematics of the sphere is compared with the experimental results of Gondret et al. [102] in terms of Y-trajectory and velocity (Fig. 6.3). Specifically, we simulate a case with Stroual number $St = 152$, relative particle density $\rho = 7.8$, Reynolds number Re = 163, and normal restitution coefficient $e_n = 0.97$. Our simulations match the experimental measurements, confirming the effectiveness of the IBM method with moving bodies, in the limit case of extreme accelerations and superposition with the fluid domain boundary. The well-known near-wall primary vortex dynamics [73, 103] is observed in the fluid motion after the impact (Fig. 6.4). [!h]



(a)

(b)

FIGURE 6.3: Trajectory (a) and velocity evolution in time (b) of the center of mass of a bouncing sphere in a quiescent fluid. The variable t_c defines the instant of the sphere-wall contact.

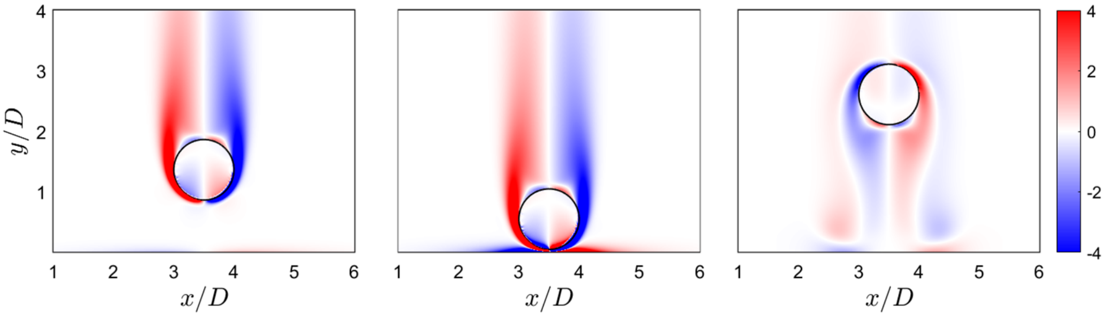
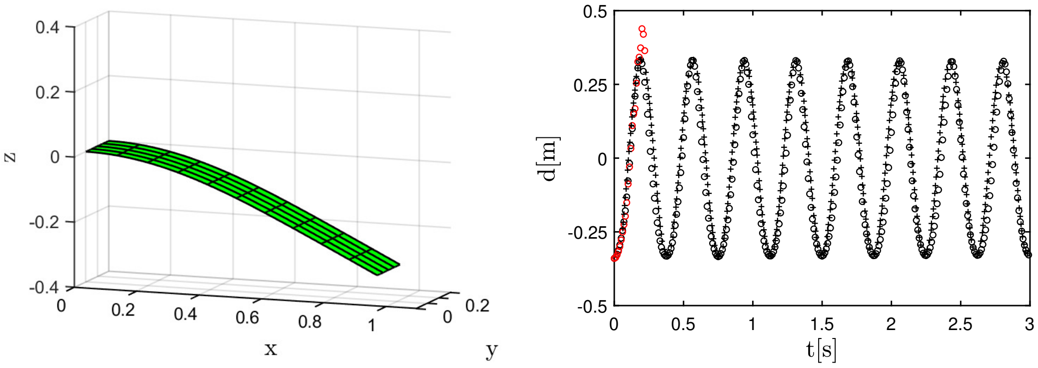


FIGURE 6.4: Snapshots of the sphere-wall interaction with a restitution coefficient $e_n = 0.97$ and modeled lubrication forces. The spurious fluid motions inside the sphere are intentionally illustrated.

6.3 Verification of the shell solver

The shell solver is embedded in the FSI framework by re-implementing the software developed by Kiendl. et al [8], which has been extensively tested in previous works [28, 104, 31]. Hence, only a simple verification case is presented in this thesis. A freely vibrating, undamped beam is considered. The beam dimensions are $1 \times 0.1 \times 0.001$ m (length × width × thickness). The material parameters are $E = 210.1 \times 10^{10}\, Pa$, $\nu = 0.3$, $\rho = 7850\, kg/m^3$. The plate is clamped at the left edge and initially deformed corresponding to a static load of $\mathbf{F} = 225\, \hat{\mathbf{z}}\, N/m$ at the right edge. At time $t = 0$, the load is removed so that the beam is subject to free vibrations. The validity of the solution is verified comparing the present simulation with the results presented in [104], in terms of tip displacement (see Fig. 6.5). The present simulation, obtained with 10 cubic elements in the bending direction and 4 linear elements in the other direction, matches the reference solution in terms of amplitude and frequency. A time step-size of $\Delta t = 0.01\, s$ is employed. In Fig. 6.5 the solution obtained with a Newmark time-stepping rule is presented to illustrate the necessity of controlled numerical damping. The Newmark method ($\gamma = 1/2$, $\beta = 1/4$), which is a conservative scheme, presents non-physical oscillations arising due to the unresolved high frequency modes. The spurious oscillations bring the solver to a divergent solution within the first period.



(a) (b)

FIGURE 6.5: Freely vibrating beam in the configuration corresponding to the largest displacements (a). Time traces of the tip z-displacement (b): present solution obtained with a Generalized-α integration (black circles), present solution obtained with a Newmark integration (red circles) reference solution (black crosses).

On the contrary, the Generalized-α method provides a regular kinematics. The numerical dissipation introduced with the Generalized-α scheme [40] is essential in the FSI framework, where large displacements and a highly scattered load distribution are experienced by the structure.

6.4 Numerical tests for Fluid-Structure Interaction

The aim of this Section is the validation of the proposed mathematical approach for the simulation of fluid-structure interaction problems. A number of significant test have been selected among a variety of literature cases to get a meaningful sample of different physical and numerical conditions.

6.4.1 Hinged flag

The first test consists of a widespread benchmark case for FSI schemes involving large displacements and a fully three-dimensional dynamics. A flat plate invested by a uniform flow is simulated, as first investigated by Huang et al. [105] with the aim of replicating a flag-in-the-wind problem. The numerical challenges associated with the rigid body motion modes and the complex blending of inertial and elastic effects, still make this case a demanding test for any FSI procedure. In fact its dynamics was broadly investigated [105, 106] under different flag material properties and inflow conditions, as well as it was proposed in several works as validation problem [107, 16]. The flag is modeled as a square plate hinged at the leading edge. The leading edge is aligned with the Z-axis, with its midpoint coinciding with the origin of the Eulerian coordinate system. The flag is initially held at an angle $\alpha = 0.1\pi$ from the ZX plane, being Z the streamwise direction. In the present work, a computational domain of extent $[-1.5L, 1.5L] \times [-5L, 5L] \times [-2L, 8L]$ is employed. The boundary conditions of the fluid problems are: periodic in the X direction, inviscid wall in the Y direction and inflow/radiative outflow in the Z direction (Fig. 6.6). The Eulerian grid is uniform with spacing $\Delta = 0.02L$, while the flag is discretized by 14 third order elements in both parametric directions. A constant time step

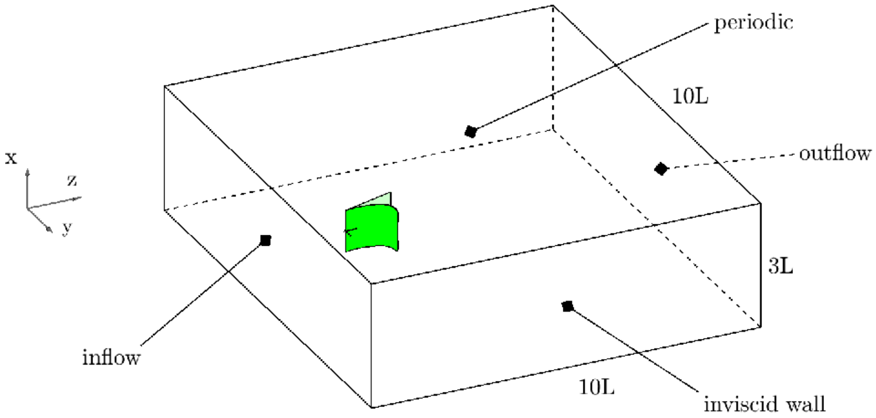


FIGURE 6.6: Schematic of the computational domain and boundary conditions for the FSI simulation of a hinged flag subject to uniform free stream.

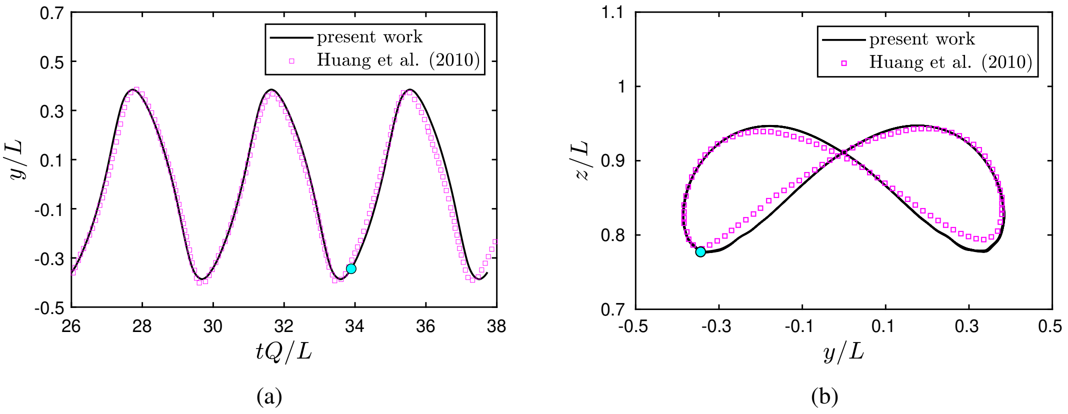
size of $\Delta t Q/L = 0.001887$ was considered in order to get a bulk $CFL = Q_\infty \Delta t/\Delta x = 0.1$. The simulation is carried out at a Reynolds number $Re = 200$.

The shell model presented by Huang et al. [105] was derived from a definition of elastic energy based on a decoupling of bending and membrane strains, therefore the Poisson's ratio is chosen to be $\nu = 0$. The nondimensional bending stiffness and the mass ratio are assumed to be:

$$\gamma = \frac{Eh^3}{12\,(1-\nu^2)\,\rho_f Q^2 L^3} = 0.001\,, \quad \mu = \frac{\rho_s h}{\rho_f L} = 1.0\,. \tag{6.1}$$

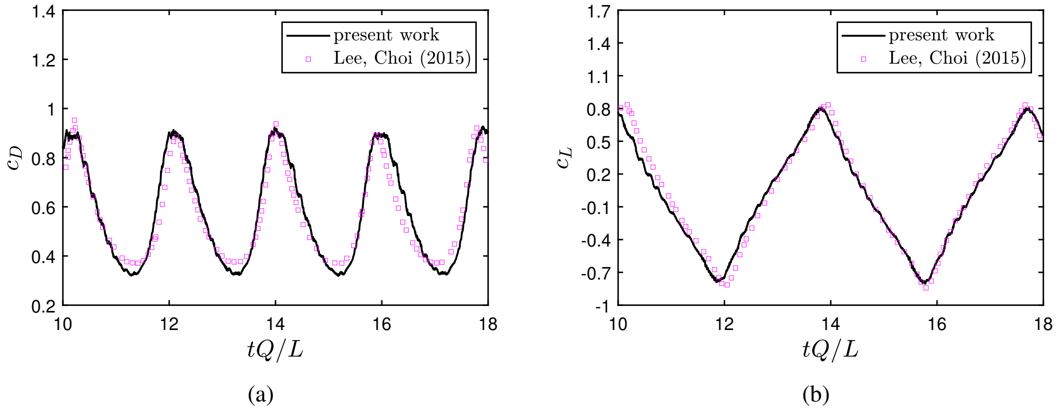
The former assumptions allow to keep the physical consistency with the problem illustrated in [105], despite the fact that very different shell models have been employed. Buoyancy effects are neglected in the present simulation.

The problem is initialized with a uniform flow field at the target bulk velocity Q_∞, then the flag undergoes about 8 transient oscillations before settling to a periodic flapping motion. Time-traces



(a) (b)

FIGURE 6.7: Time traces of the middle trailing edge point (a). Projection of the middle trailing edge point position on the YZ plane (b). Filled circles indicate the time instant at which the vortical structures are shown in Fig. 6.9. The figures refer to the hinged flag test case.



(a) (b)

FIGURE 6.8: Time evolution of the lift (a) and drag (b) coefficients of the hinged flag case within the periodic flapping regime.

of three periods are shown in Fig. 6.7, as proof of agreement with numerical results of the Huang and Sunge [105]. The fidelity of the numerical results is further investigated by inspecting the evolution of the force coefficients $c_D = 2F_z/(\rho_f Q_\infty^2 L^2)$, $c_L = 2F_y/(\rho_f Q_\infty^2 L^2)$. A satisfactory agreement (Fig. 6.8) is found with the results presented by Lee and Choi [106]. Eventually, the instantaneous vortical structures, identified by a Q-criterion (isosurface of Q = 0.0) are reported in Fig. 6.9, showing the characteristic hairpin-like structure shed at each flapping [105, 106].

6.4.2 Accuracy analysis

The hinged flag problem is then used to perform a refinement study with the aim of investigating the accuracy of the FSI procedure. Given the partitioned nature of the present FSI method, two independent refinement analyses have been performed for the fluid and the structural solver. Each solver involves nonlinear phenomena and different convergence rate, as well as different discretization approaches. To this extent errors are evaluated as:

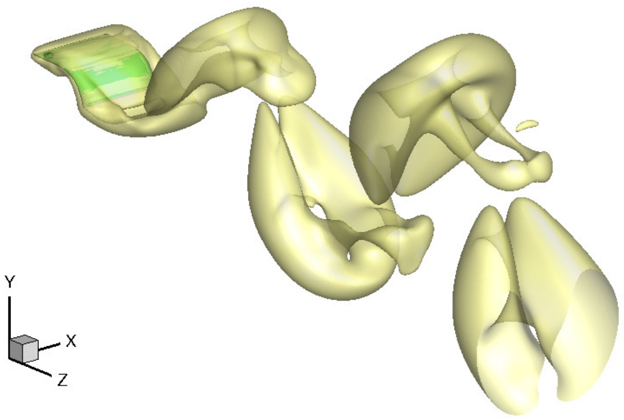


FIGURE 6.9: Instantaneous vortical structures around the flapping flag described by isosurfaces of $Q = 0.0$, at the time instant $tQ/L = 33.89$, corresponding to the filled circle in Fig. 6.7.

$$\epsilon(\theta) = \frac{|\theta - \theta^r|}{\theta^r}. \tag{6.2}$$

First, the FSI convergence rate under Eulerian grid refinement is investigated. Four uniform Cartesian grids are employed with a constant refinement ratio of 1.5. A corresponding Lagrangian marker spacing equal to 0.6 times the Eulerian grid size is adopted. In order to investigate the method accuracy both in space and time, the time step size is selected with the aim of keeping the $CFL = Q_\infty \Delta t/\Delta x = 0.1$; however, the actual CFL value never exceeds 0.2 in each simulation. For the accuracy analyses the fluid domain is $[-L, L] \times [-4L, 4L] \times [-L, 7L]$. The reduced extension of the computational domain and the refinement ratio allow to keep a reasonable computational expense for the analysis. The relative errors are evaluated as in Eq. (6.2) with respect to an extrapolated solution, which was obtained following the procedure proposed by [108]. A summary of the accuracy study is reported in Tables 6.1, 6.2. A quantitative measure

i	refinement ratio	x nodes	y nodes	z nodes	Δ	time step size
1	-	29	113	113	0.0714L	7.143e-3
2	1.5	43	169	169	0.0476L	4.762e-3
3	1.5	65	257	257	0.0313L	3.125e-3
4	1.5	97	385	385	0.0208L	2.083e-3

Table 6.1: Summary of the computational setting for the Eulerian grid refinement study. The shell resolution is kept constant for each case.

j	shell elements per side	refinement ratio	shape functions order	m
1	4	-	3	147
2	6	1.5	3	243
3	9	1.5	3	432
4	14	1.5	3	867

Table 6.2: Summary of the computational setting for the shell h-refinement study. The shell resolution is kept constant for each case.

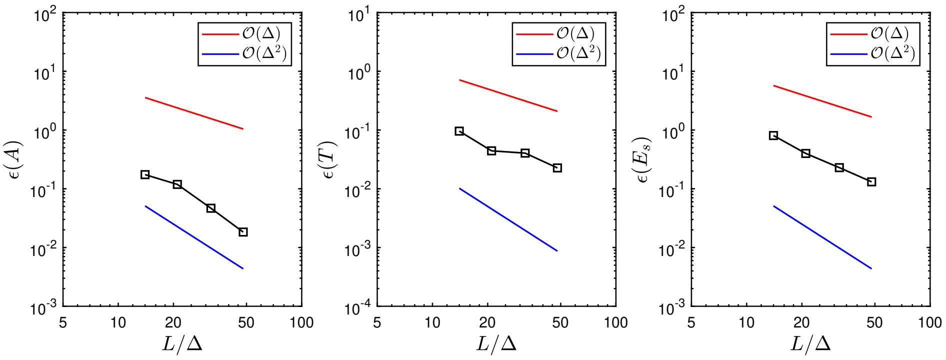


Figure 6.10: Relative error convergence with respect to the Eulerian grid refinement for a pinned flag in a free stream.

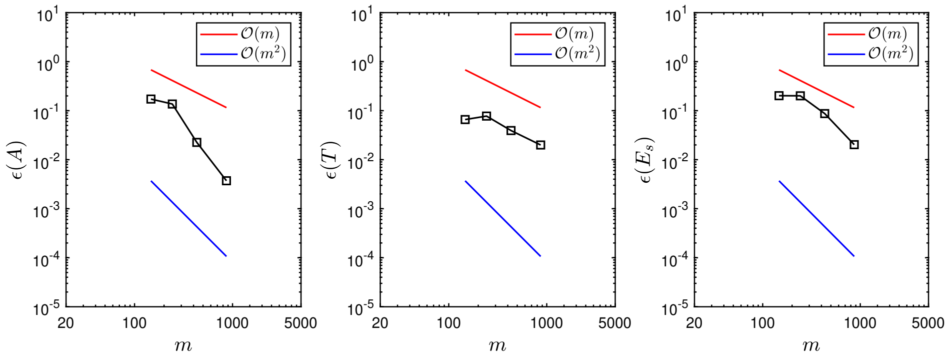


Figure 6.11: Relative error convergence with respect to the shell h-refinement for a hinged flag in a free stream. The overall degrees-of-freedom number m is denoted on abscissa.

of the accuracy is provided by means of the relative error defined in Eq. (6.2) with respect to the selected variable: the flapping amplitude A, the flapping period T, the nondimensional strain energy, defined by:

$$E_s = \frac{\frac{1}{2}\mathbf{u}^T\mathbf{K}\mathbf{u}}{\rho_f Q^2 L^3}. \tag{6.3}$$

The local variables are measured with respect to the tenth period, while the error on the strain energy is computed with respect to a time-averaged value. The relative errors are plotted against grid spacing in Fig. 6.10, showing a consistent convergence rate on the finer grids. In the present case it is verified that the overall FSI procedure does not degrade the order of accuracy of the fundamental solvers.

In the second instance, the FSI convergence rate under shell *h-refinement* is investigated. In the framework of IGA, the h-refinement is intended as a knots insertion procedure which leads to a larger partitioning of the NURBS surface without affecting the surface topology. To reduce the complexity of this study, the basis function order is kept constant and equal to 3, and h-refinement is carried out equally on both NURBS parametric directions. Four uniform knot vectors were employed with a constant refinement ratio of 1.5. The fluid solver setting corresponds to the finer case extracted from the Eulerian grid refinement ($i = 4$ from Table 6.1). Lagrangian marker spacing and time step size both followed this choice. As depicted in Fig. 6.11, the refinement analysis shows an overall accuracy superior to second order in the finer cases. Fig. 6.12 provides the results of the refinement studies in terms of trailing edge middle point transverse location, showing a reduction in the displacement error with respect to the reference solution.

6.4.3 Numerical features

The present subsection is devoted to the investigation of some numerical features highlighted in the description of the numerical method. The hinged flag test is taken as a basis for measuring the effectiveness of the no-slip condition enforcement as well as the added mass effect. The following numerical experiments are performed with the same discretization setting employed in section 6.4.1 with which literature results are matched. First, the Root Mean Square Error of the no-slip condition over the net of Lagrangian markers is evaluated. The velocity field interpolated at the Lagrangian marker should ideally match the local body velocity, because the forcing function is computed from the local velocity difference (5.2). However, the pressure correction step disrupts

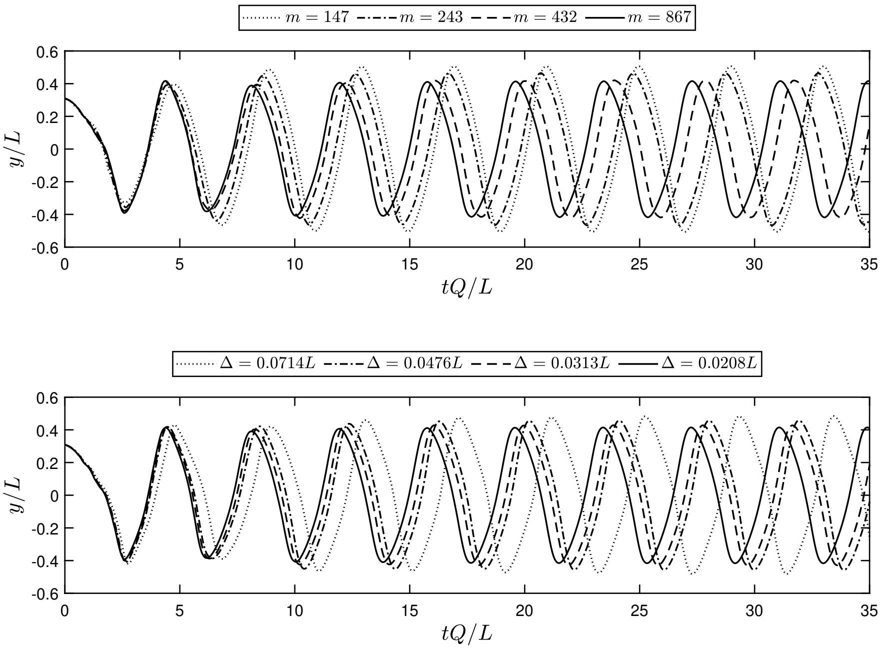


FIGURE 6.12: Time traces of the transverse location of the trailing edge middle point for the hinged flag case at different shell refinement levels (top), and at different Eulerian grid resolutions (bottom). Magenta curves correspond to the reference solution.

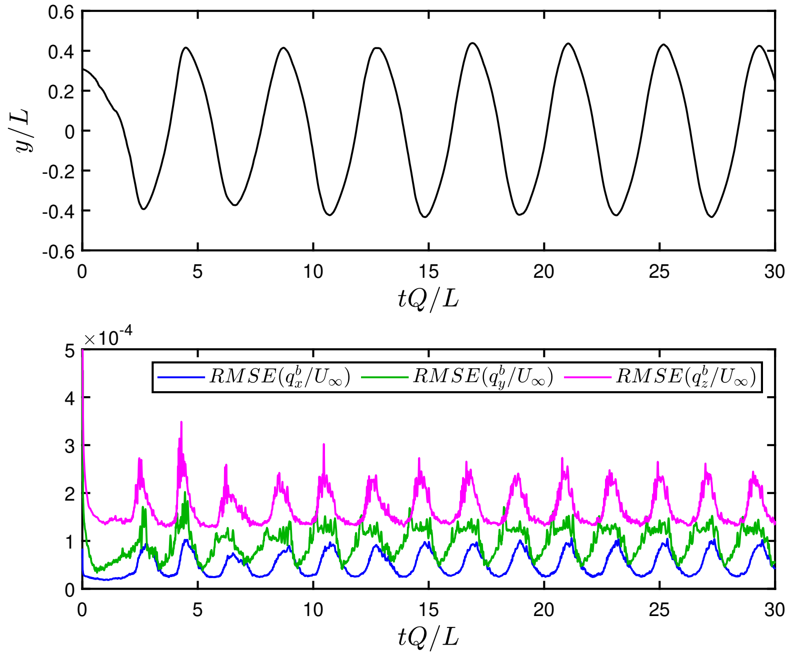


FIGURE 6.13: Time evolution of the root mean square error of the velocity components at the Lagrangian marker location (bottom), compared with the time-traces of the flag trailing edge middle point (top).

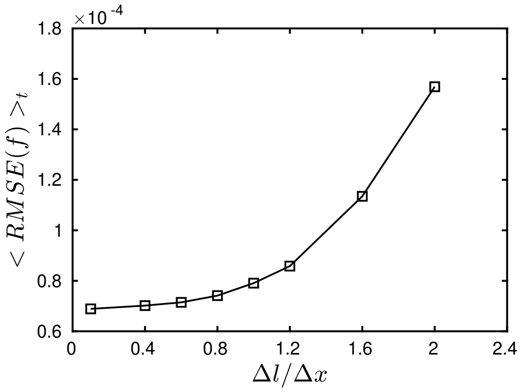


FIGURE 6.14: Time-averaged root mean square error of the normal flux on the surface as function of the ratio between Lagrangian marker spacing and Eulerian grid spacing.

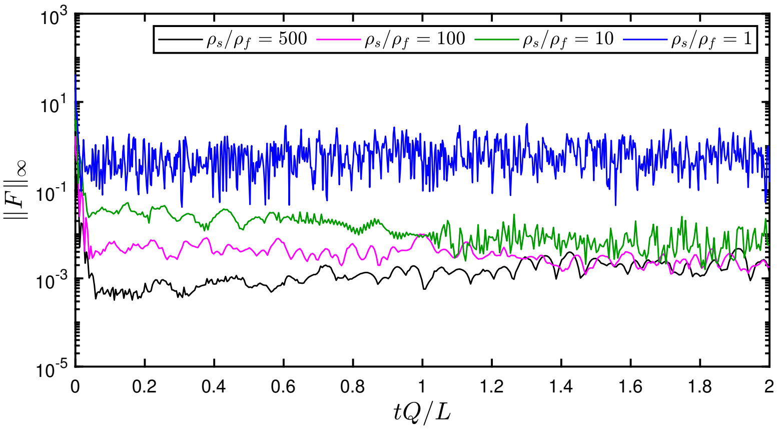


FIGURE 6.15: Time evolution of the L^∞ norm of the FSI coupling error evaluated on the surface degrees of freedom.

the prescribed no-slip condition to enforce the continuity of the fluid solution at local level. The forcing procedure is repeated at each Runge-Kutta (RK) substep, but the detection of the marker position is performed just at the first substep. The final discrepancy between local body velocity and interpolated fluid velocity is measured after the third RK substep, by interpolating again the final velocity field at each marker. As depicted in Fig. 6.13, the RMSE of all velocity components is maintained as a $O(10^{-4})$ fraction of the intrinsic bulk velocity. The same order of magnitude was found in a similar forcing scheme by [74] for both stationary and moving boundaries. In the present test, the error on the velocity component Z is magnified in correspondence of the peak flag deflection (Fig. 6.13), i.e. when the largest local accelerations are encountered. The error is thus amplified when the fluid field is forced to a severe redistribution of the local momentum, like in the case of a uniform flow meeting a normal wall. Beyond a certain threshold the no-slip condition error is insensitive to the CFL as well as to the time-step size of the simulation; according to the experience of the authors, this refinement threshold does not provide a limit in the method efficiency, since it is often less stringent than the refinement necessary to capture the correct FSI dynamics.

Another relevant issue in the no-slip condition enforcement is the local spacing between Lagrangian markers. In the present implementation it represents an independent variable; the markers are collocated in the NURBS parametric space with the aim of matching the local Eulerian grid resolution. A resolution match is necessary to avoid a "porous" representation of the shell in the fluid solution. Numerical experiments show that an optimal Lagrangian marker resolution can be achieved. The hinged flag test is repeated with different marker spacing values. Then, the RMSE of the normal flux is evaluated with respect to a very fine net of check markers ($0.005L \times 0.005L$). On each check marker the normal flux is evaluated with respect to the relative local flow velocity $f_c = (\mathbf{q}_c - \mathbf{u}_c) \cdot \mathbf{n}_c$. The time-averaged surface flux (Fig. 6.14) is assumed to be converged at the relative spacing $\Delta l/\Delta x = 0.6$. Hence, this value is considered as an optimal choice for all the test cases presented. It should be mentioned that the marker projection is repeated at each time step, thus a large surface stretching would not affect the mean surface flux value.

The accuracy of the load computation is investigated, too. High density-ratio cases ($\rho_s/\rho_f >> 1$) allow for a time-staggered FSI coupling, where the solution at previous time step is used to compute loads for the structural solver. An explicit treatment becomes less accurate in time as the density-ratio gets closer to one. The hinged flag case is repeated with the strong coupling procedure at four decreasing density ratios (ρ_s/ρ_f = [500, 100, 10, 1]), and the effect of the time-lag is measured by the L^∞ norm of the hydrodynamic forces error between the predicted and the corrected solution (the reader should refer to the iterative coupling procedure described in Section 5.5). The first 1000 time steps are presented (Fig. 6.15) to isolate the effects of different density ratios from the effects of a markedly different dynamics. A different order of magnitude in ρ_s/ρ_f corresponds to a different order of magnitude in the force error. Until the force error has the same order of magnitude as the no-slip conditions, the weak coupling could be considered sufficient for reproducing the global FSI dynamics. The hinged flag case can be taken as a numerically challenging example in terms of added mass effects because of the presence of rigid body modes. The predictive capability of the iterative coupling procedure is tested against an experimental investigation in Section 6.4.7, where a significant numerical challenge is provided by a shell-to-fluid density ratio lower than one.

6.4.4 Sealed channel

The intrinsic error in the interface conditions is further investigated by assessing the volume conservation on a simple test-case. As a consequence of the approximate interface conditions, the fluid volume conservation within a subset of the computational domain suffers from a systematic error. This is an issue of fundamental relevance in the simulation of capsules and closed vesicles, where such error can lead to a spurious changes in the internal volume. Several efforts have been done to address this issue in direct-forcing schemes, with different degrees of complication [109, 110, 47]. In the present implementation, the causes of the volume conservation error on a subset of the fluid domain are connected mainly to the aforementioned effects of the pressure correction, as well as to the fact that the forcing function comes from the interpolation of a locally non-solenoidal field. It should be pointed out that staggered-grid schemes showed superior volume conservation properties with respect to collocated-grid schemes [111]. On the former schemes, further developments were made by using forcing/spreading operators with intrinsic conservation properties, ensuring that the interpolated velocity is constructed to be nearly divergence-free [47].

In order to limit the complexity of the interpolation/spreading operations, a simple iterative forcing is proposed in the present section: the volumetric flow rate leakage can be reduced up to two order of magnitude by reiterating the forcing procedure within each Runge-Kutta substep

(Fig. 6.17). The repetition of the forcing/spreading procedure produces smaller local volume forces at each iteration which bring little changes in the momentum. The reduction in the interface velocity error is thus linked to a relative reduction of the interpolation/spreading errors. However, the accuracy enhancement is limited by the subsequent pressure correction stage.

This solution is assessed by measuring the mass flow rate in a viscous channel subject to a constant pressure gradient, but completely sealed by an elastic shell. As shown in Fig. 6.16, a channel of dimension $[-0.125H, 0.125H]\times[0, H]\times[-2H, 2H]$ is set to have a viscous wall condition on the Y faces, and periodic conditions on the other faces. The fluid domain is discretized by the spacing $\Delta_z = 0.007H$, $\Delta_x = \Delta_y = 0.01H$, whereas the shell is discretized by 32 second order elements. A constant pressure gradient $\partial p/\partial x = -12(\rho_f Q^2)/(Re\, H)$, corresponding to the driving load required to get the mean channel velocity Q for a laminar, planar flow, is applied to an initially quiescent fluid field. The Reynolds number and the shell properties are set to $Re = 100$, $E/(\rho_f Q^2) = 0.01H$, $t = 0.01H$, $\rho_s/\rho_f = 100.0H$, $\nu = 0.3$, respectively. After a short transient behavior, fluid and structure reach an equilibrium configuration where the elastic force provided by the deformed shell balances the pressure gradient applied at the channel.

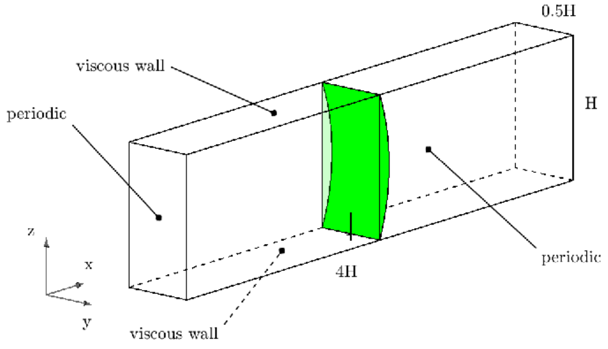
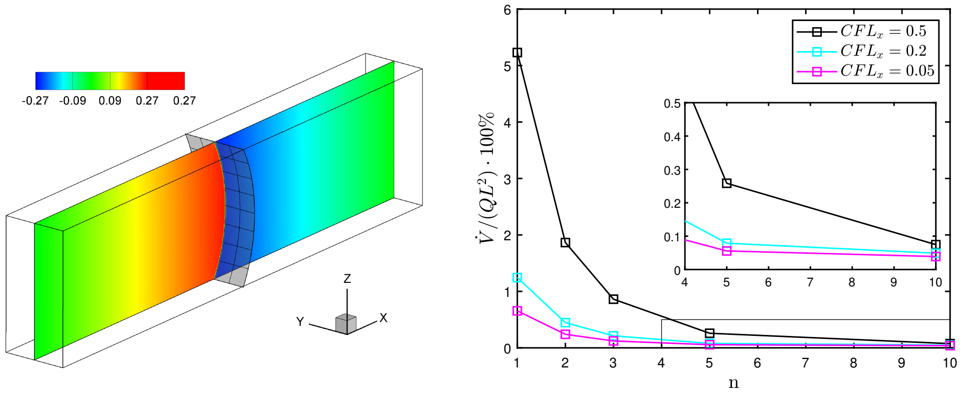


Figure 6.16: Schematic of the computational domain and boundary conditions for the FSI simulation of a channel sealed by an elastic membrane.

The effect of the iterative forcing is evaluated by measuring the volumetric flow rate fraction across the channel section for several forcing loops $n = \{1, 2, 3, 5, 10\}$, at different time step sizes, corresponding to $CFL_x = \{0.5, 0.2, 0.05\}$. The flow rate $\dot{V}$ is given in percentage of the target volumetric flow rate of an incompressible laminar flow between infinite flat plates, $\mathcal{V} = \dot{V}/(QL^2)\cdot 100\%$. The reduction in the flow rate with the number of forcing loops follows a nearly asymptotic trend, which is mitigated by reducing the CFL number. As suggested by [74], a computationally-effective choice for the reduction of the flow leakage across the interface would occur at $n = 2$. At different CFL conditions, the error is more than halved after the second forcing iteration. At $CFL = 0.2$, the relative error drops below 1% after two iterations: $\mathcal{V}_{n=2} = 0.447\%$, $\mathcal{V}_{n=3} = 0.213\%$, $\mathcal{V}_{n=5} = 0.0785\%$.

6.4.5 Inverted flag

The dynamics of a flag invested by a uniform flow was broadly investigated in the literature to find the conditions under which the self-excited flapping occurs. In this context, the configuration where the flag is free at the leading edge and clamped at the trailing edge is referred to as *inverted flag*. Besides the purpose of understanding the physics of the self-excited motion, this problem raised interest for the possibility of energy harvesting by means of a piezoelectric device. In fact,



(a) (b)

FIGURE 6.17: Pressure contours $(p- < p >)/(\rho Q^2)$ of a sealed channel subject to a prescribed constant pressure gradient in the equilibrium configuration at $CFL_x = 0.2$ (a). Volumetric flow rate within the sealed channel at the equilibrium configuration; values are given in percentage of the target volumetric flow rate of an incompressible laminar flow between infinite flat plates (b).

once the kinetic energy of the fluid flow is converted in strain energy of the flag, the piezoelectric layers accumulate electric charge in response to the mechanical stress. An efficient application of the concept requires an accurate design of the flag, which should easily become unstable at low inflow velocity and have an high excitation amplitude [112]. Experimental and numerical investigations have shown that the dynamic response of the flag can undergo three regimes. A straight regime, where the flag just vibrates around the initial equilibrium position; a flapping mode, where the flag undergoes large amplitude oscillations; a deflected mode, in which the flag flutters with small oscillations around a highly bent configuration. The flag motion which maximizes the harvested energy depends on many parameters such as the flag geometry, bending stiffness, inflow velocity. Thereby a computational model which accurately reproduces the mechanics of the problem would be an effective tool for optimization purposes or design space exploration [113, 114, 115].

In order to compare results with the experimental campaign carried out by Kim et al. [112], the following governing parameters have been considered for the present test:

$$\beta = \frac{Eh^3}{12\,(1-\nu^2)\rho_f Q^2 L^3}, \quad \mu = \frac{\rho_s h}{\rho_f L}, \quad Re = \frac{\rho_f Q L}{\mu_f}, \quad a = \frac{H}{L}, \tag{6.4}$$

where β indicates the nondimensional bending stiffness, μ is the mass ratio, Re is the Reynolds number, and a is the flag aspect ratio. A set of five simulations $\beta = [0.06, 0.1, 0.18, 0.26, 0.32]$ contemplating the three different regimes is carried out. For each regime the parameter β is varied by changing the Young's modulus, whereas the other nondimensional parameters are kept constant: $\mu = 0.29, \quad a = 1.1$. All simulations are performed at $Re = 200$. Even though experiments by Kim et al. [112] were conducted at higher Reynolds numbers ($Re \approx 10^4$), it has been found that the leading edge vortex dynamics, crucial for the large amplitude flapping, was reproduced similarly at much lower Reynolds numbers [113]. A fluid domain of dimension $[-2L, 2L] \times [-5L, 5L] \times [-2L, 10L]$ is employed, and a local grid size of $\Delta = 0.02L$ is adopted in the flapping region $[-2L, 2L] \times [-2L, 2L] \times [-0.5L, 3.5L]$; however the grid was stretched far from the body with a cubic spline law. The fluid domain arrangement is similar to that of the

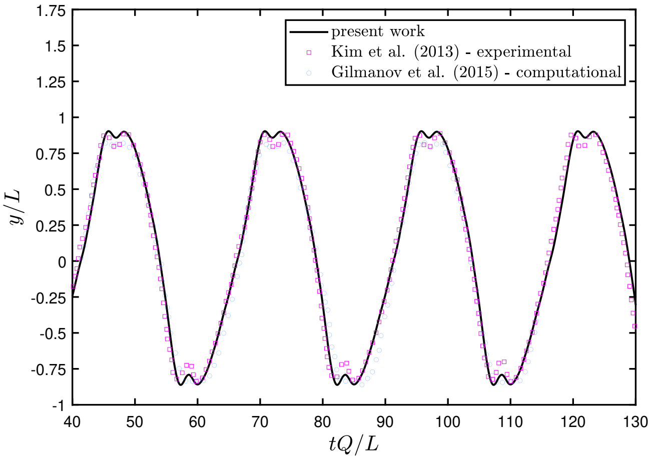
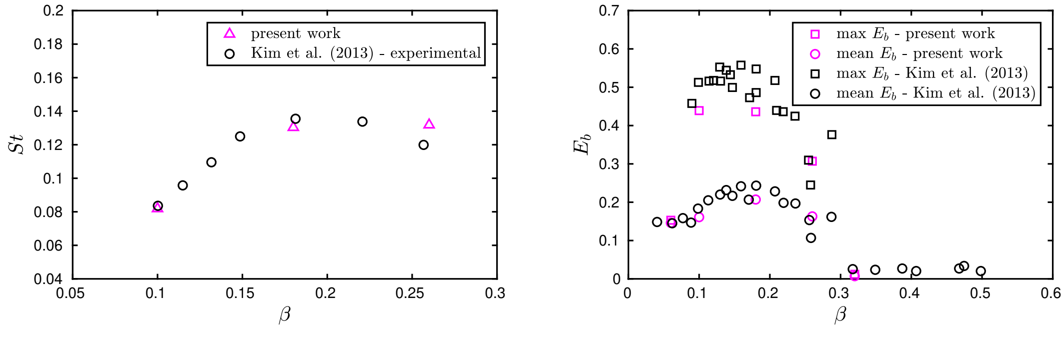


FIGURE 6.18: Time trace of the middle leading edge point y coordinate for $\beta = 0.1$; superposition with the experimental results by Kim et al. [112] and computational results by Gilmanov et al. [14].



(a) (b)

FIGURE 6.19: Strouhal number as a function of the nondimensional bending stiffness for the flapping regime of the inverted flag case (a). Mean (circles) and maximum (squares) normalized bending energy against nondimensional bending stiffness (b).

hinged flag test, sketched in Fig. 6.6. The flag is clamped on the trailing edge at an angle equal to 1.5 degrees with the inflow direction. This allows to brake the symmetry in the problem and accelerate the rise of unstable modes without affecting the hysteresis loop [112], even though a slightly asymmetric flapping is observed. In the experiments, the gravity force acted along the spanwise direction, where negligible deflections were observed, therefore its contribution was omitted in the present simulations. The shell is discretized by 14 cubic isogeometric elements in the flow direction, whereas 6 quadratic elements are employed in the spanwise direction, since the flapping dynamics is expected to be mostly bi-dimensional. A constant nondimensional time step of $\Delta tQ/L = 10^{-3}$ is used; this allows to have more then 12000 time steps within one period at $\beta = 0.26$.

The strain energy associated with the bending modes is crucial for energy harvesting applications. Therefore, the nondimensional bending energy is measured along the flag center-line on a local curvilinear axis $s = s(\mathbf{x}(t))$. The definition provided by Kim et al. [112] is:

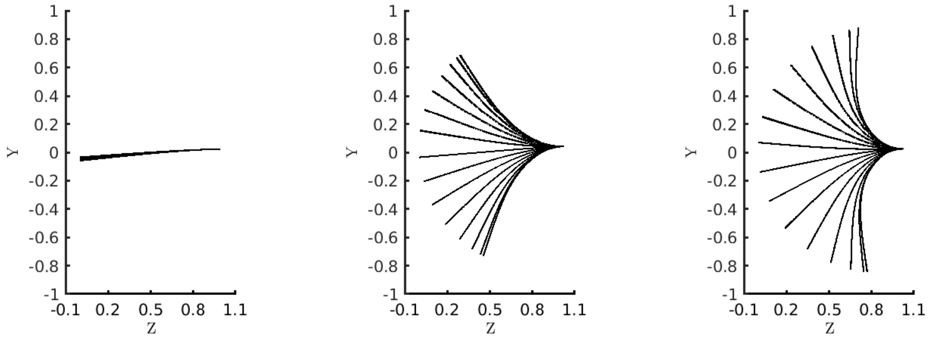
$$E_b(t) = \frac{\frac{1}{2}\int_0^L BK(s,t)^2 ds}{\rho Q^2 L^2} = \frac{1}{2}\beta \int_0^1 \kappa(s,t)^2 d\left(\frac{s}{L}\right). \tag{6.5}$$

Mean and maximum bending energy values are plotted for the set of bending stiffness value considered (Fig. 6.19 b). As shown by the maximum E_b values, two bifurcations in the flag dynamics are found in proximity of $\beta = 0.1$ and $\beta = 0.3$, consistently with the experimental observations of Kim et al. The simulations confirm the nature of the nonlinear dynamics and the presence of two fixed points and one limit cycle in between. A detailed stability analysis of the problem was provided by [116]. Furthermore, the maximum bending energy associated to the limit-cycle flapping mode is found to be substantially larger than the deflected mode; despite a higher bending stiffness, the flag in the flapping mode can bend more downstream (Fig. 6.20) and achieve higher curvature values near the trailing edge. This was widely recognized in experimental and numerical studies [112, 113].

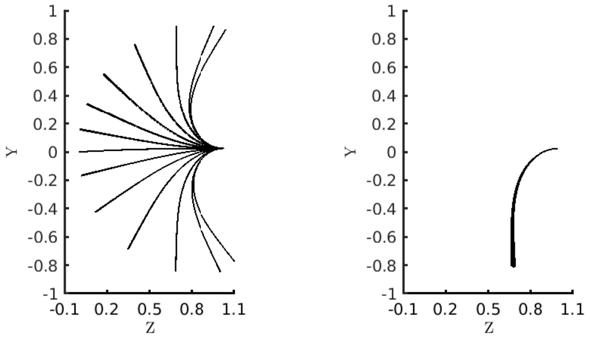
The time traces of the vertical displacement of the leading edge is provided in Fig. 6.18 for the largest amplitude flapping ($\beta = 0.1$). A comparison with the experimental measurements shows a good agreement. Our method is clearly able to capture the inertial over-bending of the tip when the flag reaches the amplitude peak: this is reflected in a local minimum in the leading edge time-traces (Fig. 6.18) which was observed in the experimental investigation, too. Furthermore, the computational data provided by [14] are also reported to show the consistency with simulations carried out at the actual Reynolds number, with a highly turbulent wake. Finally, the Strouhal number (defined as $St = fA/Q$) plot reveals a satisfactory agreement in terms of flapping amplitude and flapping period (Fig. 6.19 a). A little mismatch in the Strouhal number and bending energy is found for $\beta = 0.26$. This can be attributed to the sensitivity to initial conditions of the system in the transition region from straight mode to flapping mode; as a matter of fact, Ryu et al. defined a further regime, named *biased regime*, in the phase space spanned by Reynolds number and bending stiffness [116].

6.4.6 L-shaped plate

The present case consists of a L-shaped plate invested by a wall-bounded flow, as sketched in Fig. 6.21. The flexible plate is clamped at its shortest edge, therefore the shank of the plate interacts with the boundary layer. This test case was first proposed by Glück et al. [117] as benchmark for lightweight structures occurring in Civil Engineering. This was chosen to corroborate the sensitivity of the implementation against an FSI case dealing with a complex strain pattern involving torsion and bending. According to [117] the plate was kept fixed until $t = 20\,s$, when the flow field is fully developed and a periodic solution is reached. Thereafter, the shell solver is activated, and the plate is able to develop structural oscillations. Because of the sudden release of the plate, the displacement overshoots several times the steady state deformation, which are reached after a large transient. The numerical experiment is repeated three times for different values of the nondimensional elastic modulus $E^* = E/(\rho_f Q^2) = [1.165 \cdot 10^5, 0.350 \cdot 10^5, 0.177 \cdot 10^5]$. The first value of E^* was adjusted to achieve the tuning of the main vortex shedding frequency and the first eigenfrequency of the plate [117], so that self-excited oscillations occur once the transient is finished. A schematic of plate dimensions is reported in Fig. 6.21. Thickness, density, Poisson's ratio and Reynolds number are assumed to be $h = 0.01L$, $\rho_s/\rho_f = 1200.0$, $\nu = 032$, $Re = 200$. The fluid domain is $[4L, 4L] \times [0, 5L] \times [-8L, 25L]$. Here, the forward domain region along Z-direction is fundamental for the test outcome since it affects the thickness of the boundary layer investing the plate. The flow fleld is uniformly discretized in the region $[4L, 4L] \times [0, 2.5L] \times [-2L, 4L]$,



(a) $\beta = 0.32$ (b) $\beta = 0.26$ (c) $\beta = 0.18$



(d) $\beta = 0.10$ (e) $\beta = 0.06$

Figure 6.20: Snapshots of the inverted flag motion for decreasing bending stiffness values.

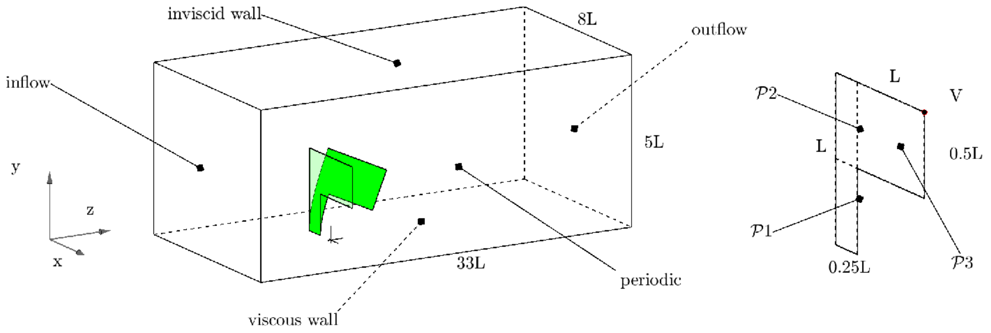
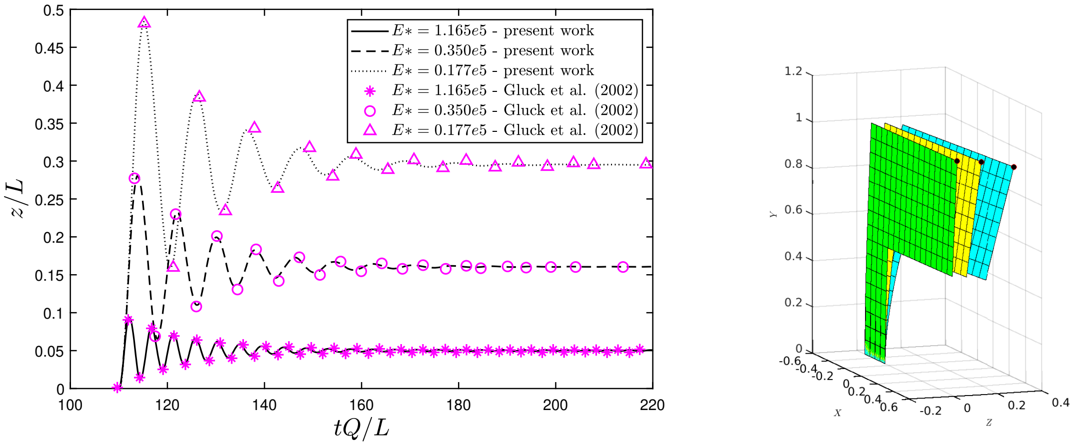


Figure 6.21: Schematic of the computational domain and boundary conditions for the FSI simulation of an L-shaped plate (left). Proportions of the plate and location of the vertex V (right).

where local grid spacings $\Delta x = \Delta y = 0.02L$, $\Delta z = 0.016L$ are employed. The present grid resolution was deemed sufficient to capture the boundary layer gradients. The plate is represented by means of three patches (Fig. 6.21) to achieve a simple one-to-one correspondence of the interface control points. By doing so, the necessary C^1 continuity is enforced with a trivial master-slave approach [5]. Third order basis functions and 16 elements per unit length L have been employed



(a)

(b)

FIGURE 6.22: Time-traces of the streamwise coordinate of the vertex V of the L-plate, from plate release to periodic oscillations regime (a). Superposed plate configurations at $tQ/L = 220$ for different E^* values; the black dot highlights the displacement of the vertex V (b).

for the shell discretization. Calculations are performed with a nondimensional time step size $\Delta t = 0.005tQ/L$, in order to resolve one oscillation period with the first plate eigenfrequency by at least 50 time steps.

For $E^* = 1.165 \cdot 10^5$ the plate is damped up to a self-excited periodic oscillation state, therefore the condition described by Glück et al. is met. The other cases led to steady plate configurations. As pointed out in Fig. 6.22, the mean displacement of the vertex V is accurately reproduced in the three cases.

The vibrating plate $E^* = 1.165e5$ was found to oscillate with a dimensional frequency $f = 1.16Hz$, against the value $f = 1.19Hz$ measured by Glück et al. A spectrum plot of the vertex V streamwise oscillation is provided in Fig. 6.23. The agreement with results from Glück et al. can be considered satisfactory. This case had to deal with very small amplitude oscillations in the periodic regime, whose condition represents a demanding task, given the approximate enforcement of the no-slip condition. The largest peak-to-peak length is expected to be entirely contained in one fluid cell. However, the information transfer operator is sufficiently sensitive to the local flow field changes to capture such self-excited oscillations. A comparison of the transient regime with the results available in literature are presented in Fig. 6.23 (left), where an envelop curve of the plate transient was fitted on the peaks of the present simulations and superposed to the oscillation peaks reported in the reference work. The linear trend over a logarithmic time scale actually confirms a pure viscous damping mechanism.

6.4.7 Clamped seaweed

The next case is selected in order to test the robustness of the strong FSI coupling against a problem involving large accelerations and a low shell-to-fluid density ratio. The case concerns the reconfiguration of a buoyant seaweed induced by the seawater flow. Luhar and Nepf performed laboratory-flume experiments on a single seaweed-like specimen to investigate the seaweed posture in the marine environment [118]. The experimental data have already been used in the literature to check iterative coupling algorithms with reduced computational expense

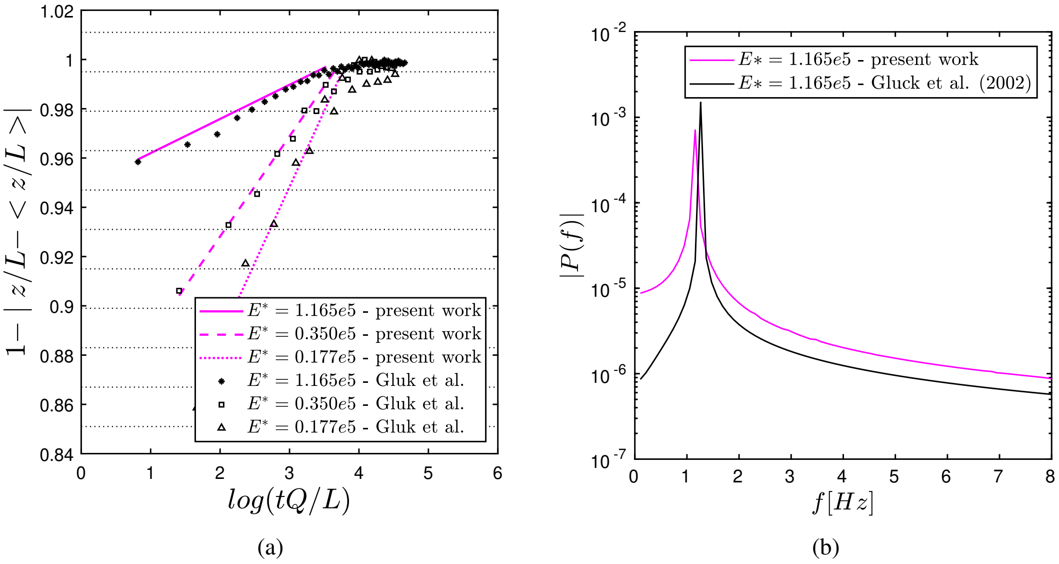


FIGURE 6.23: Normalized transient envelop of the stramwise displacement of the L-plate vertex V (a). Dashed horizontal lines show the local grid spacing in Z-direction; self-excited oscillations occur within one fluid cell. Frequency spectrum of the vertex V streamwise displacement for $E^* = 1.165 \cdot 10^5$ in the self-exciting regime (b).

[107, 43]. A blade resembling a specimen of the seagrass *Zostera Marina* is modeled as a clamped plate of width L, height $H = 5L$ and thickness $h = 0.2L, 0.04L$ [118]. The fluid domain extends to $[-4.5L, 4.5L] \times [0, 15L] \times [-5L, 20L]$. Inviscid wall and periodic boundary conditions are applied respectively in directions Y and X; a radiative outflow boundary condition is used at the flow outlet. In the reference experiment [118], the blade was positioned above the bottom boundary layer to get a uniform flow speed over the length of the model blade by means of the blade holder. This condition is reproduced numerically by setting a free-slip condition on the bottom wall. A schematic of the fluid domain and boundary conditions is provided in Fig. 6.24. The shell is discretized by 6 quadratic elements in the width direction and 14 cubic elements in the height direction because the deformation is expected to be mostly two-dimensional. Two simulations are performed in fluid-dynamic similitude with the experimental conditions [118], providing a Reynolds number $Re = 1490$. The buoyant force, which plays a relevant role in the seaweed reconfiguration, is applied as a net surface load $f_g = g/(\rho_f)(\rho_s/\rho_f - 1)h$ to be integrated over the surface. Simulations are carried out on a fluid grid uniformly refined within the subset $[-4.5L, 4.5L] \times [0, 8L] \times [-2L, 5L]$ where a local grid spacing of $\Delta = 0.025L$ is achieved; a constant nondimensional time-step size $\Delta t = 0.002Q/L$ is employed in all the simulations, with the CFL number never exceeding 0.2. A resume of the simulations and global outcomes is reported in Table 6.3. The results are compared in terms of drag coefficient $c_D = 2F_z/(\rho_f Q^2 LH)$, and time-averaged streamwise and transverse displacements of point P (Fig. 6.24). In each simulation the fluid density and dynamic viscosity of the seawater at laboratory conditions are assumed: $\rho_f = 1025.0kg/m^3$, $\mu = 0.0011Ns/m^2$. The materials employed by Luhar and Nepf in their experiments are modeled as follows: Sylicone foam ($\rho_s/\rho_f = 0.678$, $E/(\rho_f Q^2) = 1.905e4$, $\nu = 0.4$, $h/L = 0.2$, $4gL/Q^2 = 3.831$) and HDPE ($\rho_s/\rho_f = 0.951$, $E/(\rho_f Q^2) = 3.544e7$, $\nu = 0.46$, $h/L = 0.04$, $gL/Q^2 = 3.831$). Under the experimental conditions, the flexible plates settle down to a nearly static conditions, were small amplitude oscillations in the ZY plane are observed. The

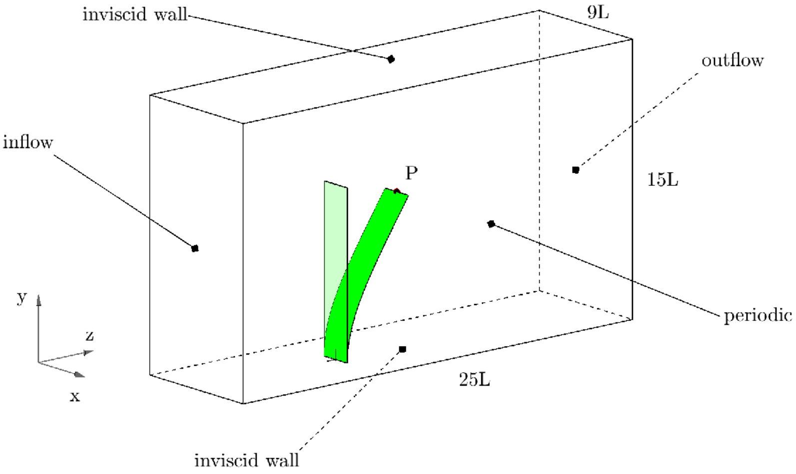


FIGURE 6.24: Schematic of the computational domain and boundary conditions for the FSI simulation of a seaweed.

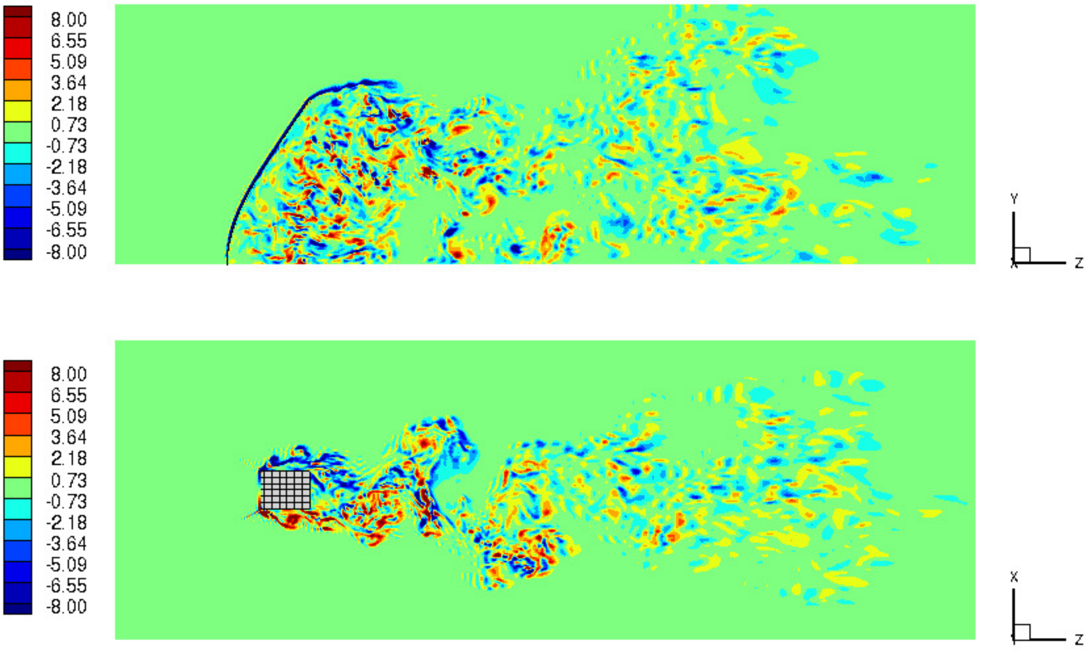
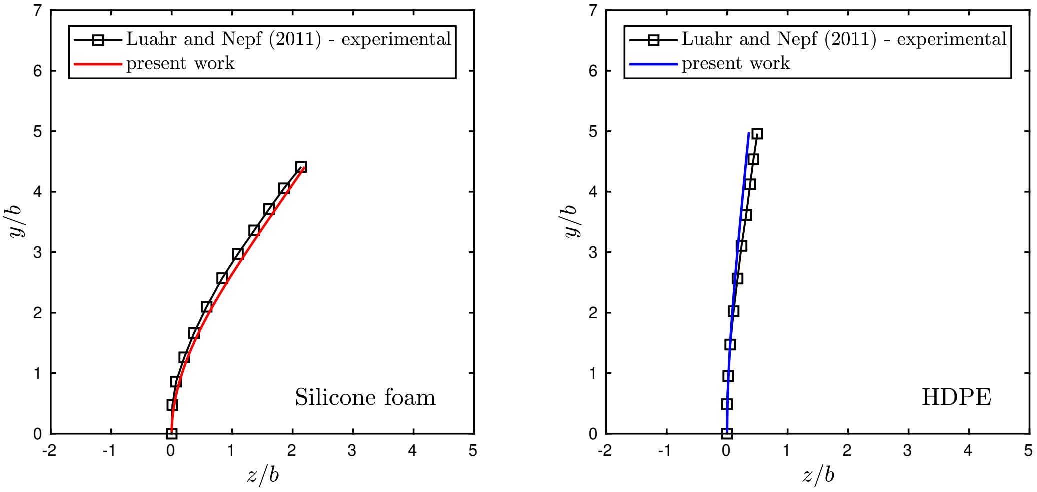


FIGURE 6.25: Instantaneous out of plane vorticity magnitude $\omega L/Q$ over the $X = 0.0$ section (top) and the $Y = 2.5L$ section (bottom). The contours refer to the Silicone Foam seaweed specimen at experimental inflow conditions, at $t = 10.0Q/L$. Contours are plotted over a subset of the fluid domain.

steady configuration achieved in the present simulations corresponds to the minimal drag force as compared with the corresponding rigid plate. The reconfigured plate at experimental inflow conditions is superposed to the experimental observation in Fig. 6.26, showing a satisfactory agreement. The instantaneous contours of out-of-plane vorticity magnitude are plotted at two

	Re	ρ_s/ρ_f	c_D	D_z/b	D_y/b
present work	1490	0.678	1.12	2.22	0.61
	1490	0.951	1.40	0.37	0.03
Luhar and Nepf [118] (experimental)	1490	0.678	1.15	2.14	0.59
	1490	0.951	1.39	0.52	0.05
Tian et al. [107] (computational)	1600	0.678	1.03	2.14	0.59
Kim et al. [43] (computational)	1600	0.678	1.12	2.19	0.59

TABLE 6.3: Mean drag coefficients and mean streamwise and transverse displacement of the point P. A comparison with results available in literature is provided.



(a) (b)

FIGURE 6.26: Steady blade posture for two different material properties with superposed experimental observations. Silicone foam (a) and HDPE (b).

cross-sections of the computational domain (Fig. 6.25). Despite complex vertical structures are exhibited in the wake, the characteristic vortex-shedding pattern of a bluff body can be recognized in the ZX plane. A shear layer vortex periodically detaches from the free edge of the plate. The present test proved that the method is able to cope with a very low shell-to-fluid density ratio without incurring in numerical instabilities. In all simulations, the iterative FSI coupling converged to the predefined tolerance in 2-8 iterations, depending on the magnitude of the local accelerations in the shell.

Eventually, a general over-bending of the plate is observed in the simulations at a lower density ratio. This is attributed to the absence of the blade thickness, which could lead to a smaller flow recirculation region in the body downstream.

7 Electrophysiological excitation of thin muscular tissues

7.1 A computational approach to electrophysiology

Computational methods for the simulation of electrical excitation of biologic tissues have experienced tremendous advances in the last decades. The diffusion of high-performance-computing infrastructures allowed the computational science community to realize accurate organ-scale simulations of the propagation of electrical impulses over the muscular tissues. Detailed anatomical models have been developed, both on cellular- and tissue-level. The purposes of such models are the comprehension of multiscale mechanisms, the study of pathological states [20], the conceptualization of next-generation prosthetic devices [119], and ultimately, the possibility to perform clinically relevant studies with patient-specific simulations [21, 120]. However, the intrinsic multiscale nature of the electrophysiological phenomena still offers significant challenges to computational analysis. The acceptance of computational methods in the medical community will largely depend on their efficiency, robustness and reliability, which are currently open avenues of research [121]. On the other hand, computational approaches are recognized as a valuable tool to assist the experimental procedures bearing in mind that replicable in-vivo measurements are certainly difficult to obtain.

The roots of the mathematical formulation of electrophysiological models date back to the work of Hodgkin and Huxley [122] on modeling the electrical propagation in squid giant axons. Following their work, a large number of electrophysiological models, especially focused on myocardial cells and pacemaker cells have been proposed in the literature [123]. The computational research has been focused mainly on cardiac tissues [124], providing in recent years multiphysic, multiscale solvers for fully coupled electro-mechano-fluidic ventricle simulations that incorporate detailed mechanical contraction models, crossbridge dynamics, and the fluid-structure interaction [124]. Recently, similar electrophysiological models found application in the study of the locomotion of simple organisms, with the scope of understanding its nature or building artificial swimmers with in-vitro cultured tissues. [22, 125].

Currently, the scientific community distinguishes two main classes of electrophysiological models: biophysical and phenomenological models. Biophysical models [126, 127, 128] aim at describing the complex exchanges occurring at the excitable cell membrane, by quantifying the transmembrane fluxes of the main ionic species (Calcium, Potassium, Sodium and Chlorides ions) through the ion channels. Differently, phenomenological models [129, 130, 131] neglect the sub-cellular ion dynamics and propose simplified formulations which mimic the transmembrane potential shape. Regardless of their nature, all deterministic electrophysiological models result in a reaction–diffusion equation [132] whose reaction term falls into one of the model's category aforementioned. In this scenario, the choice of the numerical method strongly depends on the range of temporal and spatial scales which one aims at describing. The molecular mechanisms develop at scales orders of magnitude smaller than those of muscle contraction (see Fig.

7.1), but each stage of the electrophysiological chain is connected with biochemical and mechanical processes. A full level modeling is practically impossible, but the mutual interaction of the different scales is often necessary to obtain physiologically meaningful results.

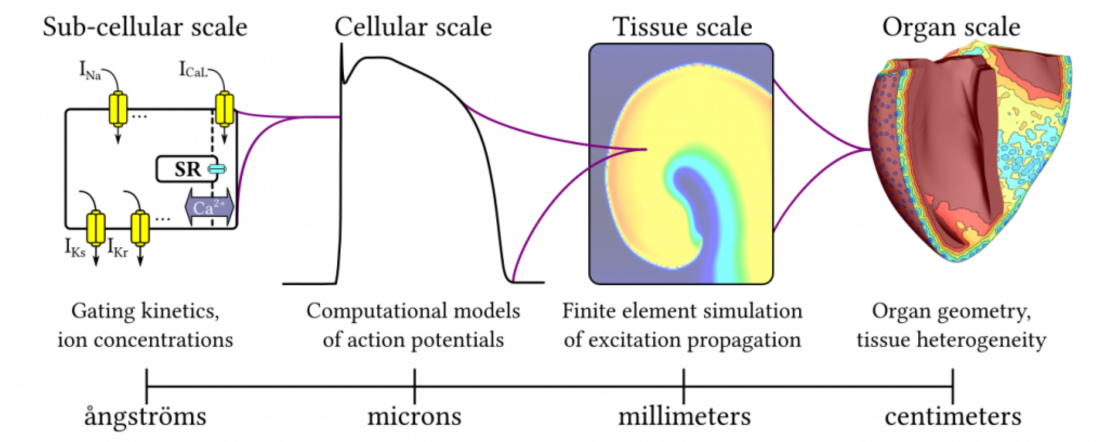


FIGURE 7.1: Spatial scales of the mechanisms contributing to the electrophysiologic activation of the heart. Each of these stage has a strong functional coupling with the others. Figure taken from `https://cardsslab.org/`.

7.2 Thin muscular tissues

The present work aims at providing an efficient numerical tool for the simulation of the electromechanical activation of thin muscular tissues. Recent experimental and computational studies have turned their attention to the electro-mechanics of thin tissues, as objects of layer-wise cardiac tissue engineering [133], or actuators for bio-inspired robots for aquatic locomotion [125]. Different research groups have found that artificial cultured tissues can match certain temporal, spatial, or force regimes typical of biological muscle [134], although they cannot fully replicate all the electro-chemical mechanisms, nor can they use the same high-density energy sources [22]. Unlike mechanical actuators, cultured tissues demonstrated moderate contractile force, fast actuation, large strain, possible remote control by electric field imposition, and low power consumption [135]. Thus, soft engineered tissues proved their superiority with respect to classical robotics due to their ability to replicate the hierarchical architecture typical of biological systems, which allows for the replication of some complex biologic functionalities. The efficiency of artificial muscles can be easily enhanced if the biochemical actuation is coupled with the ability of shell-like structures to carry large transversal loads with minimal volume. Recently, Feinberg et al. [22] built thin micro-actuators with a combination of a polydimethylsiloxane substrate and rat ventricular cardiomyocytes, cultured on top of the former. In their experiments, the substrate and the interface have been designed to provide a strong anisotropy in the tissue architecture, which induces a visible bending of the substrate during contraction of the excitable cells. Based on the ability of 2D planar shapes to adopt complex 3D conformations, they leveraged the inherent contractility of the cardiomyocytes to create a variety of soft robotic devices [22]. Nawroth et al. [125] reported the construction of a freely swimming medusoid (see Fig. 7.2), built from chemically dissociated rat tissue and silicone polymer. They employed a traditional reverse engineering approach to build a millimeter-scale construct from living and synthetic materials that emulates the jellyfish locomotion. They demonstrated how a synthetic tissue with a shell structure can be deigned to mimic the biological function of a living organism.

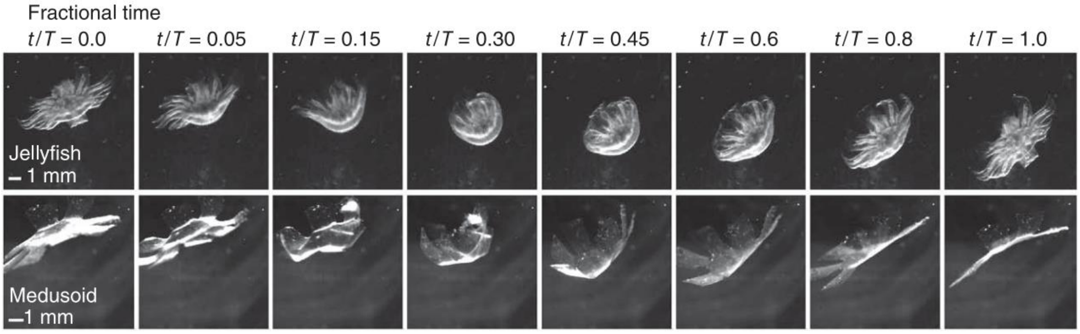


FIGURE 7.2: Medusoid engineered to mimic the jellyfish-like stroke kinetics. The medusoid is built from chemically dissociated rat tissue and silicone polymer layered in a thin structure. Figure adapted from [125].

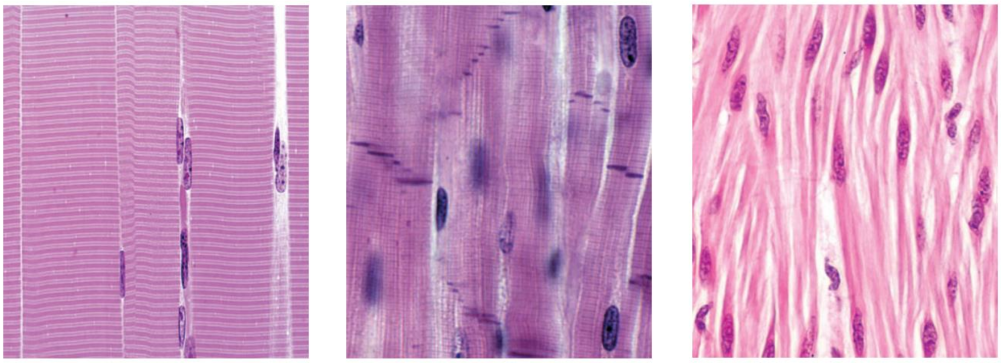
In this scenario, a computational method for the activation of thin tissues could be a valuable tool for a preliminary design space exploration, or a support to dissect complex biologic events into simple mechanistic functionalities to be mimicked by engineered solutions. Numerical tools can provide guidance for experiments as well as shed the light on such strongly coupled problems [136].

The present chapter is devoted to the introduction of the electrophysiological model adopted in our computational framework.

7.3 Electrophysiological activation of syncytial muscles

From the continuum mechanics perspective, muscles can be classified as active materials, since the tissue strain is not uniquely determined by external forces. A muscle is a soft tissue which fulfills different purposes such as producing a movement and displacing substances in the organism [137]. There exist three types of muscular tissue – skeletal, cardiac and smooth – which differ in location, microscopical anatomy and regulatory mechanism (see Fig. 7.3). Skeletal muscles provide voluntay displacement of bones through contraction, and they are attached to bones through tendons. The tissue of skeletal muscles is defined *striated*, since it consists of multiple fascicles of cylindrical unbranched fibers made of elongated cells. As a consequence, skeletal muscles have a clear directionality. Cardiac muscles are involuntary, striated muscles that constitute the main tissue of the walls of the heart. Unlike skeletal muscles they are made of branched fibers joined by trasversal bands, called intercalated disks, which are responsible for a strongly anisotropic nature at a macroscopic level. Cardiac fibres, formerly believed to be a syncytium (i.e. a multinucleated cell), are actually made of separate cells, the myocytes [138]. The syncytial nature of the cardiac tissue allows to model the tissue as an electrically homogeneous medium. As explained further on, both skeletal and cardiac muscle fibers are composed of myofibrils. The myofibrils are composed of actin and myosin filaments, repeated in units called sarcomeres, which are the basic functional units of the muscle fiber. The sarcomeres are responsible for the striated appearance and form the fundamental contractile unit. Cardiac muscles contract in a similar manner to skeletal muscles, although with some important differences. The electrical stimulation triggers the release of Calcium from the cell's internal Calcium store, the sarcoplasmic reticulum. The rise in Calcium causes the cell's myofilaments to slide past each other in a process called excitation-contraction coupling.

Eventually, smooth muscles can be found in hollow viscera, airways, blood vessel walls. They are made of elongated ellipsoidal fibers with an almost random arrangement, which endow the tissue with a nearly isotropic behavior at contraction.



(a) skeletal muscle (400x) (b) Cardiac muscle (600x) (c) Smooth muscle (350x)

FIGURE 7.3: Longitudinal sections of the three different muscle tissue types. Image adapted from [139].

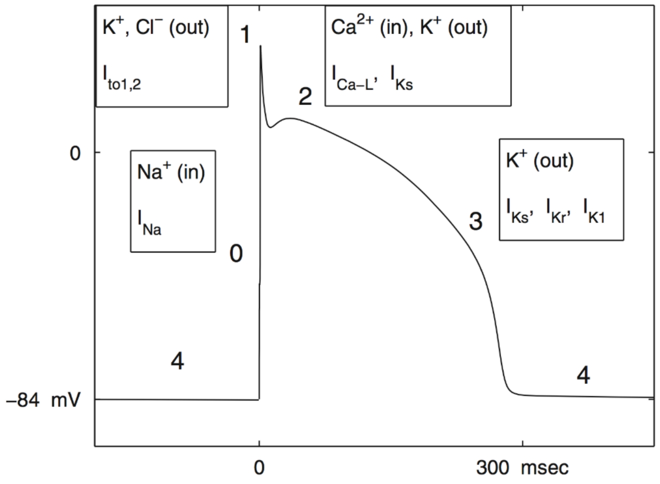


FIGURE 7.4: Evolution of the transmembrane potential in time after electrical stimulation. The numbers 0-4 correspond to the five phases of the action potential described in the present section. Figure taken from [140]

The myocytes in the cardiac tissue (cardiomyocytes) are enclosed in a lipid membrane (sarcolemma) that separate the external and internal region (a sketch of the cardiomyocyte components is provided in Fig. 7.5). The inner region contains sub-cellular organelles, sarcomeres and the sarcoplasmic reticulum immersed in the sarcoplasm. The outer region and the sarcoplasm contain a variety of ions; the flow of these ions through the sarcolemma happens via some channels (see Fig. 7.6), called ionic channels, where only the transit of specific particles is permitted. The change in concentration of the charged particles is the key for the electrophysiologic excitation of the cell. An insight of the cardyomyocites histology can be found in [141]. Due to their

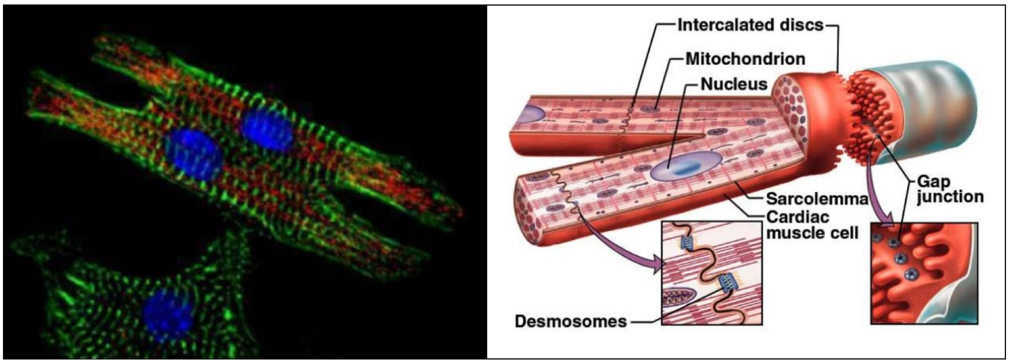


FIGURE 7.5: Schematic of the structure of a cardiomyocyte (right). Microscopy picture of a cardiomyocyte (left). Figure taken from `http://www.cardio-research.com/cardiomyocytes`

branched nature and interconnections, cardiomyocytes are electrically connected and the stimulus to contract can be transmitted to adjacent cells. The aforementioned intercalated discs are connective structures which include complex conductive proteins (desmosomes) and channels which allow the intra-cellular ion transit (gap junctions). The different densities of gap junctions in the longitudinal and transversal directions make the propagation of the electric stimulation anisotropic. The peculiar strong electrical interconnection of cardiac cells does not allow the regulation of the amount of activated cells, which makes the cardiac muscle activation not adjustable.

The Action Potential (AP), or transmembrane potential, denotes the variation of the difference between internal and external electric potential occurring during the electrical stimulation of the myocytes. The action potential can be subdivided in five phases [140], schematically reported in Fig. 7.4. The first stage consist of a transit of Na^+ ions through the appropriate sarcolemma channels, that makes the transmembrane potential raise from the resting (negative) value to the maximum value (positive). The Na^+ species transit takes a few nano-seconds, therefore it represents a nearly-discontinuous change in the action potential dynamics. Right after the depolarization, the transmembrane potential goes through a rapid decrease due to an outward flow of K^+ and Cl^- ions, occurring after the inactivation of the Na^+ channels. Phase 2 is characterized by an inward current linked to the transit of Ca_2^+, and an outward current caused by the transit of K^+. This balance keeps the potential almost constant in stage 2. Subsequently, the repolarization of the cell (phase 3) take place as a consequence of the closing of the Ca_2^+ channels. At this stage, the residual flow ok K^+ causes an outward current that makes the potential return to its resting value. The myocyte remains in the resting phase until the next electric stimulation. Although the shape of the action potential does not vary significantly, the duration and the amplitude of the action potential can change even within different regions of the same tissue. Among the several species regulating the transmembrane potential, the most important is the calcium ion Ca_2^+. The change in intracellular Ca_2^+ concentration regulates the contraction of the cardiomyocytes and influences the pacemaking [141]. As a matter of fact, all biophysically detailed models propose ion-flux mechanisms based on the Calcium dynamics.

In the present work a continuum model for the propagation of the action potential over a surface is employed, and a phenomenological description is accepted for the ionic currents. The process behind the generation of the action potential has been described for the cardiac tissue, but it does not represent a strict limitation to the applicability of the Mono- and Bidomain models. As long as a prosthetic device or soft actuators is built with cultured myocardial tissue of animal origin, these models are perfectly suitable. In recent works they have been employed in the simulation of gastric motility [142, 143] and jellyfish locomotion [144, 145].

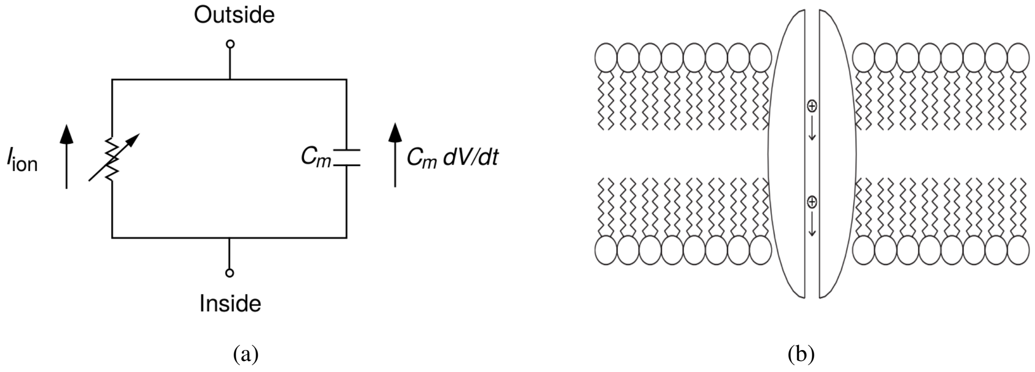


FIGURE 7.6: Lumped model of the cell membrane (a). Schematic representation of the ionic channels through the lipidic membrane of a myocyte (b) [146]

7.4 Action potential propagation: a reaction-diffusion model

The experimental measurements of Hodgkin and Huxley [147] on the propagation of an electrical signal along a squid giant axon led to a phenomenological model, which describes how the action potential is propagated in neurons. Their model, beyond the purpose of unveiling the action potential mechanism of the neurons of a primitive swimmer, set the groundwork for the investigation of the electrical activity of a wide variety of excitable tissues.

Since the cell membrane separates charges, it can be viewed as a capacitor. Thus, the simplest model which can be used to describe the membrane electrical activity is those of a perfect condenser, composed by a capacity in parallel with a resistance (see Fig. 7.6). By defining the capacitive current as $C_m\, dv/dt$, and assuming C_m to be constant and homogeneous over the membrane, the charge balance yields:

$$C_m \frac{dv}{dt} + i_i(v, w) = i_a\,, \tag{7.1}$$

being i_i and i_a the ionic current density and the total membrane current density, respectively. v represents the transmembrane potential and C_m is the membrane capacity per unit area. The evidence that the capacity current and the ionic current are in parallel was provided by Hodgkin and Huxley, which showed the similarity between the ionic currents measured with $dv/dt = 0$ and those calculated from the capacity term with $i_a = 0$ [147]. The number of resistances in parallel with the capacity depends on the number of ionic species, thus on the chemical complexity of the model. The circuit equation must be completed with a relation between the ionic current and the transmembrane potential, which provides a constitutive model of the membrane. The ionic current i_i must depend on the amount of open channels per unit area, conceptualized by the variable w. A further generalization of the problem can be obtained by introducing the concentrations of the species involved **c** as further variable:

$$\begin{cases} C_m \dfrac{dv}{dt} + i_i(v, \mathbf{w}, \mathbf{c}) = i_a \,, \\ i_i(v, \mathbf{w}, \mathbf{c}) = \displaystyle\sum_{k=1}^{N} G_k(v, \mathbf{c}) \prod_{j=1}^{M} w_j^{p_{jk}} (v - v_k(\mathbf{c})) + i_n(v, \mathbf{w}, \mathbf{c}) \,, \\ \dfrac{d\mathbf{w}}{dt} = \mathbf{F}(v, \mathbf{w}, \mathbf{c}) \,, \\ \dfrac{d\mathbf{c}}{dt} = \mathbf{G}(v, \mathbf{w}, \mathbf{c}) \,, \end{cases} \tag{7.2}$$

where the model functions **F** and **G** provide the time evolution of the gating variable array **w**. N and M represent the number of ionic channels and the number of ionic species described.
This is the generalized version of the Hodgkin-Huxley model, which provides the evolution of an electric pulse over an infinitely elongated, isolated neuron. In this model the dielectric losses in the membrane have been neglected; however, the error introduced is not thought to be large since the time course of the capacitive source is reasonably close to that calculated for a perfect condenser. From the mathematical perspective this is a system of coupled ordinary differential equations.

7.5 Homogenized continuum models for the action potential propagation

The Hodgkin-Huxley model was derived from experimental measurements over single cells. However, reproducing the macroscopic mechanics over a muscular tissue by resolving the neuromechanics of every single cell would be a prohibitive task. Instead, the macroscopic effects over a muscular tissue are modeled by taking the active domain as a continuum media. Several models have been presented in literature with respect to the action potential propagation over active tissues by bridging the conceptual gap between the neuron representation and the continuum model with different level of complexity [148]. A continuum approach relies on the hypothesis that the cells in the tissue are electrically independent one another, then a superposition of the cellular activity can be accounted to compose the macroscopic effects [148]. However, the cellular elements are interactive, since the intracellular spaces are functionally interconnected via side-to-side or end-to-end membrane junctions. Thus, in a macroscopic description, the macroscopic current density and potential are considered as averaged effect of a small subset of cellular bunches, then the interconnection effects are intended as secondary effect [149].
In the following paragraph the governing equations of the action potential model are obtained by applying some phenomenological observations on muscular tissues to general considerations of electromagnetism [146][149].

7.5.1 The bidomain model

The transmembrane effects are simulated by modelling the active tissue as a homogeneous membrane separating the intracellular and the extracellular region, which should be intended as the convoluted description for the intracellular and extracellular volumes. In other words, the individually coupled cells are replaced in the model with a syncytium. In this averaged sense, the tissue is intended as a two-phase medium, as if every point in space is composed of a certain fraction of intracellular space and a fraction of extracellular space, and any discrete quantity is intended as averaged over the volume. For instance, the discrete junctional resistances, along with the myocites conductivity, get combined and smeared over a volume of interest to create an equivalent conductivity [150]. Hence, the electromechanic response to the changes in ionic concentration is resolved over two overlapped domain Ω_i and Ω_e, with $\Omega_i = \Omega_e$, representing the inner and outer space of the myocites [149]. At each point of the domain, two different values of local potential are associated, the intracellular potential and the extracellular potential respectively.

We consider a volume Σ in the three-dimensional space $\mathbb{R}^3$ where the two surfaces Ω_i and Ω_e are enclosed; the subscripts $\{i, e\}$ denotes the intracellular and extracellular space. In this domain a conservative electric field can be defined as:

$$\mathbf{E} = -\nabla u \,. \tag{7.3}$$

being u the electric potential field in the domain Σ. Defined the current density $\mathbf{J}$ propagating over the membrane, one can state that:

$$\mathbf{J}_i = -\mathbf{D}_i \nabla u_i; \quad \mathbf{J}_e = -\mathbf{D}_e \nabla u_e \,. \tag{7.4}$$

where the potential u_i, u_e are locally defined over the surfaces Ω_i and Ω_e respectively, $\mathbf{D}_i$ and $\mathbf{D}_e$ are the internal and external conductivity tensors (symmetric, elliptic tensors), which account for the anisotropic propagation of the current over the two surfaces.

In the three-dimensional volume enclosing the membrane the charge conservation for the intracellular end extracellular sub-space lets:

$$\frac{\partial}{\partial t} \int_{\Sigma} q_{i,e} dV = - \int_{\partial\Sigma} \mathbf{J}_{i,e} \cdot \mathbf{n} \, dS + \int_{\Sigma} \phi_{i,e} \, dV \,, \tag{7.5}$$

where $\mathbf{n}$ is the outward directed unit vector normal on the control volume boundary $\partial\Sigma$, $q_{i,e}$ represents the local charge density over the subspace, and $\phi_{i,e}$ is the amount of charge that transits from the extracellular space in the intracellular space and vice-versa. In the intracellular conservation equation it appears as a generation term, but it must hold $\int_{\Sigma} \phi_i \, dV = - \int_{\Sigma} \phi_e \, dV$. Assumed the regularity of the argument functions in (7.5), and the topology restrictions on the domain, for the divergence theorem it holds:

$$\int_V \frac{\partial q_{i,e}}{\partial t} \, dV = \int_V (-\nabla \cdot \mathbf{J}_{i,e} + \phi_{i,e}) \, dV \,. \tag{7.6}$$

Since the choice of the control volume is arbitrary, the former relation must hold locally. Then, recalling (7.4), the local charge conservation can be expressed in the extra- and intracellular sub-domains as:

$$\begin{aligned} \frac{\partial q_i}{\partial t} &= -\nabla \cdot (\mathbf{D}_i \nabla u_i) + \phi_i \,, \\ \frac{\partial q_e}{\partial t} &= -\nabla \cdot (\mathbf{D}_e \nabla u_e) + \phi_e \,. \end{aligned} \tag{7.7}$$

The charges transition across the membrane must be related to the ionic membrane currents through the ionic gates:

$$\phi_i = -\chi i_i, \quad \phi_e = \chi i_e \,, \tag{7.8}$$

being χ the average surface-to-volume ratio of the tissue and i defined as current per unit area. The sign of the charges is considered as positive if going from the intracellular to the extracellular space. Given the reduced thickness of the myocites membrane, and the high density of ionic gates, the model can be completed with the assumption of no charge accumulation at local level [149]. Thus, with a local charge balance across the membrane:

$$q_i + q_e = 0\,. \tag{7.9}$$

By applying this relation to (7.7), it holds:

$$\nabla \cdot (\mathbf{D}_i \nabla u_i) + \nabla \cdot (\mathbf{D}_e \nabla u_e) = 0\,, \tag{7.10}$$

or

$$\nabla \cdot \mathbf{J}_{tot} = 0\,. \tag{7.11}$$

The transmembrane potential v, which represents the final unknown of the homogeneous problem, can be defined locally as the charge difference across the membrane thickness:

$$v = u_i - u_e = \frac{1}{2C_m}(q_i - q_e)\,. \tag{7.12}$$

Taking the derivative of Eq. (7.12) and exploiting the charge balance (7.9), it yields:

$$\begin{aligned} C_m \frac{\partial v}{\partial t} &= \frac{\partial q_i}{\partial t}\,, \\ -C_m \frac{\partial v}{\partial t} &= \frac{\partial q_e}{\partial t}\,. \end{aligned} \tag{7.13}$$

Once the transmembrane potential is inserted in the charge conservation law (7.7), the bidomain model can be finally stated by means of two coupled parabolic equations. The model is presented in closed form introducing two equations for the dynamic evolution of the recovery variable and for the ions concentration and one model for the ionic current activation:

$$\begin{cases} C_m \dfrac{\partial v}{\partial t} - \nabla \cdot (\mathbf{D}_i \nabla u_i) + \chi i_{ion}(v, \mathbf{w}, \mathbf{c}) = \chi i_i^a\,, \\ -C_m \dfrac{\partial v}{\partial t} - \nabla \cdot (\mathbf{D}_e \nabla u_e) - \chi i_{ion}(v, \mathbf{w}, \mathbf{c}) = -\chi i_e^a\,, \\ \dfrac{d\mathbf{w}}{dt} = \mathbf{F}(v, \mathbf{w}, \mathbf{c})\,, \\ \dfrac{d\mathbf{c}}{dt} = \mathbf{G}(v, \mathbf{w}, \mathbf{c})\,, \\ i_{ion} = \mathbf{H}(v, \mathbf{w}, \mathbf{c})\,. \end{cases} \tag{7.14}$$

A schematic of the bidomain model is providded in Fig. 7.7. The standard measurement units employed for this model are:

$$C_m\left[\frac{F}{m^3}\right], \quad \chi\left[\frac{1}{m}\right], \quad i\left[\frac{A}{m^2}\right], \quad v\,[V]\,, \quad w\,[-]\,, \quad D_{ij}\left[\frac{S}{m}\right]. \tag{7.15}$$

The gating variable w is dimensionless because it represents the portion of the membrane area covered by the open ionic gates. The ODE for $\mathbf{w}$ describes the evolution in time of the surface fraction of open gating channels in the membrane, and the ODE for $\mathbf{c}$ describes the evolution of the concentration of the ionic species. The system (7.14) must be solved in time for the unknown array $\{v,\ \mathbf{w},\ \mathbf{c}, I,\ u_{i,e}\}$.

The bidomain model entails the transmembrane potential propagating over the membrane as a response to the applied current $i^a_{i,e} = i^a_{i,e}(t, \mathbf{x})$, which is included in the problem (7.14) to model the current stimulus generated by the pacemaker cells enclosed in localized regions of the contractile tissue. From a modeling perspective the action potential evolves as a sharp wave front diffusing over the surface according to the conductivity tensors $\mathbf{D}_i$, $\mathbf{D}_e$.

The assumptions under which the bidomain model was formulated are summarized:

- The rate of change of the electric field and magnetic field over the surface is assumed to be negligible, then any transformation occurring over the tissue is considered quasi-static;
- The model is solved under the assumptions of conservation of the local charge.
- The action potential is assumed to exit the domain with minimal reflections, as if further tissue with similar properties surrounded the domain. Thus, homogeneous Neumann conditions are assigned at the membrane edges: $\mathbf{n}^T\mathbf{D}_i\nabla u_i = 0$, $\mathbf{n}^T\mathbf{D}_e\nabla u_e = 0$. Including the applied current impulse in the charge conservation, equation (7.10) must be corrected as:

  $$\nabla \cdot (\mathbf{D}_i \nabla u_i) + \nabla \cdot (\mathbf{D}_e \nabla u_e) = i^a_e - i^a_i\,. \tag{7.16}$$

  Integrating the former equation over the respective domains Ω_i, Ω_e, and applying the divergence theorem, from the Neumann conditions the following compatibility condition is given:

  $$\int_{\Omega_i} i^a_i\, dS = \int_{\Omega_e} i^a_e\, dS\,. \tag{7.17}$$

- The charges do not accumulate in any point of the sub-domains but a local charge transition from the intra and extracellular space is governed by the local conservation law (7.9);

The derivation of the bidomain model from the discrete unit cell model (represented in Fig. 7.6) has been proposed with mathematical formalism by means of a homogenization process [151][132].

Most of the continuum models are simplified by assuming just one recovery variable and uniform ionic concentration over the membrane. Since this model is derived from a microscopic description, it requires a detailed knowledge of the underlying microscopic mechanism, as the ionic concentration evolution in time and space. However, the present work is focused on the issues concerning the relations between the action potential propagation and the elastic response of the active tissue. Thus, only one ionic specie with uniform concentration is considered, being regarded as the chemical engine governing the tissue reaction to the current impulse. The simplified system reads:

$$\begin{cases} C_m \dfrac{\partial v}{\partial t} - \nabla \cdot (\mathbf{D}_i \nabla u_i) + \chi i_{ion}(v, w) = \chi i^a_i\,, \\ -C_m \dfrac{\partial v}{\partial t} - \nabla \cdot (\mathbf{D}_e \nabla u_e) - \chi i_{ion}(v, w) = -\chi i^a_e\,, \\ \dfrac{dw}{dt} = g_r(v, w)\,, \\ i_{ion} = h(v, w)\,. \end{cases} \tag{7.18}$$

The original problem unknown array is reduced to $\{v,\ w, i_{ion},\ u_{i,e}\}$

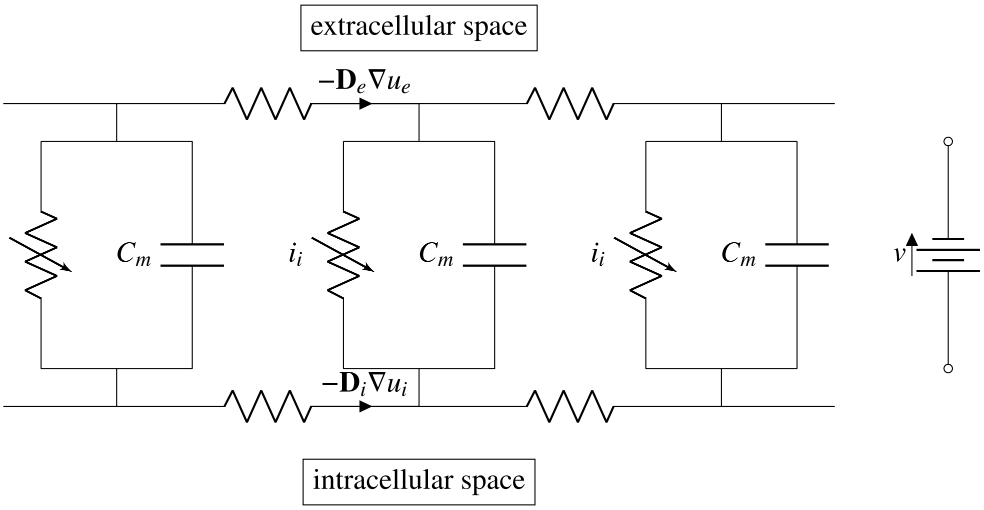


FIGURE 7.7: Circuit diagram of the bidomain model. The intracellular and extracellular spaces are represented by the external part of the circuit. Each block in the transmembrane space represents the model-dependent nonlinear current-voltage relation.

7.5.2 The monodomain model

A further simplification of the problem can be introduced by assuming that a intra- and extracellular media have the same conductivity anisotropy ratio.
At any point in the domain, the total current is $\mathbf{J}_t = \mathbf{J}_i + \mathbf{J}_e$, thus:

$$\mathbf{J}_t = -\mathbf{D}_i\nabla u_i - \mathbf{D}_e\nabla u_e\,. \tag{7.19}$$

By some algebraic passages, one can get:

$$\nabla u_i = (\mathbf{D}_i + \mathbf{D}_e)^{-1}\,(\mathbf{D}_e\nabla v - \mathbf{J}_t)\ . \tag{7.20}$$

After substitution of the latter relation in the first equation of the system (7.24), it yelds:

$$C_m\frac{\partial v}{\partial t} - \nabla\cdot\left(\mathbf{D}_i\,(\mathbf{D}_i + \mathbf{D}_e)^{-1}\,\mathbf{D}_e\nabla v\right) - \nabla\cdot\left(\mathbf{D}_i\,(\mathbf{D}_i + \mathbf{D}_e)^{-1}\,\mathbf{J}_t\right) + \chi i_{ion}(v,w) = \chi i_i^a \tag{7.21}$$

The term involving the total density current $\mathbf{J}_t$ is zero if the matrix $\mathbf{D}_i\,(\mathbf{D}_i + \mathbf{D}_e)^{-1}$ is proportional to a constant multiple of the identity matrix, i.e., if the two conductivity tensors are proportional $\mathbf{D}_i = \lambda\mathbf{D}_e$, with λ constant. Thus, the first PDE of the system (7.24) reduces to:

$$C_m\frac{\partial v}{\partial t} - \nabla\cdot(\mathbf{D}\nabla v) + \chi i_{ion}(v,w) = \chi i^a\,, \tag{7.22}$$

where:

$$\mathbf{D} = \mathbf{D}_i\,(\mathbf{D}_i + \mathbf{D}_e)^{-1}\,\mathbf{D}_e\,. \tag{7.23}$$

Repeating the same procedure with the second parabolic PDE, one obtains two linearly dependent equations. Thus, the problem is reduced to a monodomain system, where just the transmembrane potential needs to be defined locally. This represents a system made up by one reaction-diffusion parabolic equation, one first-order ordinary differential equation for the recovery variable and one algebraic equation for the ionic current model:

$$\begin{cases} C_m \dfrac{\partial v}{\partial t} - \nabla \cdot (\mathbf{D} \nabla v) + \chi i_{ion}(v, w) = \chi i^a \,, \\ \dfrac{dw}{dt} = g_r(v, w) \,, \\ i_{ion} = h(v, w) \,. \end{cases} \tag{7.24}$$

Eventually, the problem unknown array is further reduced to the scalar fields $\{v,\ w, i\}$. The monodomain model allows to achieve a satisfactory accuracy, sufficiently close to that of the bidomain model, especially under physiological conditions [152] even if the assumptions on the diffusivity tensors is not exactly fulfilled. However, some important physiological phenomena vanishes when equal anisotropy rates are assumed [146], especially in the simulation of pacing and fibrillation mechanisms on cardiac tissue. Furthermore, even with the support of experimental measurements, it is difficult to specify the conductivity values which can better approximate the actual potential propagation.

Beyond the mathematical and numerical simplification arising from the domain degeneration, the monodomain model still presents different numerical challenges. The reactive term represents a source of nonlinearity in the partial differential equation for the transmembrane potential.

The monodomain model provides a significant simplification in the memory storage and computing time with respect to the bidomain model. In fact, Skouibine et al. [153] reported a reduction in CPU requirements by two orders of magnitude when dealing with a fully implicit time-advancement scheme.

7.5.3 Restrictions and limitations of the mono- and bidomain models

The ”syncytial” nature of the tissue is a fundamental requisite for the application of the mono/bidomain formulation, since they are characterized by an electrically connected fibers.

Recalling the hypostheses presented in the preceding section, the validity of the homogenized syncytium model is assured deep in the presence of external fields that are nearly uniform and limited in strength. On the contrary, the derived model is not formally valid under strong or rapidly changing electrical fields [151].

The electrical components arrangement in the mono- and bidomain models allows a perturbation to ideally propagate at infinite speed. Actually, in mono- and bidomain simulations the velocity of the propagating front is connected to the dynamics of the recovery variable, thus the speed is not infinite. The models differs from the coaxial cable standard model by the absence of the inductance, which accounts for the relaxation time in the signal propagation. In standard electrophysiology models the inductances are neglected since their effect is considered negligible. These models also neglect the fact that neurons, skeletal muscle cells, and cardiomyocytes show typical resonance effects due to inductances [154]. The common conclusion that inductances are negligible may not be valid in the complex arrangement of cardiac tissue, where the inhomogeneities together with highly nonlinear reactions can lead to reentrant waves and chaotic behavior. In a recent work Rossi et al. [155] incorporated the axial inductances in the mono/bidomain model providing a hyperbolic version of the classical PDEs.

Along these lines, Gizzi et al. [156] extended the standard cable equation by including a nonlinear voltage dependence in the diffusion term. They also include thermo-electric effects in the reaction terms, finding that a nonlinear diffusion affects the repolarization process and facilitates the rise of unstable propagation effects at temperatures higher and lower than standard physiological temperatures.

7.6 Ionic current modelling

The evolution of the ionic current in time defines the reaction effect in the reaction-diffusion equation (7.2). A well-suited model must reproduce the changes in the ionic current induced by the migration of the ionic species. Generally, the more species are involved in the model, the more realistic the shape of the action potential. Nevertheless, the ionic current model is the source of nonlinearity in the mono- and bidomain equations, therefore, more sophisticated models, provide a significant increase in the computational cost (see section 7.6). Purely by way of example, the action potential corresponding to four models is illustrated (Fig. 7.8). This is obtained considering a one-dimensional domain, in which all the charge transported by the ionic current accumulates locally at the membrane and uniquely determines the transmembrane potential:

$$C_m \frac{dv}{dt} = -i_i + i_a \,. \tag{7.25}$$

Generally, the ionic current model must be chosen with respect to the specific tissue which one aims at describing, and to the affordable computational cost. Biophysically detailed models provide a sophisticated description of the ions dynamics, as well as a significant number of ionic currents and gating variables, in order to match the action potential shape measured in experiments. However, complicated models need a larger number of coefficients, that need to be fitted on experimental data.

In the present work we use the Aliev-Panfilov model [157], whose action potential is shown in Fig. 7.8 (a), since it is one of the simplest phenomenological models, but it provides sharp depolarization front the numerical method must to cope with. The model's equations read:

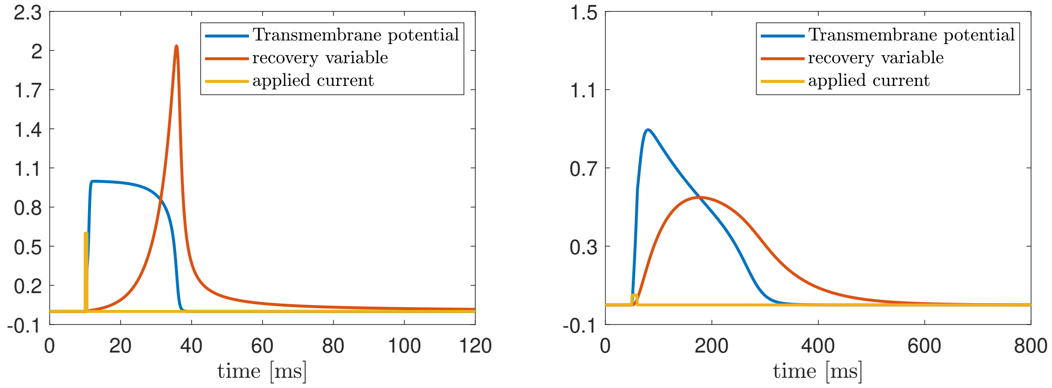
$$\begin{cases} C_m \dfrac{dv}{dt} = kv(v-a)(1-v) - vw + i_a \\ \dfrac{dw}{dt} = \left(\epsilon_0 + \dfrac{\mu_1 w}{\mu_2 + v}\right)(-w - kv(v-b-1)) \,, \end{cases} \tag{7.26}$$

where k, μ_1, μ_2 and ϵ_0 are the model parameters. The Aliev-Panfilov model's equations are presented with nondimensional variables. Such state variables can be rescaled by means of the following affine transformations, which provide the dimensional counterparts $\{\bar{v},\, \bar{w},\, \bar{i}_i\}$:

$$\begin{aligned} \bar{v} &= v_a\, v + v_r \,, \\ \bar{w} &= v_a\, v \,, \\ \bar{i}_i &= v_a\, i_a \,, \end{aligned} \tag{7.27}$$

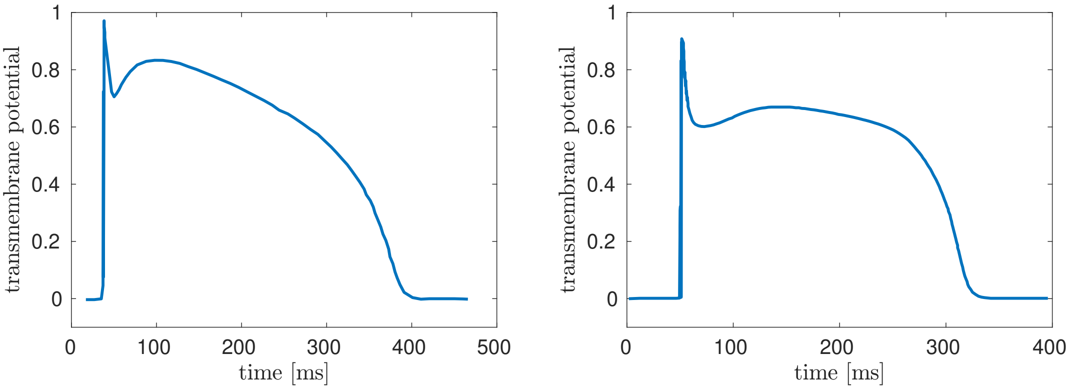
with $v_a = v_p - v_r$. Here, v_a, v_p and v_r represent the action potential amplitude, the peak value and the resting value, respectively.

Each model has its own constants and was tailored for specific applications. Widespread examples of chemically detailed models are given by the Bueno-Orovio model [158] (Fig. 7.9a), which was found to be suitable for the cardiac tissues, and by the Winslow model [159] (Fig. 7.9b), which was created for the dog cardiac cells.



(a) Aliev-Panfilov model (b) Rogers-McCulloch model

FIGURE 7.8: Field variable evolution for two phenomenological membrane models, in nondimensional units.



(a) Bueno-Orovio model (b) Winslow model

FIGURE 7.9: Nondimensional transmembrane potential evolution for two biophysically detailed membrane models.

Nowadays, ionic membrane models continue to increase in complexity since Hodgkin-Huxley models appeared; a modern, chemically detailed membrane model has upwards of a hundred state variables [160].

8 The monodomain model over general surfaces

The monodomain model, in conjunction with a phenomenological ionic current model, is chosen to solve the propagation of the action potential over a surface. The surface approximation if often adopted as it brings a significant reduction in the computational cost. However, it has been mathematically justified through asymptotic analysis [161], and adopted for realistic simulations of the electrical activity of the atria [162, 163].

Isogeometric Analysis has already been employed for mono- and bidomain models over a general surface, and several advantages have been recognized in the discretization of the differential operators. The high continuity of NURBS basis functions provides enhanced accuracy in the approximation of the steep depolarization front provided by these models [164]. Furthermore, in previous numerical experiments, IGA was found to capture the velocity of propagation of the front with a limited number of degrees of freedom and limited numerical dissipation [164, 163]. In the present chapter we present the temporal and spatial discretization of the monodomain model in a curvilinear coordinate framework.

8.1 Problem discretization in time

The monodomain model is considered as a parabolic partial differential equation coupled with a set of nonlinear ordinary differential equations. A significant complication for the solution of the excitation problem comes from the inherent multiscale nature of the reaction-diffusion process, both in space and time. The fundamental length scale in the problem is determined by the width of the wave front, which can be smaller than the geometric length scale by several orders of magnitude. From the temporal perspective, the time scale given by the diffusion of the wave front must be compared with the much faster reaction processes. Relevant diffusion effects occur in the time scale of hundreds of micro-seconds, whereas the gating kinetics of the channels in the membrane occur in the time scale of microseconds. In order to address these issues, some researchers have developed several approaches as the adaptive local refinement of the mesh, at the cost of an increased algorithmic and implementation complexity [165].

A fully implicit solution of the problem is often considered expensive in terms of time and memory storage, unless parallel computing techniques are utilized [165]. The fully implicit formulation generates large, sparse matrices, with size given by the product of the number of nodes and the number of dependent variables in the model, which must be inverted efficiently. Furthermore, the different time-scales involved in the phenomenon make the convergence process cumbersome if dedicated approaches are not employed. On the other hand, a backward approximation of all the time derivatives ensures the stability of the scheme and allows to use a time-step size dictated by the time scales rather than the time-step size prescribed for stability [166]. This must be accounted as a relevant feature in the choice of the time scheme, because the stability requirements might be severe in presence of a fine spatial resolution (which is often required).

A widespread approach for the bidomain model consists of the operator splitting (OS) method [146, 167]. The key benefit in the separation of the reaction and diffusion processes comes from the segregation of two spatially uncoupled problems, where dedicated time stepping coefficients can be implemented. Another choice is the fully explicit integration. If the reaction term is resolved explicitly, the source of nonlinearity is segregated from the diffusion operator, and therefore the nonlinear system (7.18) is reduced to a set of nonlinear ODEs and a linear PDE. A common choice is the Strang splitting scheme, which is second order accurate in time [124] [146]. The main drawback of a splitting technique is the loss of accuracy associated with the non-simultaneous solution of diffusive and reactive terms.

A further alternative is represented by the semi-implicit time stepping, where the diffusion term is treated implicitly and the reaction term explicitly. In first place, the segregation of the gating variable equations is provided by the explicit treatment of the reaction term. Likewise OS methods, a linear system must be resolved due to the diffusion term, thus a comparable computational effort is required. Moreover, care must be taken in the choice of the time step size due to the stability constraint. The semi-implicit scheme is often employed when dealing with simplified membrane models [168].

In the present work a semi-implicit time scheme is implemented, since it is believed to provide a satisfactory compromise between accuracy and stability [169][170]. The reaction term is discretized by an explicit second order Adams-Bashforth method and the diffusion term is discretized by an implicit Crank-Nicholson method. The present scheme is outlined for the resolving time step $n+1$, which refers to the discrete time $t^{n+1} = (k+1)\Delta t$, being k the discrete time counter, and $(\Delta t)^n = t^{n+1} - t^n = \Delta t$ the constant time step size defined over the integration period. At each discrete time interval the state variables are defined consistently: $v^n = v(t^n)$, $w^n = w(t^n)$, $i^n_{ion} = i_{ion}(t^n)$. The reaction-diffusion equation in (7.24), is discretized in time as:

$$C_m \frac{v^{n+1} - v^n}{\Delta t} - \frac{1}{2} \nabla \cdot \left[\mathbf{D} \nabla \left(v^{n+1} + v^n\right)\right] + \frac{3}{2} \chi i^n_{ion} - \frac{1}{2} \chi i^{n-1}_{ion} = \chi i^{n+1}_a , \tag{8.1}$$

which, rearranged for the time advancement, reads:

$$\frac{C_m}{\Delta t} v^{n+1} - \frac{1}{2} \nabla \cdot \left(\mathbf{D} \nabla v^{n+1}\right) = \frac{C_m}{\Delta t} v^n + \frac{1}{2} \nabla \cdot \left(\mathbf{D} \nabla v^n\right) + \frac{1}{2} \chi i^{n-1}_{ion} - \frac{3}{2} \chi i^n_{ion} + \chi i^{n+1}_a . \tag{8.2}$$

After the solution of equation (8.2), the computed value of transmembrane potential v^{n+1} is used to solve the gating variable equation at each node of the spatial discretization by means of a fourth order Runge-Kutta method:

$$\begin{aligned} w^{n+1} &= w^n + \Delta t \left(\frac{1}{6} k_1 + \frac{1}{3} k_2 + \frac{1}{3} k_3 + \frac{1}{6} k_4\right) , \\ k_1 &= g\left(t^n , w^n\right) , \\ k_2 &= g\left(t^n + \frac{1}{2} \Delta t , w^n + \frac{1}{2} k_1\right) , \\ k_3 &= g\left(t^n + \frac{1}{2} \Delta t , w^n + \frac{1}{2} k_2\right) , \\ k_4 &= g\left(t^n + \Delta t , w^n + k_3\right) , \end{aligned} \tag{8.3}$$

being g the gating variable function regulating the recovery varible dynamics.

8.2 Variational formulation and space approximation

A Finite Element approximation in space is preferred since it is a well-suited choice for solving PDEs on irregular geometries with Neumann boundary conditions. Furthermore, the smooth representation of the geometry and the high-continuity shape functions provided by the NURBS-based IGA, allow for an accurate approximation of the sharp action potential front with a reduced number of degrees of freedom with respect to the standard FEA.

The variational formulation and the discretization by a Galerkin approach are briefly reported in the present paragraph; for a rigorous mathematical formulation one should read [164]. In view of writing the variational formulation of the monodomain PDE, the suitable functional sub-spaces defined over the domain Ω are considered. It is assumed that these spaces fulfill the necessary topological requirement to guarantee the existence of the week solution to the monodomain problem with any of the membrane models illustrated in Section 7.6.

Multiplying Eq. (8.2) by the test function ψ and integrating over the whole domain Ω, one obtains:

$$\begin{aligned}\frac{C_m}{\Delta t}\int_\Omega v^{n+1}\,\psi\,dS - \frac{1}{2}\int_\Omega \nabla\cdot(\mathbf{D}\nabla v^{n+1})\,\psi\,dS = \frac{C_m}{\Delta t}\int_\Omega v^{n}\,\psi\,dS + \frac{1}{2}\int_\Omega \nabla\cdot(\mathbf{D}\nabla v^{n})\,\psi\,dS + \\ \frac{1}{2}\chi\int_\Omega i_{ion}^{n-1}\psi\,dS - \frac{3}{2}\chi\int_\Omega i_{ion}^{n}\psi\,dS + \chi\int_\Omega i_a^{n+1}\,\psi\,dS\,.\end{aligned} \tag{8.4}$$

By applying the Green's identity to the diffusion terms, it results:

$$\int_\Omega \nabla\cdot(\mathbf{D}\nabla v)\,\psi\,dS = -\int_\Omega \nabla\psi\cdot(\mathbf{D}\nabla v)\,dS + \underbrace{\int_{\partial\Omega}\mathbf{n}^T(\mathbf{D}\nabla v)\,\psi\,dl}_{=0}\,. \tag{8.5}$$

where the boundary integral vanishes because of the Neumann boundary conditions.

In view of reducing the dimensionality of the trial spaces, according to the IGA approach, the space spanned by the NURBS basis functions is considered as the finite dimensional space of the approximate solution of the problem (8.4). Thus, the transmembrane potential can be approximated as a linear combination of basis functions ϕ_j and discrete nodal values v_j,

$$v \approx v^h = \sum_{j=1}^{n}\phi_j v_j\,, \tag{8.6}$$

being n the number of basis functions defined at the nodal location. Substituting the approximate solution in (8.4), and taking advantage of the linearity of the operators, one can exploit the NURBS basis functions ϕ_i as test functions and finally get the IGA approximation of the PDE. For the i-th test function it reads:

$$\begin{aligned}\frac{C_m}{\Delta t}\sum_{j=1}^{n} v_j^{n+1}\int_\Omega \phi_j\,\phi_i\,dS + \frac{1}{2}\sum_{j=1}^{n} v_j^{n+1}\int_\Omega \nabla\phi_i\cdot D_i\nabla\phi_j\,dS = \frac{C_m}{\Delta t}\sum_{j=1}^{n} v_j^{n}\int_\Omega \phi_j\,\phi_i\,dS - \\ \frac{1}{2}\sum_{j=1}^{n} v_j^{n}\int_\Omega \nabla\phi_i\cdot D_i\nabla\phi_j\,dS + \frac{1}{2}\chi\int_\Omega i_{ion}^{n-1}\phi_i\,dS - \frac{3}{2}\chi\int_\Omega i_{ion}^{n}\phi_i\,dS + \chi\int_\Omega i_a^{n+1}\,\phi_i\,dS\,.\end{aligned} \tag{8.7}$$

In matrix notation a system of linear equations is given in the form:

$$\mathbf{A}\,\mathbf{v}^{n+1} = \mathbf{b}^n\,, \tag{8.8}$$

with

$$\mathbf{A} = \frac{C_m}{\Delta t}\mathbf{M} + \frac{1}{2}\mathbf{K}, \tag{8.9}$$

$$\mathbf{b}^n = \left(\frac{C_m}{\Delta t}\mathbf{M} - \frac{1}{2}\mathbf{K}\right)\mathbf{v}^n + \frac{1}{2}\chi\,\mathbf{i}_{ion}^{n-1} - \frac{3}{2}\chi\,\mathbf{i}_{ion}^{n} + \chi\mathbf{i}_a^{n+1}\,. \tag{8.10}$$

In element notation, the standard matrices in the previous equation are:

$$[M]_{ij} = \int_\Omega \phi_i\phi_j\, dS\,, \tag{8.11}$$

$$[K]_{ij} = \int_\Omega \nabla\phi_i \cdot D_i\nabla\phi_j\, dS\,, \tag{8.12}$$

$$[i_{ion}]_i = \int_\Omega i_{ion}\phi_i\, dS\,. \tag{8.13}$$

It is worth noting that the matrix **A** must only be assembled once, since its components do not depend on time. On the other hand, in case of deforming domain, the stiffness matrix must be recomputed at each time step to account for the changes in the spatial conductivity coefficients. Hence, the only computationally demanding task in the linear system assembly is in the numerical integration of the ionic current field $\mathbf{i}_{ion}^n$.
The fundamental steps of the algorithm are resumed for the generic time step n:

1. Integrate the Ionic current at previous time step $\mathbf{i}_{ion}^n$;
2. Assemble the right-hand-side of the linear system (8.8);
3. Solve for $\mathbf{v}^{n+1}$;
4. Solve the gating variable equation (8.3) at the node locations;
5. Update the state variables for the next time step:

$$\mathbf{v}^n = \mathbf{v}^{n+1}, \quad \mathbf{w}^n = \mathbf{w}^{n+1}, \quad \mathbf{i}_{ion}^{n-1} = \mathbf{I}_{ion}^n; \tag{8.14}$$

Care must be taken in the spatial discretization of the domain due to the aforementioned multiscale nature of the problem. Specifically, the spatial accuracy is fundamentally dependent on the characteristic width of the wave front, and the maximum element size must be small enough to capture such a scale [171]. Since the shape of the action potential depends on the membrane model as well as on the membrane characteristics, the appropriate spatial resolution can only be found by numerical investigation.

8.2.1 Conductivity tensor on surfaces

The only spatial derivative in the present monodomain formulation lies in the diffusion term. The weak form of the reaction-diffusion equation requires the computation of the gradient of the shape functions (8.12). In this section we present a description of the diffusion term in local curvilinear coordinates, with the aim of providing a flexible implementation of the diffusion term on arbitrary surfaces in the three-dimensional space.

For each basis function the argument of the diffusion term yields in index notation:

$$\frac{\partial \phi}{\partial \theta^{\alpha}} \mathring{\mathbf{a}}^{\alpha} \cdot D^{\gamma\beta} \left(\mathring{\mathbf{a}}_{\gamma} \otimes \mathring{\mathbf{a}}_{\beta} \right) \cdot \frac{\partial \phi}{\partial \theta^{\beta}} \mathring{\mathbf{a}}^{\beta} = \frac{\partial \phi}{\partial \theta^{\alpha}} \frac{\partial \phi}{\partial \theta^{\beta}} D^{\alpha\beta} , \tag{8.15}$$

where $(\partial \phi / \partial \theta^{\alpha}) \, \mathbf{a}^{\alpha}$ is the covariant derivative of a scalar [172]. Assuming that the in-plane conductivity coefficients $D_{ca}^{\alpha\beta}$ are provided as input on a global Cartesian frame of reference $\{\mathbf{e}_1, \mathbf{e}_2\}$, the conductivity tensor coefficients on curvilinear bases $D_{cu}^{\gamma\delta}$ must be computed by a basis transformation in order to account for the in-plane deformations of the domain.

$$\begin{aligned} D_{cu}^{\alpha\beta} \left(\mathring{\mathbf{a}}_{\alpha} \otimes \mathring{\mathbf{a}}_{\beta} \right) &= D_{ca}^{\alpha\beta} \left(\mathbf{e}_{\alpha} \otimes \mathbf{e}_{\beta} \right) , \\ D_{cu}^{\gamma\delta} &= D_{ca}^{\alpha\beta} \left(\mathring{\mathbf{a}}^{\gamma} \cdot \mathbf{e}_{\alpha} \right) \left(\mathbf{e}_{\beta} \cdot \mathring{\mathbf{a}}^{\delta} \right) , \\ D_{cu}^{\gamma\delta} &= D_{ca}^{\alpha\beta} \left(\mathring{\mathbf{a}}^{\gamma} \cdot \mathbf{e}_{\alpha} \right) \left(\mathbf{e}_{\beta} \cdot \mathring{\mathbf{a}}^{\delta} \right) . \end{aligned} \tag{8.16}$$

One should notice that we defined two base vectors for the conductivity of a surface in the three-dimensional space, thus we inherently assume that the local out-of-plane distortions in the surface do not affect the conduction properties. This can be considered a congruent assumption by the moment that the complex fiber network which characterizes the tissue conductivity is homogenized in space and approximated by a surface.

8.2.2 Considerations on the scheme stability

The explicit treatment of the reaction term imposes a restriction on the time-step size for stability reasons. The numerical analysis in [170] proved that the stability limit of semi-implicit methods applied to the monodomain model coincide with the stability limit applied solely to the ionic models. Thus, the stiffness of the most complex ionic current models affects the stability of the scheme. In this scenario, the stability of the scheme is estimated by the local CFL condition:

$$\text{CFL} = CV \frac{\Delta t}{h} \leq CFL_{lim} , \tag{8.17}$$

where CV represents the local velocity of the propagating front, and h is the local element size. However, this very general expression of the stability threshold is unpractical because the computation of the local wave front in a sufficiently general framework is a demanding task, and it is a current avenue of research both in computational and experimental practice.
The stability limit can be estimated prior to the execution by exploiting the expression obtained in [173] for the forward-Euler method:

$$\text{CFL} = \frac{2(\sigma_a + \sigma_b)}{C_m} \frac{\Delta t}{h^2} . \tag{8.18}$$

This can provide a rough estimate of the stability limit, which in the case of semi-implicit methods can be highly conservative. However, as proved by different authors [170][169], the stability of semi-implicit methods does not really depend only on the spatial grid size, unlike fully explicit approaches.

8.3 Ionic current interpolation

The ionic current physically describes the current flux induced by the ion migration across the gates in the membrane. The ionic current is determined by the transmembrane potential across the membrane as well as by the gating variable evolution, which is governed by a ODE, and coupled back to the PDE through the reaction term (7.18). The determination of the ionic current and the gating variable plays a crucial role in the accurate resolution of the mono- and bidomain models.

Within the time scheme presented in section 8.1, the ionic current array has to be computed explicitly, therefore it is included in the right-hand-side of Eq. (8.8). In any traditional implementation the ionic current must be integrated over the elements, and therefore the local value of ionic current is required at the quadrature points of each element.

$$\mathbf{i}_i = \sum_{j=1}^{n} \int_{\Omega_e} [i_i(v, w)]_j \phi_k d\Omega. \tag{8.19}$$

A variety of strategies has been presented in the literature for the computation of the latter integral in a Finite Element framework. The transmembrane potential is defined at the nodes, whereas the discrete values of ionic current and gating variable can be defined either at the nodes or at the quadrature points. In the latter approximation, the membrane model needs to be evaluated at each quadrature points of the FE mesh. Even in the simple case of single ion specie model, this requires the solution of a nonlinear ODE and the explicit evaluation of the ionic current at far more points than the number of nodes. This often results in a prohibitively expensive approach, especially for biophysically detailed models. A widespread alternative consists of computing the ionic currents and the gating variables at the nodes, and interpolating the solution at the quadrature points. This method allows to solve the nonlinear ODEs for the gating variables only at the nodes, but is found to be less accurate than the former [148]. Both approaches are addressed within the present IGA implementation.

The interpolation at the interior of the element, is thus needed for the ionic current integration, and can be performed via “Ionic Current Interpolation” (ICI) or “State Variable Interpolation” (SVI). In the ICI approach, the ionic current is interpolated from its nodal values on the quadrature point, using the same basis functions as for the transmembrane potential:

$$\left(i_{ion}^{ICI}\right)_k = \sum_{j=1}^{n_e} i_i(\mathbf{w}_j, v_j)\, \phi_j. \tag{8.20}$$

at the k^{th} quadrature point. In the latter approach (SVI), the state variables $(\mathbf{w}, v)$ are prior interpolated at the element interior to evaluate the ionic current:

$$\left(i_{ion}^{SVI}\right)_k = i_i\left(\sum_{j=1}^{n_e} \mathbf{w}_j\, \phi_j, \sum_{j=1}^{n_e} v_j\, \phi_j\right). \tag{8.21}$$

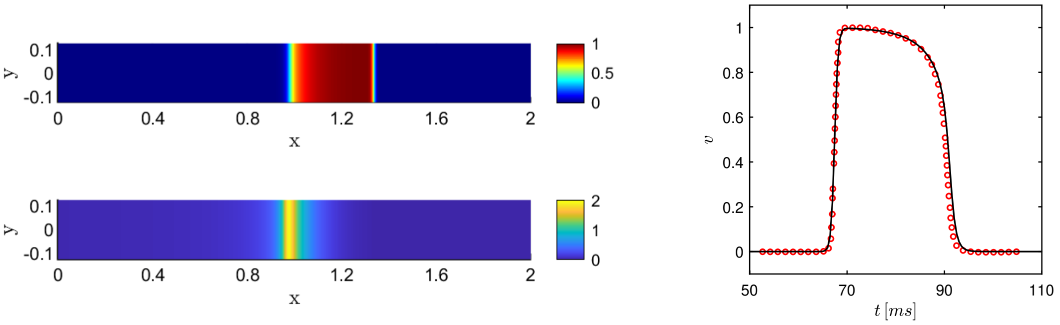
The SVI technique is the most accurate on a peer degrees of freedom basis, but it is associated with a high computational cost (especially with detailed membrane models), whereas the ICI is relatively less expensive but less accurate according to literature [174, 171]. The SVI needs larger memory storage than the ICI. However, considering that i_{ion} is usually a sharp function in space, its enhanced accuracy has proven to be fundamental in the solution of the problem [174]. On the other hand, the less accurate ICI has been successfully coupled with computationally cheap techniques as matrix lumping [171] for the assembly of the right-hand-side of Eq. (8.8) [174]. The choices of the time-advancement scheme and the numerical approach for the ionic current are strongly related in terms of numerical efficiency and method accuracy. In the present implementation, a state variable interpolation is employed in conjunction with a semi-implicit time scheme, because a phenomenological membrane model is adopted.

8.4 Verification and numerical experiments

The present section is devoted to the verification of the proposed monodomain implementation against well-established numerical tests.

8.4.1 Planar wave propagation over a rectangular slab

The planar propagation over a flat rectangular slab is simulated to verify the implementation of the proposed algorithm in terms of accuracy of the approximated conduction velocity and method convergence rate. In this context we compare the method convergence rate with both the ionic current interpolation strategies presented in section 8.3. We consider a computational domain of size $2 \times 0.2\,cm^2$. The tissue is assumed to have uniform capacity $C_m = 1.0\,mF/cm^3$, uniform surface-to-volume ratio $\chi = 1.0\,cm^{-1}$, isotropic conductivity $D_{ISO} = 10^{-4}\,S/cm$, and the ionic current is modeled by a normalized Aliev-Panfilov model, whose parameters are: $k = 8.0$, $a = 0.15$, $\varepsilon_0 = 2.0 \times 10^{-3}$, $\mu_1 = 0.2$, and $\mu_2 = 0.3$. Null initial values for dimensionless transmembrane potential v and recovery variables w are assumed. The stimulus for the depolarization is applied at the discrete level by forcing the transmembrane potential v associated to the outermost left control points to $v = 1.0$ from the initial instant up to $t = 0.5\,ms$, allowing the potential fronts to travel towards the right side of the slab. Such numerical stimulus mimic the applied current in Eq. (7.24). The same test-case was proposed by [164] with a different time-stepping scheme. The field variables v, and w are dimensionless; the dimensional counterpart of v can be recovered as $v = V_{min} + (V_{max} - V_{min})v$, with $V_{min} = -80.0\,mV$ and $V_{max} = 20.0\,mV$. In first instance the evolution in time of the transmembrane potential was compared (Fig. 8.2 - right) with the profile shown in [164]. The level of agreement between the numerical solutions suggests that



(a) (b)

FIGURE 8.1: (a) Contours of the nondimensional transmembrane potential v (top panel) and the recovery variable w (bottom panel) at time $t = 90\,[ms]$, by using a mesh size $h = 0.005\,cm$ and quadratic basis functions. (b) Time evolution of the dimensionless action potential v superposed with the profile in [164] with red dots.

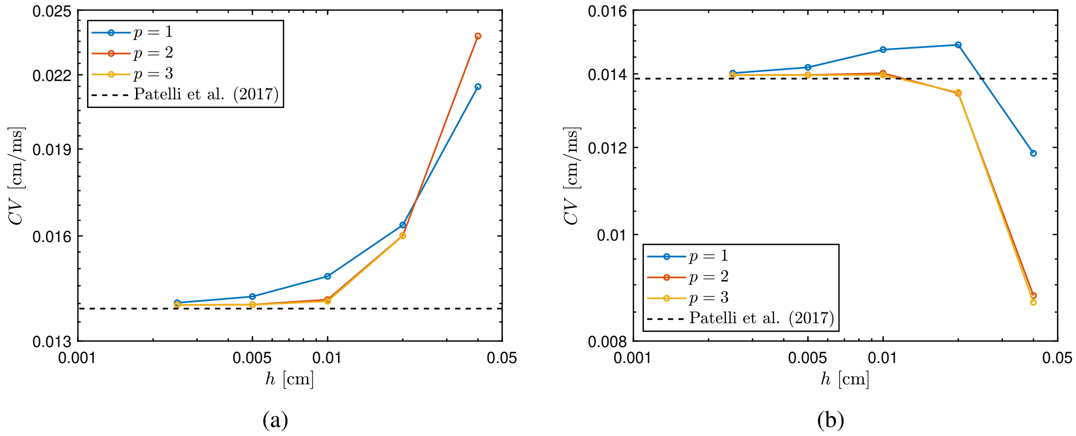


FIGURE 8.2: Convergence plot of the conduction velocity CV under h-refinement at different basis functions orders; ionic current integration by SVI approach (a), and ICI approach (b).

Case	$\Delta t\,[ms]$	$h\,[cm]$
1	0.006	0.04
2	0.003	0.02
3	0.0015	0.01
4	0.00075	0.005
5	0.000375	0.0025

TABLE 8.1: Summary of the computational setting for the refinement study performed on the planar propagation case.

our implementation is able to reproduce the shape of the action potential and it provides results consistent with those from other state-of-art solvers.

We exploit the planar propagation test to investigate the convergence of the conduction velocity under discretization refinement. Since a semi-implicit scheme is employed, we perform a refinement study at constant CFL –CFL = 0.02– by halving both the size of the mesh element and the size of the time step. To this extent we use the definition of CFL (8.18):

$$\mathrm{CFL} = \frac{2\,D_{ISO}}{C_m}\frac{\Delta t}{h}\,. \tag{8.22}$$

The cases investigated are resumed in Table 8.1. The mean front velocity CV is computed at discrete level by the difference in the position of the point at $v = 0.5$ (which is interpolated consistently with the order of accuracy of the scheme) between two time instants. The value obtained for the front velocity $CV = 0.01390\,cm/ms$ matches the value $CV^* = 0.01386\,cm/ms$ reported by Patelli et al. [164]. We observe that, with a SVI approach, the front velocity simulated with the coarse meshes is overestimated, whereas with a ICI approach the front velocity is underestimated. A similar trend has been highlighted by Patelli [164] under h-refinement only. By SVI, the value of the front velocity decreases monotonically with the refinement for all the basis functions employed. On the other hand, under ICI a slightly higher convergence rate is observed for the present test, although the SVI approach was expected to provide a better theoretical accuracy. The rate of convergence of the test variables is higher for high order basis functions $p = 2$, $p = 3$, but the most significant error reduction was found in the passage from $p = 1$ to $p = 2$. Thus, the p-refinement can be retained a valuable strategy to improve the approximation of the propagation speed with a limited number of degrees-of-freedom.

The suitability of IGA, and the more general paradigm of Galerkin approximating spaces with high smoothness, to advection-diffusion problems with strong gradients has been proved extensively in literature [175]. The propagation of the depolarization front in the monodomain model provides an example of time-dependent problem with very sharp gradients which can be better approximated by high order basis functions. With the aim of investigating this feature in the present context, the profiles of the action potential are plotted for various levels of refinement in Fig. 8.3 and Fig. 8.4. We notice that, on the coarsest cases, i.e. case 1 and 2, the SVI technique provides amplified spurious oscillations when increasing the order of the basis functions, compared to the ICI approach. At the control points the oscillations of the transmembrane potential are avoided thanks to the *variation diminishing* property [3] of NURBS-based IGA. We assert that this property is eluded in the SVI case on the coarsest meshes, because of the ionic current interpolation technique. As described in section 8.3, in the SVI approach the action potential v is interpolated at the element interior to evaluate the ionic current. At the quadrature points, the solution v does not benefit of the variation diminishing property, thus, the reaction term suffers from the oscillations which characterizes high degree polynomials in presence of discontinuities. A similar behavior was observed in [174] when comparing different ionic current interpolation techniques in a classic finite element framework. It is worth noting that in our simulations the ICI approach provides a stable results even in the case with the coarsest mesh and cubic basis function.

8.4.2 Spiral wave reentry

As a second example we consider the approximation of spiral waves with the Aliev-Panfilov model. Spiral wave reentry, and more generally wave meandering, is often implicated as one of the underlying causes responsible for ventricular tachycardia [176], cardiac arrhythmia, and other pathological states of the heart which lead to an irregular contraction of the cardiac muscles [177]. The disturbances responsible for meandering were found to be generated through an interaction between an unstable action potential trailing edge near the core and perturbations existing in the region immediately behind this edge [178].

The objective of this study is the replication of a self-sustained spiral wave able to maintain its coherence. We consider for the Aliev–Panfilov model the same membrane constants already used in the previous example. A domain of extension $1 \times 1\, cm^2$ is considered. The simulation is run at constant time-step size $\Delta t = 0.005\, ms$ on a domain discretized by 160×160 quadratic elements. The triggering stimulus is provided in the wake of a linearly propagating pulse such that the refractory region effectively generates a spiral wave, as in [164]. The linearly propagating front was generated as in the previous example whereas the second stimulus was obtained by applying a current $i = 2.0\, mA/cm^2$ on the region defined by the NURBS coordinates $[0.49, 0.51] \times [0, 0.5]$ between $t = 64.0\, ms$ ant $t = 64.5\, ms$.

In Fig. 8.5 we report the contours of the field variables v and w at the instants of triggering and propagation of the spiral wave, yielding qualitatively comparable results with what observed in literature [164].

In addition, we observed the trajectory of the spiral wave tip, tracked as the coordinates of the point laying on the the isoline of $v = 0.5$ with maximum absolute curvature value. The smoothness of the trajectory plotted in Fig. 8.6 is comparable with what observed by Patelli [164] on a mesh with equal base function order but larger element size. The spiral wave test provides a confirmation of the effectiveness of the domain boundary treatment, since the curved front is

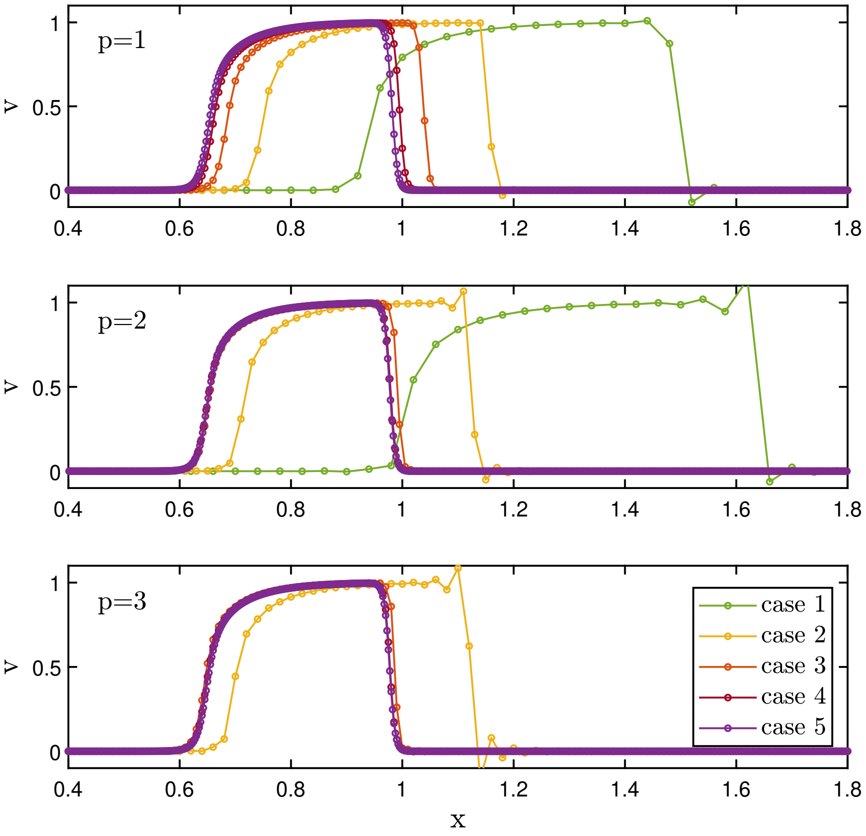


FIGURE 8.3: Action potential distribution on control points at $t = 70\,ms$ for each case in Table 8.1, where ionic currents are integrated by the SVI approach.

observed to exit the domain with minimal distortion. In second instance the test proved that the numerical method can handle complex stable pattern with a limited number of degrees of freedom.

8.4.3 Verification of the diffusion term implementation

The diffusion term of the monodomain model is discretized by an Isogeometric procedure with a local curvilinear description of the diffusion operator. The advantages of this approach are illustrated and discussed in the present section.
The implementation of the diffusion term is verified by comparing the solution on a distorted mesh with that from a standard NURBS-based implementation where the bi-dimensional gradient operator is locally applied to the basis function and the classical *strain* matrices are assembled by analogy with the linear elasticity problem on a two-dimensional domain. It is worth noting that this comparison is performed on planar geometries because in the present method the conduction tensor is not affected by local out-of-plane distortions.
In first instance the planar propagation over a rectangular slab is repeated over a mesh with dimensions $2/5\pi \times 1/5\pi\,\mathrm{cm}$ where sinusoidal distortions are introduced on the vertical element edges (Fig. 8.7). The same computational and physical parameters of the first example are chosen. On such a mesh, where the first base vector $\mathbf{a}_1$ can be chosen as parallel to the first Cartesian

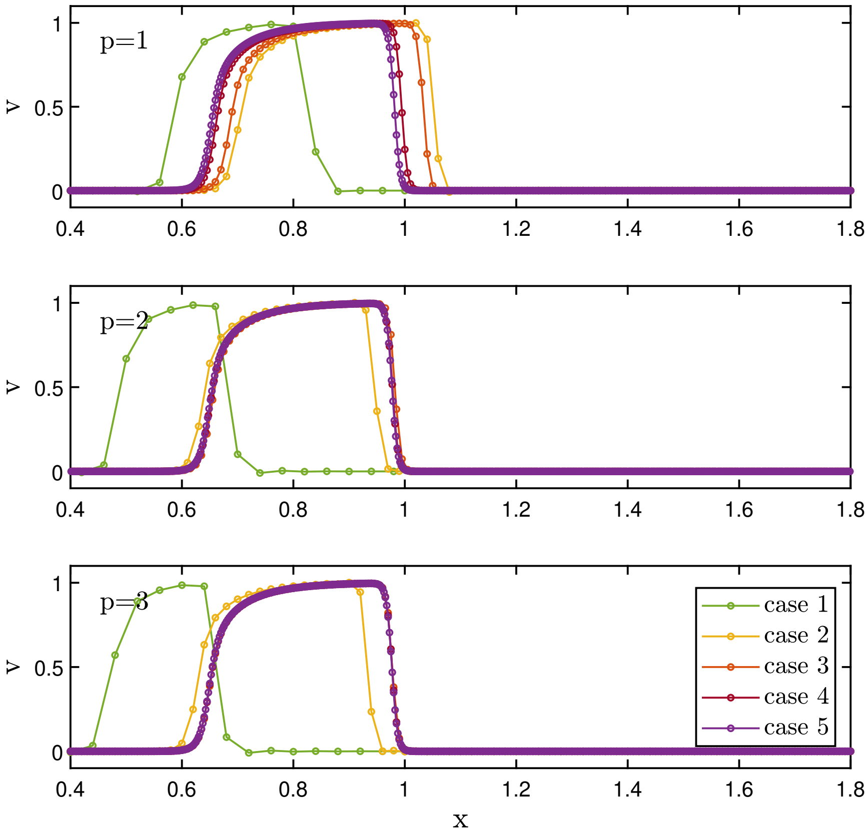


FIGURE 8.4: Action potential distribution on control points at $t = 70\,ms$ for each case in Table 8.1., where ionic currents are integrated by the ICI approach.

base vector $\mathbf{e}_1$, the loops required to perform the base transformations in (8.16) can be suppressed by means of the following considerations, with reduced computational expense. Exploiting the relations between covariant and contravariant vectors, the base vectors of the Cartesian bases can be expressed as:

$$\mathbf{e}_1 = \frac{\mathbf{a}_1}{\sqrt{a_{11}}}, \quad \mathbf{e}_2 = \frac{\mathbf{a}^2}{\sqrt{a^{22}}}, \tag{8.23}$$

where:

$$a^{ij} = \mathbf{a}^i \cdot \mathbf{a}^j, \quad a_{ij} = \mathbf{a}_i \cdot \mathbf{a}_j\,. \tag{8.24}$$

Then, substituting these definitions in Eq. (8.16), the contravariant coefficients of the conductivity tensor in curvilinear coordinates can be obtained:

$$\begin{aligned}
D^{11}_{cu} &= D^{11}_{ca}\frac{1}{a_{11}} + 2\,D^{12}_{ca}\frac{a^{12}}{\sqrt{a_{11}a^{22}}} + D^{22}_{ca}\frac{\left(a^{12}\right)^2}{a^{22}}\,,\\
D^{12}_{cu} &= D^{21}_{cu} = D^{12}_{ca}\sqrt{\frac{a^{22}}{a_{11}}} + D^{22}_{ca}a^{12}\,,\\
D^{22}_{cu} &= D^{22}_{ca}a^{22}\,,
\end{aligned} \tag{8.25}$$

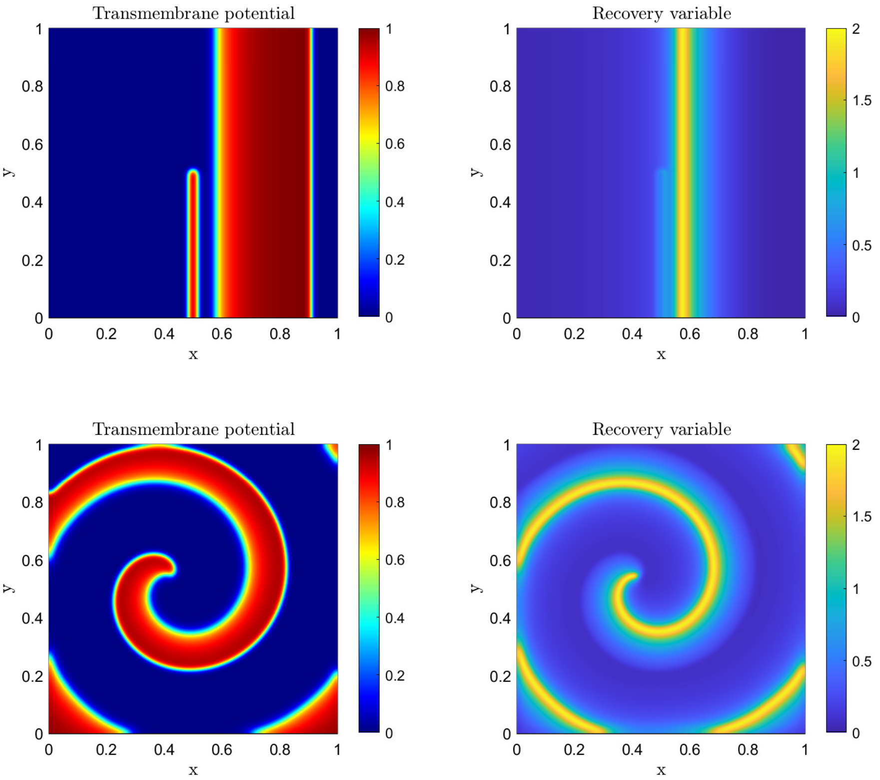


FIGURE 8.5: Contours of the field variables v (left) and w (right) for the spiral wave test, at the time instants $t = 65.0\,ms$ (top) and $t = 150.0\,ms$ (bottom).

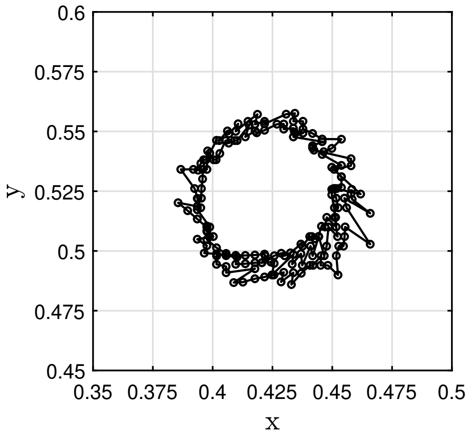
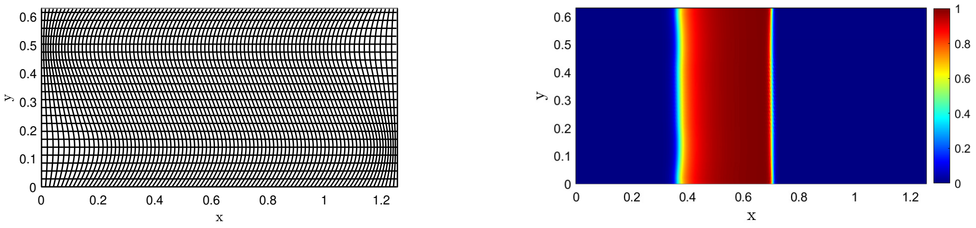


FIGURE 8.6: Trajectory of the spiral waves' tips over the last three rotation of the spiral. The corresponding mesh has 160 × 160 quadratic IGA elements.

under the hypothesis of symmetric conductivity tensor.

The contours in Fig. (8.7) exhibit the expected shape of the depolarization-repolarization wave despite the high degree of misalignment of the mesh with the propagating front. IGA has been proved to have enhanced performance on highly distorted meshes with respect to classic Finite



(a) (b)

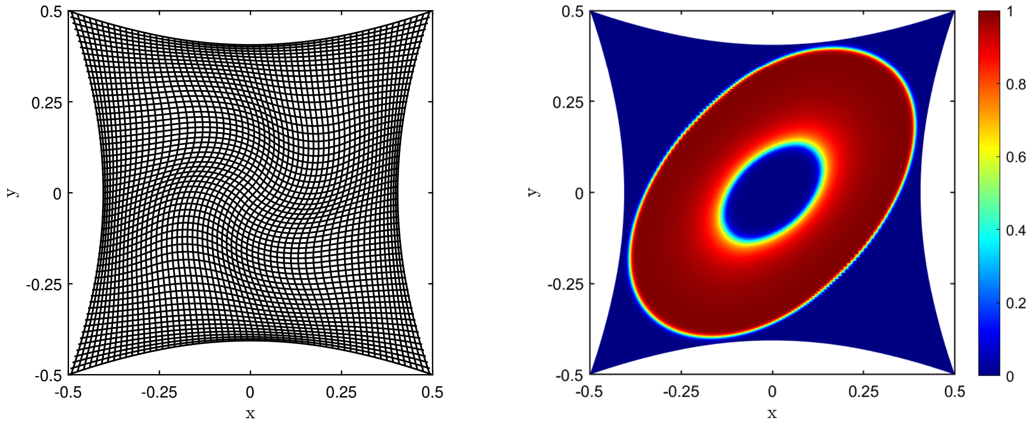
Figure 8.7: Computational grid for the test of planar propagation on an highly distorted mesh (a); one knot line every two is plotted for sake of clarity. Corresponding transmembrane potential contours (b) at $t = 50.0\,ms$.

Elements Analysis [3]. A quantitative comparison between the two implementations is provided in Fig. 8.9, where the Root Mean Square (RMS) of the difference in the maximum action potential value (8.26) and on its position on control points (8.27) are plotted against time. Numerically identical values prove the effectiveness of the implementation on curvilinear coordinates.

$$\tilde{v} = \frac{v_{curv} - v_{plate}}{v_{plate}}, \tag{8.26}$$

$$\tilde{x} = \frac{x^{cp}_{curv} - x^{cp}_{plate}}{x^{cp}_{plate}}. \tag{8.27}$$

A further test is proposed to verify the implementation of Eq. (8.16) when no base vectors alignment is present. To this extent we study the propagation of the action potential over a mesh with rotational distortions (Fig. 8.8). The simulation is initialized to null filed variables and a peacemaking stimulus of $2.0\,\text{mA/cm}^2$ is generated within a circular area of radius $R = 0.02\,\text{cm}$ for $t = 0.5\,\text{ms}$. In this case an anisotropic conduction tensor is imposed with $D^{11}_{ca} = D^{22}_{ca} = 10^{-5}\,\text{S/cm}$, $D^{12}_{ca} = D^{21}_{ca} = 5 \times 10^{-5}\,\text{S/cm}$. The comparison between the curvilinear and the 2D implementation provided a numerically coincident solution.



(a) (b)

FIGURE 8.8: Computational grid for the test of propagation on a mesh with rotational distortions (a); one knot line every two is plotted for sake of clarity. Corresponding transmembrane potential contours (b) at $t = 33.5\,ms$.

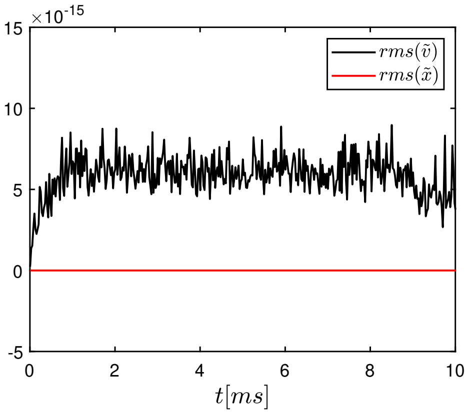


FIGURE 8.9: Statistical error indicators (defined in Eq. (8.26) and Eq. (8.27)) for the action potential propagation over rectangular domain.

9 Elecromechanical activation of a thin tissue by the active strain approach

A change in the configuration of a muscular tissue arising from an electrophysiologic stimulus requires the knowledge of the mechanical effect deriving from the sarcomere contraction. A common practice in computational electrophysiology, is the approximation of the elastic response of the tissue with a continuum model, based on the concept of active strain or active stress.

In the present chapter the active strain approach is introduced, and it is embedded in the weak formulation for KL shells presented in chapter 3. To the knowledge of the author this represents the first implementation of the active strain approach in a shell framework. Furthermore, all quantities needed for the solution of the mechanical problem are formulated in local curvilinear coordinates, and only later the shell hypotheses are enforced. Thus, the framework presented in this chapter is very general as it provides the expressions needed for a Lagrangian-active strain approach, applicable to three-dimensional solids.

9.1 The active strain approach

The effect of the microscopic mechanisms driving the tissue activation is embedded into the continuum approach by means of the well-established active strain approach [23], [179]. This approach entails the multiplicative decomposition of the deformation gradient tensor into an elastic (passive) part $\mathbf{F}_e$ and an active part $\mathbf{F}_a$ in the following form:

$$\mathbf{F} = \mathbf{F}_e \mathbf{F}_a \, . \tag{9.1}$$

The factorization of $\mathbf{F}$ took place in different nonlinear continuum mechanics scenarios [180], such as thermoelasticity [181], elastoplasticity [182] and biomechanical growth modeling [183], providing an effective mathematical tool for representing the change of configuration of a system undergoing multi-physics processes.

The active deformation $\mathbf{F}_a$ represents an effect of the microscopic cellular interactions driving the tissue contraction, thus it depends on the field variables of the electrophysiological model. A graphical representation of the multiplicative decomposition is provided in Fig. 9.1. In the present work the instantanueous active gradient is prescribed by a phenomenological law which dictates the activation parameters embedded in $\mathbf{F}_a$. The explicit expression of $\mathbf{F}_a$ is presented in section 9.7.

It should be pointed out that, unlike some of the aforementioned applications of the multiplicative decomposition, $\mathbf{F}_e$ and $\mathbf{F}_a$ are not actually given by the gradient of a vector map. From the kinematic perspective, the active deformation leads to an intermediate non-compatible configuration, which must be intended as a stress-free configuration given by a micro-structural rearrangement. The compatibility and equilibrium requirement are matched subsequently by the passive response, which relaxes the body from the intermediate to the final configuration.

Hence, Eq. (9.1) describes a theoretical decoupling that associates the microscale dynamics to the macroscale continuum mechanics indeed [23]. The thermodynamic potential associated with the electromechanical process is defined by the material Helmholtz free energy density Ψ. Without claim of generality, Ψ is assumed to describe an isothermal process following the dependencies:

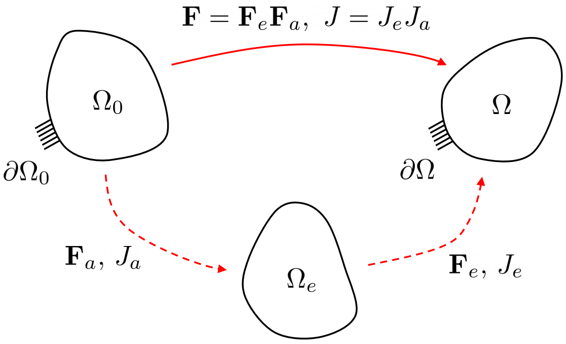


Figure 9.1: Schematic representation of the multiplicative decomposition of the deformation gradient $\mathbf{F}$ into active $\mathbf{F}_a$, and elastic $\mathbf{F}_e$ part, associated with the reference Ω_0, intermediate Ω_e, and current Ω configuration.

$$\Psi = \Psi\left(\mathbf{F}, \mathbf{F}_e, v\right) , \tag{9.2}$$

being v the transmembrane potential. According to the theoretical framework presented in [184], beside the factorization of the deformation gradient tensor, we suppose an additive split of the Helmholtz free energy by assuming a separation of its arguments. Conveniently, Ψ_e can be considered as a function of the elastic material behavior:

$$\Psi\left(\mathbf{F}, \mathbf{F}_e, v\right) = \Psi_e\left(\mathbf{F}_e\right) + \Psi_a\left(\mathbf{F}, v\right) . \tag{9.3}$$

According to these assumptions, the first term corresponds to the strain energy density of the system, which depends only on the elastic part of the deformation gradient. Such a condition has been widely employed in recent works to derive a formulation for nonlinear elastic response [185, 23]. The energy split has proven to be computationally attractive since it allows the choice of any strain energy function, and furthermore enables the use of separate simplified formulations for the active contraction and passive response subproblems.

In this work the active strain approach is embedded in a total Lagrangian formulation, which represents the classical framework for finite elasticity solvers. Thus, the equations of motion are integrated in the material configuration, and the elastic strain energy is thought as a function of the elastic right Cauchy-Green Deformation tensor,

$$\Psi_e = \Psi_e\left(\mathbf{C}_e\right) , \tag{9.4}$$

which is defined following the deformation gradient split:

$$\mathbf{C} = \mathbf{F}_a^T\mathbf{C}_e\mathbf{F}_a . \tag{9.5}$$

It is worth noting that the elastic quantities (denoted by the subscript e), as well as the elastic and active deformation gradient tensors, are all defined with respect to the reference configuration. The aforementioned intermediate configuration does not represent the result of a geometric map,

therefore any kinematic quantity is defined in the reference frame.
We consider the passive material response to be isotropic and hyperelastic. The generalization to the anisotropic case is a straightforward extension and has been already presented in literature [186]. For sake of generality we assume a weakly compressible material law. In this scenario, we address the quantification of the determinant of the two parts of the tensor deformation gradient separately. In curvilinear coordinates:

$$J_e = \frac{\det(F_{e,ij})}{\det(\mathring{G}_{ij})}, \quad J_e = \frac{\det(F_{a,ij})}{\det(\mathring{G}_{ij})}, \quad \text{with}. \tag{9.6}$$

However, they must fulfill the relation:

$$J = J_e J_a \,. \tag{9.7}$$

The active deformation gradient tensor is defined with respect to the curvilinear bases as:

$$\mathbf{F}_a = F_a^{ij} \, \mathring{\mathbf{g}}_i \otimes \mathring{\mathbf{g}}_j \,, \tag{9.8}$$

$$\mathbf{F}_a^{-1} = \bar{F}_{aij} \, \mathring{\mathbf{g}}^i \otimes \mathring{\mathbf{g}}^j \,, \tag{9.9}$$

$$\mathbf{F}_a^{-T} = \bar{F}_{aji} \, \mathring{\mathbf{g}}^i \otimes \mathring{\mathbf{g}}^j \,, \tag{9.10}$$

9.2 Active Strain and Active Stress approaches

The most consolidated approach for translating an electrophysiological activation state into a tissue deformation is based on the concept of active stress. Differently from the aforementioned active strain approach, it entails the definition of an instantaneous muscular tension which must propagate following a suitable evolution model [187]. In a recent contribution [188], the Active Strain and the Active Stress approaches have been compared on simple tests producing identical results on uniaxial deformation and different results in a shear test, unless specific conditions are enforced on the strain energy density. We implemented the active strain approach due to its robustness and thermodynamic consistency [184].
From a mathematical perspective, both approaches must satisfy the frame indifference and ellipticity of the total stress [189]. Form a biological perspective instead, an activation model must satisfy the constitutive behavior observed experimentally on the macroscopic level. The active strain approach is mathematically more robust, since the properties advocated ahead depend on the strain energy function adopted, and no active stress function needs to be tuned. Conversely, the active strain approach provides less chances to fit the physiological behavior with the experimental measurements once the inert properties of the material are prescribed [189]. Recently, active strain and active stress have been conciliated from a thermodynamic perspective through the an additive decomposition of the Helmholtz strain energy density [184].

9.3 The active strain approach in a Lagrangian formulation

Most of the implementations of the active strain approach presented in literature [24, 23, 186] provide the first variation of the elastic strain energy as a function of the first Piola-Kirchhoff stress tensor **P**. Recalling the notation used in chapter 3, it reads:

$$D_{\delta\mathbf{u}}\Pi_i(\mathbf{u}) = \int_{\Omega_0} \mathbf{P} : \nabla\phi \, dV \,, \tag{9.11}$$

for all test functions ϕ. This integral can be obtained directly from integration of the equilibrium equation. The formulation arising from (9.11) is preferred by many authors because the tensor **P** provides an explicit and simple relation with the deformation gradient tensor, on which the multiplicative decomposition is performed $\mathbf{P} = \partial\Psi/\partial\mathbf{F}$. Thus, the underlying active strain approach can be explicitly treated without further derivations if the strain energy density is presented as a function of the deformation gradient. However, the most widespread approach pursued for finite elasticity FEA is the total Lagrangian formulation, whose elastic energy variation is expressed by means of the second Piola-Kirchhoff stress tensor **S**:

$$D_{\delta\mathbf{u}}\Pi_i(\mathbf{u}) = \int_{\Omega_0} \mathbf{S} : D_{\delta\mathbf{u}}\mathbf{E}\, dV\,. \tag{9.12}$$

In order to provide an active strain approach based on the expression (9.12) for finite elasticity we present a detailed derivation of the necessary mathematical relations. The formulas are provided in the most general frame of reference, and they are not restricted by any kinematic assumption. In this connection we present a very general active strain framework to be embedded into a standard finite elasticity formulation based on the definition (9.12), without resorting to dedicated formulations.

9.4 Constitutive equations for the passive material response

In the following, the active strain approach is embedded in the NURBS-based IGA framework described in Chapter 3. The formulation requires the integration of the second Piola-Kirchhoff stress tensor **S**, which is energetically conjugate to the Green-Lagrange strain tensor **E** through the tangent material tensor $\mathbb{C}$. In local curvilinear coordinates these can be written as:

$$\mathbf{S} = S^{ij}\mathring{\mathbf{g}}_i \otimes \mathring{\mathbf{g}}_j\,, \tag{9.13}$$

$$\mathbb{C} = \mathbb{C}^{ijkl}\mathring{\mathbf{g}}_i \otimes \mathring{\mathbf{g}}_j \otimes \mathring{\mathbf{g}}_k \otimes \mathring{\mathbf{g}}_l\,. \tag{9.14}$$

Having provided the strain energy as a function of the right Cauchy-Green tensor, we report the following definitions:

$$\mathbf{S} = 2\frac{\partial\Psi}{\partial\mathbf{C}}\,, \tag{9.15}$$

$$\mathbb{C} = 4\frac{\partial^2\Psi}{\partial\mathbf{C}\partial\mathbf{C}}\,. \tag{9.16}$$

The right Cauchy-Green deformation tensor is defined in the shell continuum as it must be integrated over the thickness to provide the stress resultant (see section 3.4):

$$\mathbf{C} = \mathbf{F}^T\mathbf{F} = G_{ij}\,\mathring{\mathbf{g}}^i \otimes \mathring{\mathbf{g}}^j\,. \tag{9.17}$$

According to Eq. (9.17), which is valid for a general 3D continuum, the covariant coefficients of the deformation tensor are identical to the metric coefficients of the deformed configuration, i.e., $C_{ij} = G_{ij}$. In the shell model, this relation does not hold for the transverse normal direction, i.e., $C_{33} \neq g_{33}$, since $g_{33} \equiv 1$ due to the definition in Eq. (3.7), whereas C_{33} describes the actual thickness deformation, which cannot be neglected in hyperelastic material models. Accordingly, we represent the deformation tensor coefficients by:

$$C_{ij} = \begin{pmatrix} G_{11} & G_{12} & 0 \\ G_{21} & G_{22} & 0 \\ 0 & 0 & C_{33} \end{pmatrix} . \tag{9.18}$$

As shown in Section 3.2, C_{33} can be computed from the in-plane components $G_{\alpha\beta}$ using the plane stress condition. The coefficient matrix of the inverse of the deformation tensor, $\mathbf{C}^{-1} = \bar{C}^{ij}\, \mathring{\mathbf{g}}_i \otimes \mathring{\mathbf{g}}_j$, is obtained as:

$$\bar{C}^{ij} = \begin{pmatrix} G^{11} & G^{12} & 0 \\ G^{21} & G^{22} & 0 \\ 0 & 0 & C_{33}^{-1} \end{pmatrix} . \tag{9.19}$$

9.4.1 Second Piola-Kirchhoff stress tensor

The main algebraic rules for tensor operations employed in the present section are summarized in Appendix B. By exploiting the chain rule in the definition (9.15) and the relation between the deformation tensor and its elastic counterpart, one can write:

$$\begin{aligned} \mathbf{S} &= 2 \frac{\partial \Psi}{\partial \mathbf{C}_e} : \frac{\partial \mathbf{C}_e}{\partial \mathbf{C}} \\ &= \mathbf{S}_e : \frac{\partial}{\partial \mathbf{C}} \left(\mathbf{F}_a^{-T}\, \mathbf{C}\, \mathbf{F}_a^{-1} \right) , \end{aligned} \tag{9.20}$$

where $\mathbf{S}_e$ is the elastic part of the tensor $\mathbf{S}$. Following the dependencies assumed in section 9.6, the active part of the deformation gradient $\mathbf{F}_a$ does not depend on $\mathbf{F}$ itself, thus it does not depend on $\mathbf{C}$ as well.

$$\begin{aligned} \mathbf{S} &= \mathbf{S}_e : \left(\mathbf{F}_a^{-T} \frac{\partial \mathbf{C}}{\partial \mathbf{C}} \mathbf{F}_a^{-1} \right) \\ &= \mathbf{S}_e : \left(\mathbf{F}_a^{-T}\, \mathbb{I}_s\, \mathbf{F}_a^{-1} \right) , \end{aligned} \tag{9.21}$$

where $\mathbb{I}_s$ is the super-symmetric fourth-order identity tensor, arising from the derivative of a second-order symmetric tensor with respect to itself. According to [190], it is defined as:

$$\mathbb{I}_s = \frac{1}{2} \left(\mathring{G}^{ik}\, \mathring{G}^{jl} + \mathring{G}^{il}\, \mathring{G}^{jk} \right) \left(\mathring{\mathbf{g}}_i \otimes \mathring{\mathbf{g}}_j \otimes \mathring{\mathbf{g}}_k \otimes \mathring{\mathbf{g}}_l \right) . \tag{9.22}$$

By switching to the index notation, the second term of the expression (9.21) is computed as:

$$\mathbf{F}_a^{-T} \mathbb{I}^s \mathbf{F}_a^{-1} = \left(\bar{F}_{aab}\, \mathring{\mathbf{g}}^a \otimes \mathring{\mathbf{g}}^b \right) \left(\theta^{ijkl}\, \mathring{\mathbf{g}}_i \otimes \mathring{\mathbf{g}}_j \otimes \mathring{\mathbf{g}}_k \otimes \mathring{\mathbf{g}}_l \right) \left(\bar{F}_{acd}\, \mathring{\mathbf{g}}^c \otimes \mathring{\mathbf{g}}^d \right) , \tag{9.23}$$

with $\theta^{ijkl} = \frac{1}{2} \left(\mathring{G}^{ik}\, \mathring{G}^{jl} + \mathring{G}^{il}\, \mathring{G}^{jk} \right)$. It yields:

$$\begin{aligned} \mathbf{F}_a^{-T} \mathbb{I}^s \mathbf{F}_a^{-1} &= \theta^{ijkl}\, \bar{F}_{aba}\, \delta_i^b\, \bar{F}_{acd}\, \delta_l^c \left(\mathring{\mathbf{g}}^a \otimes \mathring{\mathbf{g}}_j \otimes \mathring{\mathbf{g}}_k \otimes \mathring{\mathbf{g}}^d \right) \\ &= \theta^{ijkl}\, \bar{F}_{aia}\, \bar{F}_{ald} \left(\mathring{\mathbf{g}}^a \otimes \mathring{\mathbf{g}}_j \otimes \mathring{\mathbf{g}}_k \otimes \mathring{\mathbf{g}}^d \right) \\ &= \frac{1}{2} \bar{F}_{aia}\, \bar{F}_{ald}\, \mathring{G}^{ik}\, \mathring{G}^{jl} \left(\mathring{\mathbf{g}}^a \otimes \mathring{\mathbf{g}}_j \otimes \mathring{\mathbf{g}}_k \otimes \mathring{\mathbf{g}}^d \right) + \frac{1}{2} \bar{F}_{aia}\, \bar{F}_{ald}\, \mathring{G}^{il}\, \mathring{G}^{jk} \left(\mathring{\mathbf{g}}^a \otimes \mathring{\mathbf{g}}_j \otimes \mathring{\mathbf{g}}_k \otimes \mathring{\mathbf{g}}^d \right) \\ &= \frac{1}{2} \bar{F}_{aia}\, \bar{F}_{ald} \left(\mathring{\mathbf{g}}^a \otimes \mathring{\mathbf{g}}^l \otimes \mathring{\mathbf{g}}^i \otimes \mathring{\mathbf{g}}^d \right) + \frac{1}{2} \bar{F}_{aia}\, \bar{F}_{ald}\, \mathring{G}^{il} \left(\mathring{\mathbf{g}}^a \otimes \mathring{\mathbf{g}}_j \otimes \mathring{\mathbf{g}}^j \otimes \mathring{\mathbf{g}}^d \right) . \end{aligned} \tag{9.24}$$

Then:

$$
\begin{aligned}
\mathbf{S}_e : \left(\mathbf{F}_a^{-T}\mathbb{I}^s\mathbf{F}_a^{-1}\right) = {} & \left(S_e^{ef}\,\mathring{\mathbf{g}}_e \otimes \mathring{\mathbf{g}}_f\right) : \left[\frac{1}{2}\,\bar{F}_{aia}\,\bar{F}_{ald}\,\left(\mathring{\mathbf{g}}^a \otimes \mathring{\mathbf{g}}^l \otimes \mathring{\mathbf{g}}^i \otimes \mathring{\mathbf{g}}^d\right)\right] + \\
& \left(S_e^{ef}\,\mathring{\mathbf{g}}_e \otimes \mathring{\mathbf{g}}_f\right) : \left[\frac{1}{2}\,\bar{F}_{aia}\,\bar{F}_{ald}\,\mathring{G}^{il}\,\left(\mathring{\mathbf{g}}^a \otimes \mathring{\mathbf{g}}_j \otimes \mathring{\mathbf{g}}^j \otimes \mathring{\mathbf{g}}^d\right)\right] \\
= {} & \frac{1}{2}\,S_e^{ef}\,\bar{F}_{aia}\,\bar{F}_{ald}\,\delta_e^a\,\delta_f^l\,\left(\mathring{\mathbf{g}}^i \otimes \mathring{\mathbf{g}}^d\right) + \frac{1}{2}\,S_e^{ef}\,\bar{F}_{aia}\,\bar{F}_{ald}\,\delta_e^a\,\mathring{G}^{il}\,\mathring{G}_{fj}\,\left(\mathring{\mathbf{g}}^i \otimes \mathring{\mathbf{g}}^d\right) \\
= {} & \frac{1}{2}\,S_e^{al}\,\bar{F}_{aia}\,\bar{F}_{ald}\,\left(\mathring{\mathbf{g}}^i \otimes \mathring{\mathbf{g}}^d\right) + \frac{1}{2}\,S_e^{af}\,\bar{F}_{aia}\,\bar{F}_{ald}\,\mathring{G}^{il}\,\mathring{G}_{fj}\,\left(\mathring{\mathbf{g}}^i \otimes \mathring{\mathbf{g}}^d\right). \qquad (9.25)
\end{aligned}
$$

By matching the bases of each term of Eq. (9.25):

$$
\begin{aligned}
S^{pq}\,\mathring{\mathbf{g}}_p \otimes \mathring{\mathbf{g}}_q &= \frac{1}{2}\,S_e^{al}\,\bar{F}_{aia}\,\bar{F}_{ald}\,\mathring{G}^{ip}\,\mathring{G}^{dq}\,\left(\mathring{\mathbf{g}}_p \otimes \mathring{\mathbf{g}}_q\right) + \frac{1}{2}\,S_e^{af}\,\bar{F}_{aia}\,\bar{F}_{ald}\,\mathring{G}^{il}\,\mathring{G}_{fj}\,\mathring{G}^{jp}\,\mathring{G}^{dq}\left(\mathring{\mathbf{g}}_p \otimes \mathring{\mathbf{g}}_q\right) \\
&= \frac{1}{2}\,S_e^{al}\,\bar{F}_{aia}\,\bar{F}_{ald}\,\mathring{G}^{ip}\,\mathring{G}^{dq}\,\left(\mathring{\mathbf{g}}_p \otimes \mathring{\mathbf{g}}_q\right) + \frac{1}{2}\,S_e^{af}\,\bar{F}_{aia}\,\bar{F}_{ald}\,\mathring{G}^{il}\,\delta_f^p\,\mathring{G}^{dq}\left(\mathring{\mathbf{g}}_p \otimes \mathring{\mathbf{g}}_q\right) \\
&= \frac{1}{2}\,\bar{F}_{aia}\,\bar{F}_{ald}\,G^{dq}\,\left(S_e^{al}\,\mathring{G}^{ip} + S_e^{ap}\,\mathring{G}^{il}\right)\left(\mathring{\mathbf{g}}_p \otimes \mathring{\mathbf{g}}_q\right), \qquad (9.26)
\end{aligned}
$$

the final coefficient equation holds:

$$
S^{pq} = \frac{1}{2}\,\bar{F}_{aia}\,\bar{F}_{ald}\,\mathring{G}^{dq}\,\left(S_e^{al}\,\mathring{G}^{ip} + S_e^{ap}\,\mathring{G}^{il}\right), \qquad (9.27)
$$

which requires the loop on four nested indices $\{i, a, l, d\}$.

9.4.2 Tangent material tensor

In a similar fashion to the derivation of the second Piola-Kirchhoff stress tensor we write:

$$
\begin{aligned}
\mathbb{C} &= 4\,\frac{\partial}{\partial \mathbf{C}}\left(\frac{\partial \Psi}{\partial \mathbf{C}}\right) \\
&= 4\,\frac{\partial}{\partial \mathbf{C}_e}\left(\frac{\partial \Psi}{\partial \mathbf{C}}\right) : \frac{\partial \mathbf{C}_e}{\partial \mathbf{C}}. \qquad (9.28)
\end{aligned}
$$

Now the first term of the previous equation will be computed separately by exploiting the definition of tensor function derivative. We note that the expression (9.26) corresponds to the argument of the derivative divided by 2:

$$
\begin{aligned}
\frac{\partial}{\partial \mathbf{C}_e}\left(\frac{\partial \Psi}{\partial \mathbf{C}}\right) &= \frac{\partial}{\partial\left(C_{e,gh}\,\mathring{\mathbf{g}}^g \otimes \mathring{\mathbf{g}}^h\right)}\left[\frac{1}{4}\,\bar{F}_{aia}\,\bar{F}_{ald}\,\mathring{G}^{dq}\,\left(2\,\frac{\partial \Psi}{\partial C_{e,al}}\,\mathring{G}^{ip} + 2\,\frac{\partial \Psi}{\partial C_{e,ap}}\,\mathring{G}^{il}\right)\left(\mathring{\mathbf{g}}_p \otimes \mathring{\mathbf{g}}_q\right)\right] \\
&= \frac{1}{2}\,\bar{F}_{aia}\,\bar{F}_{ald}\,\mathring{G}^{dq}\,\left(\frac{\partial^2 \Psi}{\partial C_{e,gh}\partial C_{e,al}}\,\mathring{G}^{ip} + \frac{\partial^2 \Psi}{\partial C_{e,gh}\partial C_{e,ap}}\,\mathring{G}^{il}\right)\left(\mathring{\mathbf{g}}_p \otimes \mathring{\mathbf{g}}_q \otimes \mathring{\mathbf{g}}_g \otimes \mathring{\mathbf{g}}_h\right) \\
&= \frac{1}{8}\,\bar{F}_{aia}\,\bar{F}_{ald}\,\mathring{G}^{dq}\,\left(\mathbb{C}_e^{ghal}\,\mathring{G}^{ip} + \mathbb{C}_e^{ghap}\,\mathring{G}^{il}\right)\left(\mathring{\mathbf{g}}_p \otimes \mathring{\mathbf{g}}_q \otimes \mathring{\mathbf{g}}_g \otimes \mathring{\mathbf{g}}_h\right), \qquad (9.29)
\end{aligned}
$$

Provided that the second term of Eq. (9.28) correspond to Eq.(9.24), the final expression of the tangent material tensor can be obtained by the following double contraction:

$$
\mathbb{C} = 4\left[\frac{1}{8}\,\bar{F}_{aia}\,\bar{F}_{ald}\,\mathring{G}^{dq}\,\left(\mathbb{C}_e^{ghal}\,\mathring{G}^{ip} + \mathbb{C}_e^{ghap}\,\mathring{G}^{il}\right)\left(\mathring{\mathbf{g}}_p \otimes \mathring{\mathbf{g}}_q \otimes \mathring{\mathbf{g}}_g \otimes \mathring{\mathbf{g}}_h\right)\right] :
$$

$$\left[\frac{1}{2}\,\bar{F}_{auw}\,\bar{F}_{ayx}\,(\mathring{\mathbf{g}}^{w}\otimes\mathring{\mathbf{g}}^{y}\otimes\mathring{\mathbf{g}}^{u}\otimes\mathring{\mathbf{g}}^{x})+\frac{1}{2}\,\bar{F}_{auw}\,\bar{F}_{ayx}\,\mathring{G}^{uy}\,(\mathring{\mathbf{g}}^{w}\otimes\mathring{\mathbf{g}}_{v}\otimes\mathring{\mathbf{g}}^{v}\otimes\mathring{\mathbf{g}}^{x})\right]. \tag{9.30}$$

The distributive property of the double contraction between fourth-order tensors [191] can be applied:

$$\begin{aligned}\mathbb{C}={}&\frac{1}{4}\,\bar{F}_{aia}\,\bar{F}_{ald}\,\bar{F}_{auw}\,\bar{F}_{ayx}\,\mathring{G}^{dq}\,\delta^{w}_{g}\,\delta^{y}_{h}\,\left(\mathbb{C}_{e}^{ghal}\,\mathring{G}^{ip}+\mathbb{C}_{e}^{ghap}\,\mathring{G}^{il}\right)\left(\mathring{\mathbf{g}}_{p}\otimes\mathring{\mathbf{g}}_{q}\otimes\mathring{\mathbf{g}}^{u}\otimes\mathring{\mathbf{g}}^{x}\right)+\\&\frac{1}{4}\,\bar{F}_{aia}\,\bar{F}_{ald}\,\bar{F}_{auw}\,\bar{F}_{ayx}\,\mathring{G}^{dq}\,\mathring{G}^{uy}\,\mathring{G}_{hv}\,\delta^{w}_{g}\,\left(\mathbb{C}_{e}^{ghal}\,\mathring{G}^{ip}+\mathbb{C}_{e}^{ghap}\,\mathring{G}^{il}\right)\left(\mathring{\mathbf{g}}_{p}\otimes\mathring{\mathbf{g}}_{q}\otimes\mathring{\mathbf{g}}^{v}\otimes\mathring{\mathbf{g}}^{x}\right)\\={}&\frac{1}{4}\,\bar{F}_{aia}\,\bar{F}_{ald}\,\bar{F}_{auw}\,\bar{F}_{ayx}\,\mathring{G}^{dq}\,\left(\mathbb{C}_{e}^{wyal}\,\mathring{G}^{ip}+\mathbb{C}_{e}^{wyap}\,\mathring{G}^{il}\right)\left(\mathring{\mathbf{g}}_{p}\otimes\mathring{\mathbf{g}}_{q}\otimes\mathring{\mathbf{g}}^{u}\otimes\mathring{\mathbf{g}}^{x}\right)+\\&\frac{1}{4}\,\bar{F}_{aia}\,\bar{F}_{ald}\,\bar{F}_{auw}\,\bar{F}_{ayx}\,\mathring{G}^{dq}\,\mathring{G}^{uy}\,\mathring{G}_{hv}\,\left(\mathbb{C}_{e}^{whal}\,\mathring{G}^{ip}+\mathbb{C}_{e}^{whap}\,\mathring{G}^{il}\right)\left(\mathring{\mathbf{g}}_{p}\otimes\mathring{\mathbf{g}}_{q}\otimes\mathring{\mathbf{g}}^{v}\otimes\mathring{\mathbf{g}}^{x}\right).\end{aligned} \tag{9.31}$$

On uniform bases it yields:

$$\begin{aligned}&\mathbb{C}^{pqrs}\,(\mathring{\mathbf{g}}_{p}\otimes\mathring{\mathbf{g}}_{q}\otimes\mathring{\mathbf{g}}_{r}\otimes\mathring{\mathbf{g}}_{s})=\\&=\frac{1}{4}\,\bar{F}_{aia}\,\bar{F}_{ald}\,\bar{F}_{auw}\,\bar{F}_{ayx}\,\mathring{G}^{dq}\,\mathring{G}^{ur}\,\mathring{G}^{xs}\,\left(\mathbb{C}_{e}^{wyal}\,\mathring{G}^{ip}+\mathbb{C}_{e}^{wyap}\,\mathring{G}^{il}\right)\left(\mathring{\mathbf{g}}_{p}\otimes\mathring{\mathbf{g}}_{q}\otimes\mathring{\mathbf{g}}_{r}\otimes\mathring{\mathbf{g}}_{s}\right)+\\&\frac{1}{4}\,\bar{F}_{aia}\,\bar{F}_{ald}\,\bar{F}_{auw}\,\bar{F}_{ayx}\,\mathring{G}^{dq}\,\mathring{G}^{uy}\,\mathring{G}_{hv}\,\mathring{G}^{vr}\mathring{G}^{xs}\left(\mathbb{C}_{e}^{whal}\,\mathring{G}^{ip}+\mathbb{C}_{e}^{whap}\,\mathring{G}^{il}\right)\left(\mathring{\mathbf{g}}_{p}\otimes\mathring{\mathbf{g}}_{q}\otimes\mathring{\mathbf{g}}_{r}\otimes\mathring{\mathbf{g}}_{s}\right)\\&=\frac{1}{4}\,\bar{F}_{aia}\,\bar{F}_{ald}\,\bar{F}_{auw}\,\bar{F}_{ayx}\,\mathring{G}^{dq}\,\mathring{G}^{xs}\left[\mathring{G}^{ur}\,\left(\mathbb{C}_{e}^{wyal}\,\mathring{G}^{ip}+\mathbb{C}_{e}^{wyap}\,\mathring{G}^{il}\right)+\right.\\&\left.\mathring{G}^{uy}\,\left(\mathbb{C}_{e}^{wral}\,\mathring{G}^{ip}+\mathbb{C}_{e}^{wrap}\,\mathring{G}^{il}\right)\right]\left(\mathring{\mathbf{g}}_{p}\otimes\mathring{\mathbf{g}}_{q}\otimes\mathring{\mathbf{g}}_{r}\otimes\mathring{\mathbf{g}}_{s}\right).\end{aligned} \tag{9.32}$$

The final coefficient equation holds:

$$\begin{aligned}\mathbb{C}^{pqrs}={}&\frac{1}{4}\,\bar{F}_{aia}\,\bar{F}_{ald}\,\bar{F}_{auw}\,\bar{F}_{ayx}\,\mathring{G}^{dq}\,\mathring{G}^{xs}\left[\mathring{G}^{ur}\,\left(\mathbb{C}_{e}^{wyal}\,\mathring{G}^{ip}+\mathbb{C}_{e}^{wyap}\,\mathring{G}^{il}\right)+\right.\\&\left.\mathring{G}^{uy}\,\left(\mathbb{C}_{e}^{wral}\,\mathring{G}^{ip}+\mathbb{C}_{e}^{wrap}\,\mathring{G}^{il}\right)\right],\end{aligned} \tag{9.33}$$

which requires the loop on eight nested indices $\{i,a,l,d,u,w,x,y\}$. It should be stressed that elastic quantities S_{e}^{ij} and $\mathbb{C}_{e}^{ijkl}$ do not have a kinematic meaning, but they are defined by analogy with their total counterpart.
Eqns. (9.26) and (9.33) provide the expressions of the stress and material tensor for an active tissue with a prescribed active deformation gradient. These expressions refer to any three-dimensional domain, and can be reduced to the particular case of a Cartesian frame. Thus, they are suitable to be implemented in the most widespread finite elasticity formulations, if accompanied with a consistent strain energy function. Eventually, one should notice that the Green-Lagrange strain tensor does not need any decomposition since it is intended as a virtual strain.

9.5 Enforcement of the plane stress condition

The plane stress condition $S^{33}=0$ is satisfied by iteratively solving for C_{33}, using the Newton linearization of the plane stress condition presented in [28, 192]:

$$S^{33}+\frac{\partial S^{33}}{\partial C_{33}}\Delta C_{33}=S^{33}+\frac{1}{2}\mathbb{C}^{3333}\Delta C_{33}=0\,. \tag{9.34}$$

From Eq. (9.34) we obtain the incremental update:

$$\Delta C_{33} = -2\frac{S^{33}}{\mathbb{C}^{3333}} \,, \tag{9.35}$$

$$C_{33} = C_{33} + \Delta C_{33} \,, \tag{9.36}$$

where I indicates the iteration step. With the updated $\mathbf{C}$, we compute the updates of $\mathbf{S}(\mathbf{C})$ and $\mathbb{C}(\mathbf{C})$. As an example, let us consider the following compressible Neo-Hookean strain energy function, which was already used in well-established active-strain formulations [24] and consists of an additive split into isochoric and volumetric parts:

$$\Psi(\mathbf{C}_e) = \frac{1}{2}\mu\left(\mathrm{tr}(\mathbf{C}_e) - 2\ln(J_e) - 3\right) + \frac{1}{2}K\left(J_e - 1\right)^2 . \tag{9.37}$$

with μ, K as the shear and bulk moduli. Regarding the cardiac tissue activation, similar compressible Neo-Hookean functions have already been used in computational models [24], and further studies suggest that a compressible model is able to adequately reproduce the experimental data [193]. The 3D stress and material tensors are obtained, according to Eqs. (9.28) and (9.33), where the elastic counterpart are:

$$S_e^{ij} = 2\frac{\partial \Psi_e}{\partial C_{e\,ij}} = \mu\left(\mathring{G}^{ij} - \bar{C}_e^{ij}\right) + K\bar{C}_e^{ij}\left(J_e^2 - J_e\right) , \tag{9.38}$$

$$\begin{aligned} \mathbb{C}_e^{ijkl} =& 4\frac{\partial^2 \Psi_e}{\partial C_{e\,ij}\partial C_{e\,kl}} = \mu\left(\bar{C}_e^{ik}\bar{C}_e^{jl} + \bar{C}_e^{il}\bar{C}_e^{jk}\right) + \\ & 2K\left[\bar{C}_e^{ij}\left(J_e^2 - \frac{1}{2}J_e\right)\bar{C}_e^{ik}\bar{C}_e^{kl} - \frac{1}{2}\left(J_e^2 - J_e\right)\left(\bar{C}_e^{ik}\bar{C}_e^{jl} + \bar{C}_e^{il}\bar{C}_e^{jk}\right)\right] . \end{aligned} \tag{9.39}$$

As initial condition we use $C^0_{ij} = G_{ij}$, where the in-plane components remain invariant throughout the iteration, $C_{\alpha\beta} \equiv G_{\alpha\beta}$, and only C^I_{33} is updated. With $C^{(I+1)}_{33}$ obtained according to Eqs. (9.35)–(9.36), the elastic part of the deformation tensor $\mathbf{C}_e$ is updated by formula (9.5), then the new values of $S^{ij}_{(I+1)}$ and $\mathbb{C}^{ijkl}_{(I+1)}$ are computed. This procedure is repeated until the plane stress condition is satisfied within a defined tolerance. Finally, the statically condensed material tensor $\tilde{\mathbb{C}}$ is computed according to Eq. (3.31), and only the in-plane components $S^{\alpha\beta}$ and $\mathbb{C}^{\alpha\beta\gamma\delta}$ are used for the shell model. In this scenario, arbitrary 3D material models can be used.

The iterative procedure to enforce the plane stress condition is carried out on the total deformation tensor $\mathbf{C}$, whereas the tensor derivatives of the strain energy function are computed with respect to its elastic counterpart. The steps of such an iterative procedure are summarized in table 9.1.

9.6 Active material response

The dynamics behind the active deformation occurring at the cell level, involves much smaller temporal and spatial scales with respect to the macroscopic modeling framework. Our numerical procedure ignores the complex intracellular dynamics and the stochastic mechanisms which regulate the ionic concentration changes responsible for the active deformation, in order to get a model with reasonable computational complexity. Thus, the active deformation does not result from balance equations, but it is directly dictated by means of a phenomenological activation model. The energy spent in the active contraction (9.3) is not neglected, however the coupling

```
(1) update the coefficient $(C_{33})^I$
(2) extract the coefficients of the elastic part $\left(C_{e,ij}\right)^{I+1}$ by:
    $(C_{e,ij})^{I+1} = F_{a,il}\, F_{a,jk} \left(\bar{C}^{mn}\right)^I \mathring{G}_{lm}\, \mathring{G}_{nk}$
(3) compute $\left(S_e^{ij}\right)^{I+1}$ and $\left(C_e^{ijkl}\right)^{I+1}$ via definition (9.38), (9.39)
(4) get $(S^{pq})^{I+1}$ and $(S^{pqrs})^{I+1}$ from the coefficient equations (9.27) and (9.33)
(5) $(\Delta C_{33})^{I+1} = -2\left(S^{33}\right)^{I+1} / \left(\mathbb{C}^{3333}\right)^{I+1}$
(6) $(C_{33})^{I+1} = (C_{33})^I + \Delta(C_{33})^{I+1}$
(7) check convergence: $\left|(\Delta C_{33})^{I+1}\right| < \epsilon$
```

Table 9.1: Algorithm for the enforcement of plane stress condition within the active-strain framework; the letter I indicates the iteration step, ϵ is the predefined convergence tolerance. The procedure needs to be repeated at each quadrature points in the thickness direction.

between passive and active configurations is enforced by a purely kinematic transformation.

In the material law presented in section 9.5, the anisotropy that characterizes the tissues of interest has not been addressed. In this work, we follow the simplified approach of [24] accounting for the anisotropic behavior of the fibers only by assigning direction-specific active deformations in the active part of the decomposition. Hence, the passive elastic response is assumed to be isotropic.

For sake of simplicity we assume that the tissue is dominated by two fiber layers with orientation corresponding to the NURBS directions θ^1 and θ^2. In a similar fashion to [142], the active part of the deformation gradient takes the following form:

$$\mathbf{F}_a = \mathbf{I} - \gamma \left[a_1 \left(\mathring{\mathbf{g}}^1 \otimes \mathring{\mathbf{g}}^1\right) + a_2 \left(\mathring{\mathbf{g}}^2 \otimes \mathring{\mathbf{g}}^2\right)\right] + \gamma_n \left(\mathring{\mathbf{g}}^3 \otimes \mathring{\mathbf{g}}^3\right), \tag{9.40}$$

where γ is a smooth activation function depending on the excitation state and a_1 and a_2 are material parameters controlling the intensity of the active contraction/dilatation in the two fiber directions.

Bringing the terms of Eq. (9.40) on the same basis, consistently with the definition in Eq. (9.8), the coefficient equation for $\mathbf{F}_a$ yields:

$$F_a^{ij} = G^{ij} - \gamma \left(a_1 \mathring{G}^{1i} \mathring{G}^{1j} + a_2 \mathring{G}^{2i} \mathring{G}^{2j}\right) + \gamma_n \mathring{G}^{3i} \mathring{G}^{3j}. \tag{9.41}$$

We assume that the active deformation is also volume preserving, thus:

$$J_a = \frac{\det\left(F_a^{ij}\right)}{\det\left(G^{ij}\right)} = 1. \tag{9.42}$$

This relation can be used to compute the activation parameter in the thickness direction:

$$\gamma_n = \frac{1}{J_a^p} \left(\mathring{G}^{11} \mathring{G}^{22} - \left(\mathring{G}^{12}\right)^2\right) - 1, \tag{9.43}$$

being J_a^p the in-plane determinant of the active deformation tensor:

$$J_a^p = \left[\mathring{G}^{11} - \gamma \left(a_1 \left(\mathring{G}^{11}\right)^2 + a_2 \left(\mathring{G}^{12}\right)^2\right)\right] \left[\mathring{G}^{22} - \gamma \left(a_1 \left(\mathring{G}^{12}\right)^2 + a_2 \left(\mathring{G}^{22}\right)^2\right)\right] -$$

$$\left[\mathring{G}^{12} - \gamma\left(a_1\mathring{G}^{11}\mathring{G}^{12} + a_2 g^{21}\mathring{G}^{22}\right)\right]^2 \tag{9.44}$$

9.7 Tissue activation function and electro-mechanical coupling

The generation of active force within a generic smooth muscle cell is assumed to depend strongly on the intracellular Ca^{2+} concentration. The Ca^{2+} concentration itself is regulated by a complex spatio-temporal dynamics involving both the activation of voltage-dependent calcium channel and intracellular stochastic mechanisms. In order to maintain the generality of the present model, we neglect the intracellular multiscale calcium-dynamics and use a direct relation between the transmembrane voltage and the active strain activation parameter γ, presented in [142]:

$$\gamma(v) = \left(1 - \exp\left(-\beta_1\left(v - v_t\right)\right)\right)\left(1 - \exp\left(-\beta_2\left(v - v_t\right)\right)\right) H\left(v - v_t\right) . \tag{9.45}$$

Here, v denotes the local transmembrane potential value. v_t is the normalized opening voltage of the voltage-dependent Ca^{2+} channels (VDCC), β_1 is a parameter resembling the homogenized Ca^{2+} dynamics, and β_2 describes the opening dynamics of the VDCC. $H(x)$ denotes the Heaviside step function. According to Eq. (9.45), the activation function must lay within the interval $\gamma \in [0, 1]$, where a positive γ produces a fiber shortening.

From the computational perspective the calculation of the activation function is performed at the quadrature points of the structural domain. This requires the interpolation of the transmembrane potential, known at the control points of the electrophysiological mesh. This can be easily implemented by means of the isoparametric paradigm naturally embedded in the finite element formulation.

9.8 Mechano-electric feedback modeling

The mechano-electric feedback is neglected from the constitutive perspective, as mentioned in section 9.6, but a purely geometric feedback is inherently achieved in the present formulation. A widespread modeling approach, relying on simple phenomenological considerations, consists of the inclusion of a Stress-Assisted-Diffusion (SAD) effect and a Stretch-Activating-Current (SAC) contribution [124, 194]. The first collects the homogenized effects of the deformation field on the diffusion processes altering the spatio-temporal patterns of the membrane voltage. The second relates the deformed mechanical state to the excitability of the medium via additional reaction functions (ionic-like currents) [194]. In the present work the deformation is thought to affect the diffusion process via the inherent geometric coupling between the computational domains. This SAD effect is the only feedback effect considered since it was proved to qualitatively match the patterns observed in experimental works, with higher fidelity then SAC [195]. By assuming a quasi-static deformation for the continuum body, its macroscopic response can be thought as independent from the diffusion process. On the contrary, the diffusion process will strongly depend on the mechanical state of the tissue.

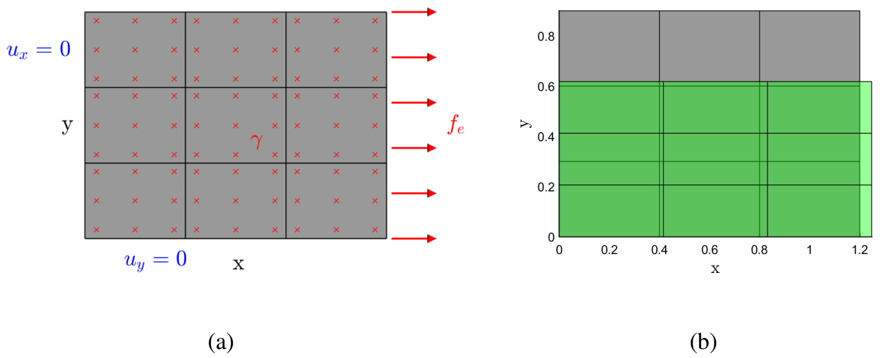
In a curvilinear framework the geometric feedback describing the stress assisted diffusion is naturally included since the conductivity tensor is computed with respect to the local metric vectors, and no further operations are needed.

9.9 Unplugged active strain solver: numerical experiments

This section is devoted to the numerical investigation of the passive material response. We present several numerical tests involving different geometric configurations as well as different distributions of the activation function. In the following cases, the activation is dictated by a prescribed function $\gamma = \gamma(\theta^1, \theta^2, h)$ defined over the shell quadrature points. We emphasize that, in the present section, the activation function field does not represent a physiologically realistic condition, but it is thought as a mere loading condition for numerical investigation purposes.

9.9.1 Flat plate with combined loads

As a first example, we simulate a plate undergoing membrane deformation only in a two-dimensional configuration. A flat plate of dimensions $1.2 \times 0.9 \times 0.01\,m$ is considered. The plate is constrained as illustrated in Fig. 9.2, and it is subject to a traction force in conjunction with a uniform activation function. Equal material activation parameters are considered $a_1 = a_2 = 1.0$. The material is assumed to have a Neo-Hookean strain energy function (9.37) with $\mu = 1.5 \times 10^6\,N/m^2$, $K = 7.445 \times 10^7\,N/m^2$. The applied line load has uniform value $f_e = 9000\,N/m$, whereas the activation function provides a uniformly distributed gradient with $\gamma = 0.2$. The line load applied at the right edge induces an element elongation in the x direction, which must be compensated by a shortening in the other directions due to volume conservation. However, this coexists with a uniformly distributed active gradient which forces the body to an in-plane contraction of the elements. As a consequence of such conflicting loads, the plate undergoes a limited elongation in the x-direction, but a significant shortening in the y-direction (Fig. 9.2). The corresponding principal stretch values (defined in Eq. (9.46)) at the plate midpoint are $\lambda_1 = 1.039$, $\lambda_2 = 0.6868$, $\lambda_3 = 1.379$.



(a) (b)

FIGURE 9.2: (a) Schematic of the boundary conditions and external forces over the flat plate; red markers denotes the quadrature points location, where the activation function γ is defined. (b) Reference and final configuration of the flat plate test.

$$\lambda_1 = \sqrt{A_{11}/\mathring{A}_{11}}, \quad \lambda_2 = \sqrt{A_{22}/\mathring{A}_{22}}, \quad \lambda_3 = \sqrt{C_{33}}\,. \tag{9.46}$$

The correctness of the tensor calculations for the multiplicative split in (9.1) is verified by monitoring the convergence rate of the Newton-Raphson procedure (Fig. 9.3) for the geometrically nonlinear treatment described in section 3.4. The convergence parameter R_i is computed as the ratio of the residual array norm and the sum of external and first-iteration internal forces. The expected quadratic convergence is correctly obtained, suggesting that the active-strain implementation does not affect the linearization of the internal work. It is worth noting that the reference configuration has non-unitary base vectors, although the local bases maintain the orthogonality in the deformed configuration. Hence, the computation of internal forces and the corresponding derivatives does include non-unitary metric tensor coefficients.

$$R_i = \frac{||\mathbf{R}_i||}{\left\|\mathbf{f}_i^{(1)}\right\| + ||\mathbf{f}_e||}. \tag{9.47}$$

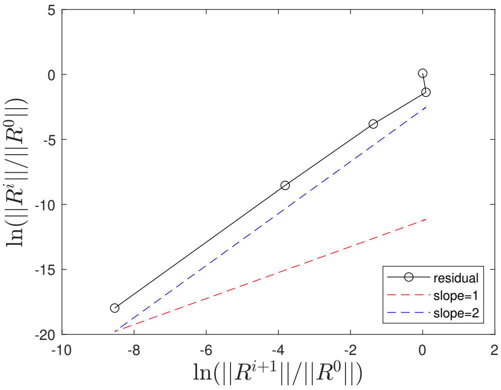


FIGURE 9.3: Convergence of the Newton-Raphson iterations for the flat plate case; R^i indicates the residual of the i-th iteration.

9.9.2 Hinged arc

As a second example we present a hinged arc subject to an active deformation gradient which induces a fiber lengthening. A 40° circular arc with radius $R = 1\,m$ and thickness $t = 0.01\,m$ is hinged at the shortest edges and free at the others. Such a constraint condition ensures that a membrane stretch acting in the azimuthal direction generates out-of-plane deformations. In particular, a uniform activation function $\gamma = -0.3$ is applied, and the material activation parameters $a_1 = 1.0$ and $a_2 = 0.0$ are assumed. The input parameters for the passive material response are taken from the previous numerical example, as well as the strain energy function definition.

The values of the material activation parameters entail an active gradient acting only the azimuthal NURBS direction only. This setting allows to perform an accuracy analysis by h-refinement along just one direction, for sake of simplicity. The analysis is performed for quadratic and cubic shape functions since they are the values used for the other cases presented in the manuscript. In the other direction one linear elements is employed.

To this extent the relative error on the generic quantity η is evaluated by:

$$\epsilon(\eta) = \frac{|\eta - \eta^e|}{\eta^e}, \tag{9.48}$$

with respect to an extrapolated reference solution η^e, which is obtained following the procedure proposed by [108]. We evaluated the convergence rate by means of a local and a global indicator, i. e. the vertical displacement of the point P, $u(P)$, and the discrete strain energy E_s. As displayed by Fig. 9.5, the solution converges to the reference value with the expected rate

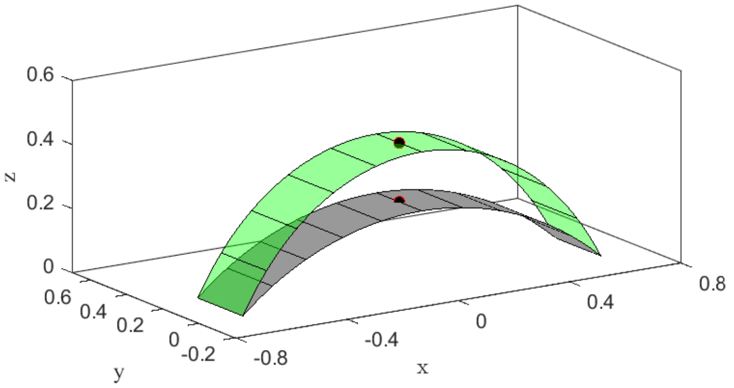
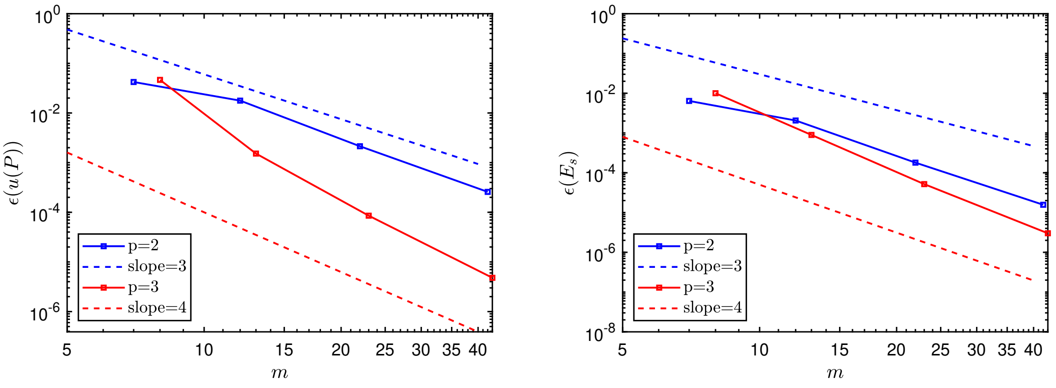


Figure 9.4: Reference (gray) and final (green with translucency) configuration for the hinged arc case. The point P, used for the accuracy analysis is located at the parametric coordinates {0.5, 0.5}.



(a) (b)

Figure 9.5: Convergence plot of the displacement of point P under an active deformation gradient (a). Convergence plot of the discrete strain energy E_s (b).

for both quantities. The vertical displacement of the point P obtained with the finest mesh is $u(P) = 0.1874\,m$.

9.9.3 Clamped shell

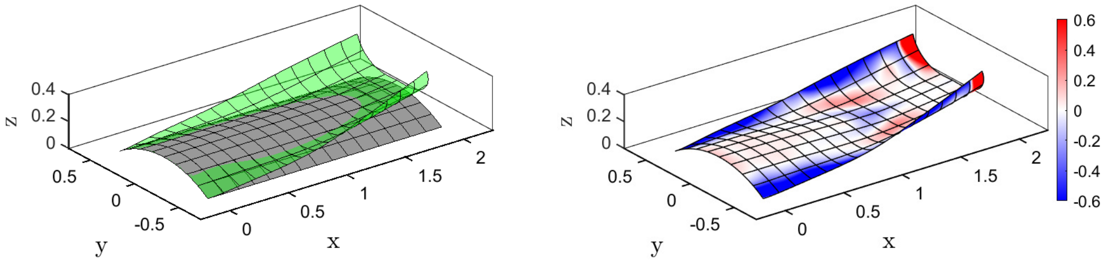
The definition of active deformation gradient presented in Eq. (9.40) entails that out-of-plane deformations can arise only as a consequence of the applied boundary conditions, unless the activation function is assumed to vary through the thickness. In this scenario, the third test-case is built by applying a thickness-variable activation function to a clamped shell. We considered a cylindrical sector spanning 30° of radius $R = 1\,m$, length $L = 2\,m$ and thickness $t = 0.0075\,m$, with one of the curved edges being clamped. The material law described in section 9.5 is employed with $\mu = 1.6 \times 10^6\, N/m^2$ and $\nu = 0.49$. Here the domain is discretized by 8×15 cubic elements. A non-uniform activation function is chosen to force the free edge to bend against its initial curvature.

$$\gamma(\theta^1, \theta^2, \theta^3) = 0.08\, \theta^3\, \theta^2 \left(-|\theta^1|\right). \tag{9.49}$$

As a consequence of such a deformation, a variation in the local curvature is expected in the other NURBS direction. This example provides a fully three-dimensional variation in the configuration, and it represents a challenging task for the nonlinear solver since severe changes in curvature are localized in tiny regions. In Fig. 9.6, the distribution of the local change in curvature $\Delta\mathcal{K}$ is plotted over the surface, suggesting that a fiber contraction in the azimuthal direction induces a significant change in the curvature (Eq. (9.50)) along the axial direction.

$$\Delta\mathcal{K} = \frac{\det(B_{\alpha\beta})}{\det(A_{\alpha\beta})} - \frac{\det(\mathring{B}_{\alpha\beta})}{\det(\mathring{A}_{\alpha\beta})}. \tag{9.50}$$

We can conclude that such a complex active gradient provides a realistic deformation pattern and it confirms the ability of the method to deal with large deformations with a limited number of degrees of freedom.



(a)

(b)

FIGURE 9.6: (a) Reference (gray) and deformed (green with translucency) configuration for the clamped shell case. (b) Surface distribution of the Gaussian curvature change $\Delta\mathcal{K}\,[\mathrm{m}^{-2}]$, defined in Eq. (9.50) over the deformed configuration.

9.10 Coupled electromechanical solver

We present two numerical examples of coupled electromechanical problem resolved in a sequential way. Several authors proposed a monolithic framework [196, 197, 198] for the implementation of the coupled problem, however the staggered approach is widely preferred for sake of simplicity and flexibility of the formulation. The mechanical problem is resolved following the widespread assumption of quasistatic behavior. The role of inertia in the cardiac electromechanics has been addressed in [199]. The quasistatic hypothesis allows to decouple the time-stepping procedure of the sub-problems. We solve the elastic problem, which is computationally far more onerous than the electric one, each n^* monodomain steps, where n^* is the number of steps needed to achieve a 2% variation in the activated surface area. Several authors exploited also the uncoupling of the computational meshes for each sub-problem, however this strategy is avoided to limit the complexity of the implementation.

9.10.1 Activation of a patch with double curvature

We consider a two-dimensional slab of tissue of dimensions $[0,1] \times [0,0.3]\, cm^2$ with constrained short edges (clamped-pinned) and free long edges. The tissue is assumed to have uniform capacity $C_m = 1\, mF/cm^3$ with a surface-to-volume cell ratio of $\chi = 1\, cm^{-1}$ and isotropic conductivity $D_{ISO} = 10^{-4}\, S/cm$. We assume that the conductivity directions are aligned with the Cartesian base vectors $\mathbf{e}_1 = [1,0]$, $\mathbf{e}_2 = [0,1]$. The Aliev-Panfilov current model described in the previous chapter is employed. Mechanically the tissue is defined by a compressible hyperelastic material with strain energy function defined as in (9.37), Poisson's ratio equal to $\nu = 0.49$, bulk modulus $\mu = 1.4 \times 10^6$. The activation function follows the definition in (9.45), with $v_t = 0.25\, mV$, $\beta_1 = 0.5, \beta_2 = 0.5$ and material excitation parameters $a_1 = 0.5$, $a_2 = 0.25$.
The domain is discretized by 150×40 quadratic IGA elements, and the monodomain simulation is advanced in time with the time-step size $\Delta t = 0.01\, ms$. We set three integration points across the thickness and an absolute tolerance for the Newton process equal to $\epsilon = 10^{-7}$. The simulation was initiated forcing the transmembrane potential to $v = 1.0$ at the right edge of the slab for $1\, ms$ to allow the propagation towards the left edge. The expected in-plane stretch of the tissue was observed (see Fig. 9.7). A strongly localized tissue contraction in the y- and x-direction was observed in correspondence of the depolarization front, compared with the less steep relaxation induced at the repolarization region. The distribution of the principal stretches is illustrated at the instant $t = 50\, ms$ over the slab center-line in Fig. 9.8. We notice that the nearly incompressible material $\nu = 0.49$, in conjunction with the algorithmic treatment for the plane stress condition enforcement (the reader should refer to section 9.5), provide a stable Jacobian determinant distribution and a minimal displacement from unity. In this connection, the percentage displacement

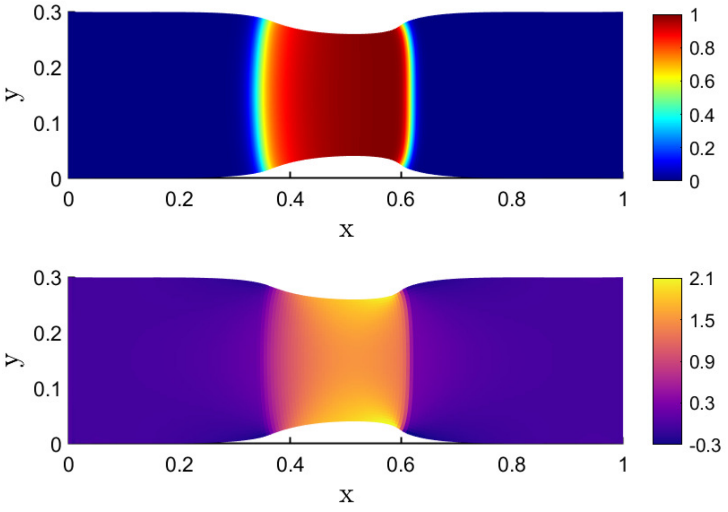


FIGURE 9.7: Transmembrane potential field in the deformed configuration at instant $t = 50\, ms$ (top panel). Correspondent percentage variation of the local Jacobian determinant from unity, as defined in (9.51) (bottom panel).

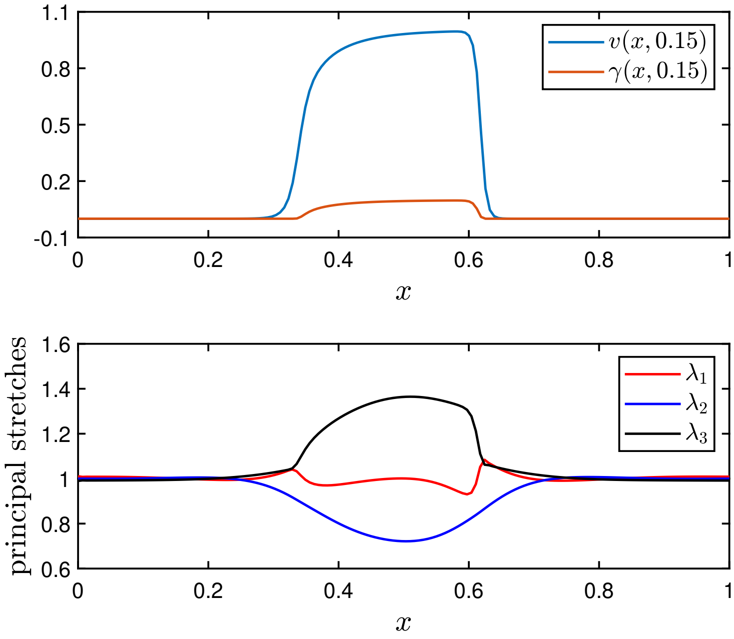


FIGURE 9.8: Transmembrane potential over the slab center-line at $y = 0.15\,cm$ in the deformed configuration at instant $t = 50\,ms$ (top panel). Correspondent distribution of principal stretches, as defined in 9.46 (bottom panel)

of the Jacobian determinant, defined for the i-th element as:

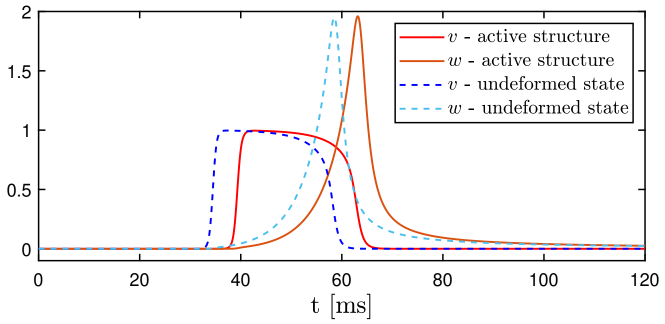
$$J_i^{\%} = |1 - J_i| \times 100\,. \tag{9.51}$$

is plotted at the time instant $t = 50\,ms$ in Fig. 9.7. Eventually, the effect of the geometric feedback enforced via diffusion effect is quantified by comparing the potential time-traces of a point in the domain (see Fig. 9.9) for the pure electrophysiologic stimulus propagation and for the active electromechanical stimulation. The physical point of coordinates $[0.5, 0.5]\,cm$ experiences almost an identical evolution of the field variables in both cases, whereas, an anticipation is noticed if the potential is measured in the displaced position, with respect to the undeformed case. This confirms the effectiveness of the geometrical feedback proposed within the present coupled framework.

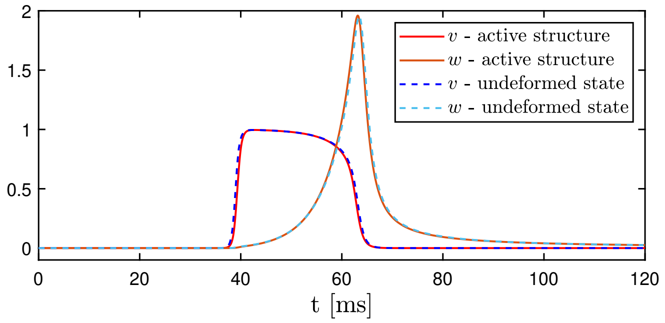
9.10.2 Contraction of a spherical patch

We finally propose a coupled test where in-plane strains induced by means of the activation function (9.45) generate complex out-of-plane deformations. A spherical patch of radius $r = 4\,cm$, angle $\Theta = 53°$, and thickness $h = 0.01cm$, with hinged edges, is considered, and a Neo-Hookean material with bulk modulus $\mu = 1.4 \times 10^6$ and Poisson's ratio $\nu = 0.46$ is assumed. Such a geometry is deformed to obtain a Gaussian curvature change in the central region. This is achieved with the static simulation of the initial geometry under a localized load $F = 13\,N\,\mathbf{e}_3$ in the shell mid-point, which provided the configuration represented in Fig. 9.10. All computations have been performed with a 160×160 cubic IGA elements and 3 quadrature points through the thickness.

Starting from the stress-free configuration in Fig. 9.10, a local stimulus of intensity i_a =



(a)



(b)

FIGURE 9.9: (a) Time evolution of the transmembrane potential v and recovery variable w detected at element-wise position in the deformed and undeformed case. (b) Time evolution of the transmembrane potential v and recovery variable w detected at the physical point $0.5, 0.15$.

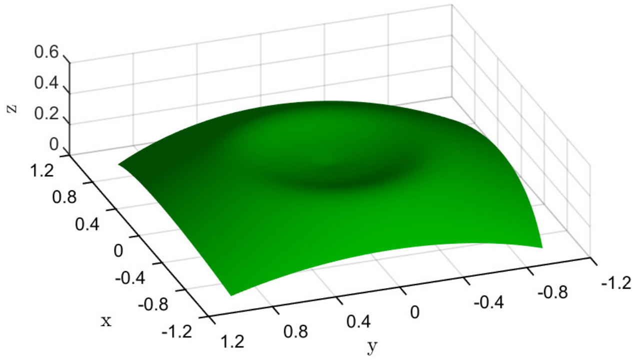


FIGURE 9.10: Reference configuration of the spherical patch test-case, after static deformation induced by a concentrated load.

$5\,mA/cm^2$ is applied for $1\,ms$ to initiate the propagation of a circular action potential. The tissue electric parameters are: capacity $C_m = 1\,mF/cm^3$, surface-to-volume cell ratio of $\chi = 1\,/cm$, isotropic conductivity $D_{ISO} = 10^{-4}\,S/cm$. We assume that the conduction fibers are aligned with the mesh in the reference configuration. The activation function follows the definition in (9.45), with $v_t = 0.25\,mV$, $\beta_1 = 0.5$, $\beta_2 = 0.5$ and material excitation parameters $a_1 = 0.65$, $a_2 = 0.65$.

A time step size of $\Delta t = 0.01\,ms$ is employed for the monodomain model. The geometry of the

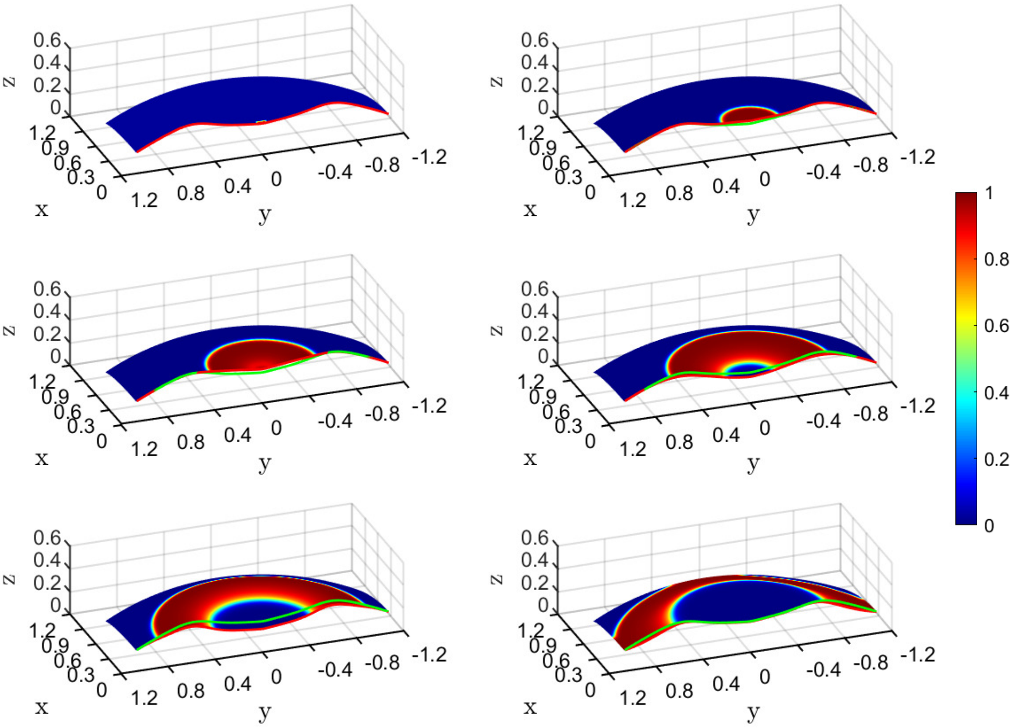


FIGURE 9.11: Time evolution of the transmembrane potential and deformed configuration for the spherical patch test, over half of the geometry. The center-line is highlighted in the reference (green line) and current (red line) configurations. From left to right, we report the time instants: $t = 0.15\,ms$, $t = 7.5\,ms$, $t = 15\,ms$, $t = 22.5\,ms$, $t = 30\,ms$, $t = 37.5\,ms$.

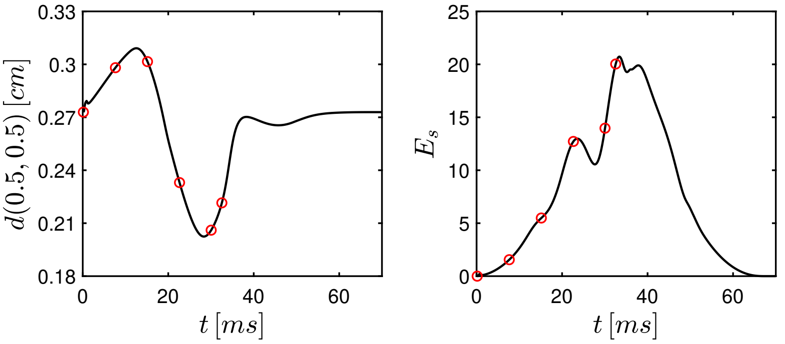


FIGURE 9.12: Time evolution of the mid-point displacement (left) and discrete strain energy (right), for the spherical patch test. Red circles denote the time instants corresponding to the configurations plotted in Fig. 9.11

tissue, as well as the constraint condition is expected to provide a upward and downward vertical displacement of the mid-point when the action potential approaches the regions of the shell with opposite curvature. We plot six instantaneous configurations (see Fig. 9.11) highlighting the position of the center-line to illustrate the expected dynamics. Until the emergence of the repolarization front, the tissue contraction in the convex central region induces a lifting of the mid-point. As the electrophysiologic stimulus forms an annulus and approaches the shell outermost region, a strong bending deformation generates a lowering of the mid-point. As a matter of fact the strain energy of the system reaches a maximum when the action potential is in the maximum curvature region (see Fig. 9.12).

This numerical experiment was conducted with the purpose of testing the numerical method with an active strain pattern generating large deformations in a high curvature region. In view of the results, the thickness integration procedure and in general the nonlinear treatment, are considered robust enough to deal with real-life scenarios.

10 Conclusions and outlook

The core of the present work is the development of numerical tools for the investigation of the two multiphysic problems by means of the Isogeometric concept. The first part is devoted to the modeling of the fluid-structure interaction dynamics of a thin shell immersed in an incompressible flow. The second part concerns the development of a solver for electromechanical stimulation of thin muscular tissues. Separated conclusions are drawn for sake of clarity.

10.1 Fluid-structure interaction

A numerical method for the solution of fluid-structure interaction problems concerning thin shells immersed in incompressible flows is presented. In the context of partitioned solvers, a finite-difference flow solver is coupled with an Isogeometric Analysis tool for thin shells, with the aim of providing accurate solutions for moderate Reynolds numbers FSI problems. The coupling between the solvers is enforced by a Direct-Forcing Immersed Boundary approach based on a Moving-Least-Squares interpolation of the field variable, to reconstruct the solution at the fluid-solid interface. A relevant advancement in the context of partitioned solvers comes from the efficient variable exchange which does not require reciprocal constraints on the spatial discretization.

The effectiveness of the interface reconstruction in the flow field as well as the spatial and temporal accuracy of the FSI procedure have been tested against several three-dimensional cases. The method was able to accurately capture the dynamics of FSI problems of biological and engineering inspiration, providing satisfactory results under fundamentally different physical and numerical operating conditions. A clamped flag for energy harvesting was simulated and different dynamical regimes have been captured in agreement with the experimental measurements. Moreover, the solver was tested against a case of flow-induced vibrations on a L-shaped plate and a buoyant seaweed. The latter required an iterative FSI procedure to get around the coupling instabilities arising from the explicit load evaluation, typical of sequentially coupled partitioned solvers. To this extent, the iterative coupling has been enhanced by means of an adaptive under-relaxation technique, which minimizes the computational expense of the iterative procedure. Furthermore, segregated refinement studies have been performed on a well-established benchmark case, isolating the effect of the fluid grid refinement and of the shell h-refinement. Such analysis provided a convergence trend consistent with the theoretical accuracy of the numerical schemes in time and space.

Further developments of the present method would include the implementation of a numerical treatment for reducing the force spurious oscillations over the surface. Another relevant advancement would come from the implementation of a dedicated near-wall treatment for the reconstruction of the boundary layer gradients occurring at high Reynolds numbers.

10.2 Electromechanical activation of thin muscular tissues

In second instance, we presented a computational framework for simulating the electromechanical activation of thin cardiac tissues. The model proposed aims at reproducing the propagation of the action potential over a surface with the corresponding tissue contraction effects. In this connection, two NURBS-based IGA solvers have been coupled with a staggered procedure.
The propagation of the action potential is described by a monodomain model, consisting of a reaction-diffusion equation and a set of nonlinear ODEs. The underlying assumptions of the monodomain model and the surface approximation are discussed in detail. The monodomain model is discretized in time by a semi-implicit scheme and in space by an IGA approach, whereas a simple phenomenological model is implemented to provide a description of the ionic current fluxes. The effectiveness of the implementation is verified comparing the shape of the action potential and the conduction velocity on a linear propagation test with those observed in literature. Furthermore, on the same test, two different ionic current interpolation techniques have been compared and assessed in terms of accuracy and numerical stability. Regarding the diffusion operator, a curvilinear coordinates description is employed, with the aim of achieving a straightforward implementation for arbitrarily surfaces in the three-dimensional space.
For mechanically active tissues the loading condition is provided by means of the well-established active strain approach. The active strain, enforced with a multiplicative decomposition of the deformation gradient tensor, is introduced theoretically and it is grafted into the weak Kirchhoff-Love shell formulation. A novel contribution is given by the fact that the active strain approach is tailored for a standard weak form used in finite elasticity, where the internal virtual work is defined by the second Piola-Kirchhoff stress tensor and the energetically conjugate Green-Lagrange strain tensor $\delta W = \mathbf{S} : \delta\mathbf{E}$. The useful tensor expressions are presented in a curvilinear framework for three-dimensional hyperelastic materials for the generality purpose.
In this scenario, the passive elastic response of the unplugged active strain solver is investigated by numerical experiments and convergence analyses. Eventually, we tested the coupled electromechanical method with two simple cases providing in-plane and out-of-plane deformations.
The present implementation can be further developed to include a decoupling of the spatial discretization of the two sub-problems by means of a nested meshes approach. From the model perspective the orthotropic material treatment can be easily addressed employing a suitable strain energy function. In addition a more realistic mechano-electric feedback can be included at the constitutive level, in place of the simple geometric feedback.

A straightforward outlook of this thesis would include the staggered coupling of the fluid-structure interaction and electromechanical solvers to address full-scale multiphysic investigations of immersed active swimmers.

A Construction of the MLS transfer function

The procedure for the calculation of the Moving-Least-Square transfer operator is described in the present section. The Eulerian velocity $\hat{q}_i^l$ at the Lagrangian marker location $\mathbf{X}$ can be approximated in its support domain by:

$$\hat{q}_i^l(\mathbf{X}) = \mathbf{p}^T(\mathbf{X})\,\mathbf{a}(\mathbf{X}) = \sum_{j=1}^{m} p_j^l(\mathbf{X})\,a_j(\mathbf{X})\,. \tag{A.1}$$

$\mathbf{p}^T(\mathbf{X})$ is the basis functions vector of length m and $\mathbf{a}(\mathbf{X})$ is vector of coefficients to be determined. Following Vanella [73] and de Tullio [16], a linear basis array $\mathbf{p}^T(\mathbf{X}) = (1, x, y, z)$ is considered a cost-effective choice. Moreover, linear basis functions do represent the variation of the field variables up to the accuracy of the spatial discretization scheme implemented here. The coefficients $a_j(\mathbf{X})$ are obtained by minimizing a weighted, discrete L_2 norm as follows:

$$J = \sum_{k=1}^{n_e} W\left(\mathbf{X} - \mathbf{x}^k\right)\left[\mathbf{p}^T(\mathbf{x}^k)\,\mathbf{a}(\mathbf{x}^k) - \hat{q}_i^k\right]^2 , \tag{A.2}$$

where n_e is the number of cells within the interpolation stencil, and $\hat{q}_i^k$ is the Eulerian velocity at the node location $\mathbf{x}^k$. In the present application, the domain of influence, also called *support domain*, is limited to 27 cells, namely, the nearest neighbors of the cell containing the selected Lagrangian marker. $W\left(\mathbf{X} - \mathbf{x}^k\right)$ is a predefined shape function. Minimizing J with respect to $\mathbf{a}(\mathbf{X})$ leads to:

$$\mathbf{A}(\mathbf{X})\,\mathbf{a}(\mathbf{X}) = \mathbf{B}(\mathbf{X})\;\hat{\mathbf{q}}_i\,, \tag{A.3}$$

being $\hat{\mathbf{q}}_i = [\hat{q}_i^1\;\hat{q}_i^2\;...\;\hat{q}_i^e]^T$ an array containing the i-th velocity components collected in the support domain. Arrays $\mathbf{A}(\mathbf{X})$ and $\mathbf{B}(\mathbf{X})$ can be used to build the transfer operator $\mathbf{\Phi}(\mathbf{X})$:

$$\begin{aligned} \mathbf{A}(\mathbf{x}) &= \sum_{k=1}^{n_e} W\left(\mathbf{X} - \mathbf{x}^k\right)\mathbf{p}(\mathbf{x}^k)\mathbf{p}^T(\mathbf{x}^k)\,, \\ \mathbf{A}(\mathbf{x}) &= \left[W\left(\mathbf{X} - \mathbf{x}^1\right)\mathbf{p}(\mathbf{x}^1),\; W\left(\mathbf{X} - \mathbf{x}^2\right)\mathbf{p}(\mathbf{x}^2),\; ...,\; W\left(\mathbf{X} - \mathbf{x}^{n_e}\right)\mathbf{p}(\mathbf{x}^{n_e})\right]^T . \end{aligned} \tag{A.4}$$

Combining Eq. (A.4) and Eq. (A.1) one can get the final expression for the transfer operator:

$$\mathbf{\Phi}^T = \mathbf{p}^T(\mathbf{X})\,\mathbf{A}^{-1}(\mathbf{X})\,\mathbf{B}(\mathbf{X}) = \sum_{k=1}^{n_e} \Phi_k^l(\mathbf{X})\,\hat{q}_i^k\,. \tag{A.5}$$

In summary, the calculation of the MLS transfer operator requires the inversion of the square matrix $\mathbf{A}(\mathbf{X})$ at each Lagrangian marker. The size of the matrix $\mathbf{A}(\mathbf{X})$ depends on the size of the basis vector $\mathbf{p}(\mathbf{X})$: in the present application a 4 × 4 matrix is inverted by a Gaussian elimination algorithm.

The weight functions $W\left(\mathbf{X} - \mathbf{x}^k\right)$ play an important role in the performance of the method. They are positive, C^1 continuous, with unique solution for $\mathbf{a}(\mathbf{X})$ guaranteed in $\mathbf{X}$. They must me shaped to decrease in magnitude as the distance from $\mathbf{X}$ to $\mathbf{x}^k$ increases [200]. In the present work cubic splines are employed:

$$W\left(\mathbf{X} - \mathbf{x}^k\right) = \begin{cases} 2/3 - 4r_k^2 + 4r_k^3 & \text{for} \quad r_k \leq 0.5 \\ 4/3 - 4r_k + 4r_k^2 - 4/3r_k^3 & \text{for} \quad 0.5 < r_k \leq 1.0 \\ 0 & \text{for} \quad r_k > 1.0 \end{cases}, \tag{A.6}$$

where r_k is given by:

$$r_k = \frac{\left|\mathbf{X} - \mathbf{x}^k\right|}{r_i}, \tag{A.7}$$

with r_i the size of the support domain in the i-th direction. These cubic splines are monotonically decreasing and are sufficiently smooth in the support domain [80]. The resulting transfer functions reproduce exactly the linear polynomial contained in their basis and fulfill the properties of compatibility and partition of unity [201].

B Tensor algebra for the Active Strain approach

The base vectors follow the definition of reciprocal bases called *covariant* and *contravariant* and are denoted by subscript and superscript respectively. From the first fundamental form of surfaces:

$$\mathbf{g}^i \cdot \mathbf{g}^j = g^{ij}\,, \tag{B.1}$$

$$\mathbf{g}_i \cdot \mathbf{g}_j = g_{ij}\,, \tag{B.2}$$

where $g_{ij} = g_{ji}$, and $g^{ij} = g^{ji}$ for the symmetry of the metric tensor. Covariant and contravariant base vectors are related by the condition:

$$\mathbf{g}_i \cdot \mathbf{g}^j = \delta^j_i, \quad \mathbf{g}^i \cdot \mathbf{g}_j = \delta^i_j\,, \tag{B.3}$$

being δ the Kroneker delta. It follows that:

$$A_{ij}\delta^j_k = A_{ik}, \quad A^{ij}\delta^k_j = A^{ik}\,. \tag{B.4}$$

The change of basis between reciprocal vectors can be performed by:

$$\mathbf{g}^i = g^{ij}\mathbf{g}_j\,, \tag{B.5}$$

$$\mathbf{g}_i = g_{ij}\mathbf{g}^j\,. \tag{B.6}$$

The transpose and the inverse of a second order tensor $\mathbf{A} = A_{ij}\,\mathbf{g}^i \otimes \mathbf{g}^j = A^{ij}\,\mathbf{g}_i \otimes \mathbf{g}_j$ can be computed as:

$$\mathbf{A}^T = A_{ij}\,\mathbf{g}^j \otimes \mathbf{g}^i = A^{ij}\,\mathbf{g}_j \otimes \mathbf{g}_i\,, \tag{B.7}$$

$$\mathbf{A}^T = A_{ji}\,\mathbf{g}^i \otimes \mathbf{g}^j = A^{ji}\,\mathbf{g}_i \otimes \mathbf{g}_j\,, \tag{B.8}$$

$$\mathbf{A}^{-1} = \bar{A}^{ij}\,\mathbf{g}_i \otimes \mathbf{g}_j = \bar{A}_{ij}\,\mathbf{g}^i \otimes \mathbf{g}^j\,, \tag{B.9}$$

where $\left[\bar{A}^{ij}\right] = \left[A_{ij}\right]^{-1}$ and $\left[\bar{A}_{ij}\right] = \left[A^{ij}\right]^{-1}$.
The simple contraction between second order tensors is defined, following [202](1.57) and [172](1.2.7), as:

$$\begin{aligned}\mathbf{AB} &= \left(A^{ij}\,\mathbf{g}_i \otimes \mathbf{g}_j\right)\left(B_{kl}\,\mathbf{g}^k \otimes \mathbf{g}^l\right)\\ &= A^{ij}\,B_{kl}\,\left(\mathbf{g}_j \cdot \mathbf{g}^k\right)\left(\mathbf{g}_i \otimes \mathbf{g}^l\right)\\ &= A^{ij}\,B_{kl}\,\delta^k_j\,\left(\mathbf{g}_i \otimes \mathbf{g}^l\right)\\ &= A^{ik}\,B_{kl}\,\left(\mathbf{g}_i \otimes \mathbf{g}^l\right)\,.\end{aligned} \tag{B.10}$$

According to this definition, the simple contraction between a fourth-order tensor and a second-order tensor reads:

$$\begin{aligned}\mathbb{A}\mathbf{B} &= \left(A^{ijkl}\,\mathbf{g}_i\otimes\mathbf{g}_j\otimes\mathbf{g}_k\otimes\mathbf{g}_l\right)\left(A_{mn}\,\mathbf{g}^m\otimes\mathbf{g}^n\right)\\ &= A^{ijkl}\,B_{mn}\,\left(\mathbf{g}_l\cdot\mathbf{g}^m\right)\left(\mathbf{g}_i\otimes\mathbf{g}_j\otimes\mathbf{g}_k\otimes\mathbf{g}^n\right)\\ &= A^{ijkl}\,B_{mn}\,\delta_l^m\,\left(\mathbf{g}_i\otimes\mathbf{g}_j\otimes\mathbf{g}_k\otimes\mathbf{g}^n\right)\\ &= A^{ijkm}\,B_{mn}\,\left(\mathbf{g}_i\otimes\mathbf{g}_l\otimes\mathbf{g}_k\otimes\mathbf{g}^n\right).\end{aligned} \tag{B.11}$$

A double contraction of a fourth order tensor with a second order tensor (left mapping) is defined, according to [202](1.151), by:

$$\begin{aligned}\mathbb{A}:\mathbf{B} &= \left(A^{ijkl}\,\mathbf{g}_i\otimes\mathbf{g}_j\otimes\mathbf{g}_k\otimes\mathbf{g}_l\right)\left(B_{mn}\,\mathbf{g}^m\otimes\mathbf{g}^n\right)\\ &= A^{ijkl}\,B_{mn}\,\left(\mathbf{g}_k\cdot\mathbf{g}^m\right)\left(\mathbf{g}_l\cdot\mathbf{g}^n\right)\left(\mathbf{g}_i\otimes\mathbf{g}_j\right)\\ &= A^{ijkl}\,B_{mn}\,\delta_k^m\,\delta_l^n\,\left(\mathbf{g}_i\otimes\mathbf{g}_j\right)\\ &= A^{ijmn}\,B_{mn}\,\left(\mathbf{g}_i\otimes\mathbf{g}_j\right),\end{aligned} \tag{B.12}$$

whereas the double contraction between fourth-order tensors:

$$\begin{aligned}\mathbb{A}:\mathbb{B} &= \left(A^{ijab}\,\mathbf{g}_i\otimes\mathbf{g}_j\otimes\mathbf{g}_a\otimes\mathbf{g}_b\right)\left(B_{cdkl}\,\mathbf{g}^c\otimes\mathbf{g}^d\otimes\mathbf{g}^k\otimes\mathbf{g}^l\right)\\ &= A^{ijab}\,B_{cdkl}\,\left(\mathbf{g}_a\cdot\mathbf{g}^c\right)\left(\mathbf{g}_b\cdot\mathbf{g}^d\right)\left(\mathbf{g}_i\otimes\mathbf{g}_j\otimes\mathbf{g}^k\otimes\mathbf{g}^l\right)\\ &= A^{ijab}\,B_{cdkl}\,\delta_a^c\,\delta_b^d\,\left(\mathbf{g}_i\otimes\mathbf{g}_j\otimes\mathbf{g}^k\otimes\mathbf{g}^l\right)\\ &= A^{ijcd}\,B_{cdkl}\,\left(\mathbf{g}_i\otimes\mathbf{g}_j\otimes\mathbf{g}^k\otimes\mathbf{g}^l\right).\end{aligned} \tag{B.13}$$

The derivative of a tensor function with respect to a second order tensor [172](1.7.6) is computed via:

$$\begin{aligned}\frac{\partial\mathbf{A}}{\partial\mathbf{B}} &= \frac{\partial\left(A_{ij}\,\mathbf{g}^i\otimes\mathbf{g}^j\right)}{\partial\left(B_{kl}\,\mathbf{g}^k\otimes\mathbf{g}^l\right)}\\ &= \frac{\partial A_{ij}}{\partial B_{kl}}\,\left(\mathbf{g}^i\otimes\mathbf{g}^j\otimes\mathbf{g}_k\otimes\mathbf{g}_l\right)\\ &= \mathbb{C}_{ij}^{\cdot\cdot kl}\,\left(\mathbf{g}^i\otimes\mathbf{g}^j\otimes\mathbf{g}_k\otimes\mathbf{g}_l\right).\end{aligned} \tag{B.14}$$

The fourth-order identity tensor is defined here as the derivative of a tensor function with respect to itself. Consistently with the previous definition, and accordingly to [203] and Wikipedia (`https://en.wikipedia.org/wiki/Tensor_derivative_(continuum_mechanics)`), it is defined by:

$$\begin{aligned}\mathbb{I} = \frac{\partial\mathbf{A}}{\partial\mathbf{A}} &= \frac{\partial\left(A_{ij}\,\mathbf{g}^i\otimes\mathbf{g}^j\right)}{\partial\left(A_{kl}\,\mathbf{g}^k\otimes\mathbf{g}^l\right)}\\ &= \frac{\partial A_{ij}}{\partial A_{kl}}\,\left(\mathbf{g}^i\otimes\mathbf{g}^j\otimes\mathbf{g}_k\otimes\mathbf{g}_l\right)\\ &= g^{ik}\,g^{jl}\,\left(\mathbf{g}_i\otimes\mathbf{g}_j\otimes\mathbf{g}_k\otimes\mathbf{g}_l\right),\end{aligned} \tag{B.15}$$

if the tensor **A** is not symmetric. If **A** is symmetric instead:

$$\mathbb{I}^s = \frac{\partial \mathbf{A}}{\partial \mathbf{A}} = \frac{1}{2}\left(g^{ik}\, g^{jl} + g^{il}\, g^{jk}\right)\left(\mathbf{g}_i \otimes \mathbf{g}_j \otimes \mathbf{g}_k \otimes \mathbf{g}_l\right) . \tag{B.16}$$

Bibliography

[1] D Brown, P Messina, et al. *Scientific grand challenges: Crosscutting technologies for computing at the exascale. Office of Science, US Department of Energy*. 2010.

[2] David E Keyes, Lois C McInnes, Carol Woodward, William Gropp, Eric Myra, Michael Pernice, John Bell, Jed Brown, Alain Clo, Jeffrey Connors, et al. “Multiphysics simulations: Challenges and opportunities”. In: *The International Journal of High Performance Computing Applications* 27.1 (2013), pp. 4–83.

[3] J Austin Cottrell, Thomas JR Hughes, and Yuri Bazilevs. *Isogeometric analysis: toward integration of CAD and FEA*. John Wiley & Sons, 2009.

[4] J Austin Cottrell, Alessandro Reali, Yuri Bazilevs, and Thomas JR Hughes. “Isogeometric analysis of structural vibrations”. In: *Computer methods in applied mechanics and engineering* 195.41-43 (2006), pp. 5257–5296.

[5] JA Cottrell, TJR Hughes, and A Reali. “Studies of refinement and continuity in isogeometric structural analysis”. In: *Computer methods in applied mechanics and engineering* 196.41-44 (2007), pp. 4160–4183.

[6] Thomas JR Hughes, John A Cottrell, and Yuri Bazilevs. “Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement”. In: *Computer methods in applied mechanics and engineering* 194.39-41 (2005), pp. 4135–4195.

[7] Hugo Casquero, Carles Bona-Casas, and Hector Gomez. “A NURBS-based immersed methodology for fluid–structure interaction”. In: *Computer Methods in Applied Mechanics and Engineering* 284 (2015), pp. 943–970.

[8] Josef Kiendl, K-U Bletzinger, Johannes Linhard, and Roland Wüchner. “Isogeometric shell analysis with Kirchhoff–Love elements”. In: *Computer Methods in Applied Mechanics and Engineering* 198.49-52 (2009), pp. 3902–3914.

[9] Josef Kiendl. “Isogeometric analysis and shape optimal design of shell structures”. PhD thesis. Technische Universität München, 2011.

[10] J. H. Ferziger and M. Peric. *Computational methods for fluid dynamics*. Third. Springer, 2002.

[11] P Moin and R Verzicco. “On the suitability of second-order accurate discretizations for turbulent flow simulations”. In: *European Journal of Mechanics-B/Fluids* 55 (2016), pp. 242–245.

[12] Paolo Orlandi. *Fluid flow phenomena: a numerical toolkit*. Vol. 55. Springer Science & Business Media, 2012.

[13] Iman Borazjani, Liang Ge, and Fotis Sotiropoulos. “Curvilinear immersed boundary method for simulating fluid structure interaction with complex 3D rigid bodies”. In: *Journal of Computational physics* 227.16 (2008), pp. 7587–7620.

[14] Anvar Gilmanov, Trung Bao Le, and Fotis Sotiropoulos. “A numerical approach for simulating fluid structure interaction of flexible thin shells undergoing arbitrarily large deformations in complex domains”. In: *Journal of computational physics* 300 (2015), pp. 814–843.

[15] Julien Favier, Alistair Revell, and Alfredo Pinelli. “A Lattice Boltzmann–Immersed Boundary method to simulate the fluid interaction with moving and slender flexible objects”. In: *Journal of Computational Physics* 261 (2014), pp. 145–161.

[16] Marco D de Tullio and Giuseppe Pascazio. “A moving-least-squares immersed boundary method for simulating the fluid–structure interaction of elastic bodies with arbitrary thickness”. In: *Journal of Computational Physics* 325 (2016), pp. 201–225.

[17] Paola Causin, Jean-Frédéric Gerbeau, and Fabio Nobile. “Added-mass effect in the design of partitioned algorithms for fluid–structure problems”. In: *Computer methods in applied mechanics and engineering* 194.42-44 (2005), pp. 4506–4527.

[18] Christiane Förster, Wolfgang A Wall, and Ekkehard Ramm. “Artificial added mass instabilities in sequential staggered coupling of nonlinear structures and incompressible viscous flows”. In: *Computer methods in applied mechanics and engineering* 196.7 (2007), pp. 1278–1293.

[19] Joris Degroote, Peter Bruggeman, Robby Haelterman, and Jan Vierendeels. “Stability of a coupling technique for partitioned solvers in FSI applications”. In: *Computers & Structures* 86.23-24 (2008), pp. 2224–2234.

[20] Takashi Ashihara, Jason Constantino, and Natalia A Trayanova. “Tunnel propagation of postshock activations as a hypothesis for fibrillation induction and isoelectric window”. In: *Circulation research* 102.6 (2008), pp. 737–745.

[21] Viorel Mihalef, Razvan Ioan Ionasec, Puneet Sharma, Bogdan Georgescu, Ingmar Voigt, Michael Suehling, and Dorin Comaniciu. “Patient-specific modelling of whole heart anatomy, dynamics and haemodynamics from four-dimensional cardiac CT images”. In: *Interface Focus* 1.3 (2011), pp. 286–296.

[22] Adam W Feinberg, Alex Feigel, Sergey S Shevkoplyas, Sean Sheehy, George M Whitesides, and Kevin Kit Parker. “Muscular thin films for building actuators and powering devices”. In: *Science* 317.5843 (2007), pp. 1366–1370.

[23] Davide Ambrosi, Gianni Arioli, Fabio Nobile, and Alfio Quarteroni. “Electromechanical coupling in cardiac dynamics: the active strain approach”. In: *SIAM Journal on Applied Mathematics* 71.2 (2011), pp. 605–621.

[24] Fabio Nobile, Alfio Quarteroni, and Ricardo Ruiz-Baier. "An active strain electromechanical model for cardiac tissue". In: *International journal for numerical methods in biomedical engineering* 28.1 (2012), pp. 52–71.

[25] Thomas JR Hughes, John A Evans, and Alessandro Reali. "Finite element and NURBS approximations of eigenvalue, boundary-value, and initial-value problems". In: *Computer Methods in Applied Mechanics and Engineering* 272 (2014), pp. 290–320.

[26] RL Taylor. "Isogeometric analysis of nearly incompressible solids". In: *International Journal for Numerical Methods in Engineering* 87.1-5 (2011), pp. 273–288.

[27] F Auricchio, M Conti, M Ferraro, S Morganti, A Reali, and RL Taylor. "Innovative and efficient stent flexibility simulations based on isogeometric analysis". In: *Computer Methods in Applied Mechanics and Engineering* 295 (2015), pp. 347–361.

[28] Josef Kiendl, Ming-Chen Hsu, Michael CH Wu, and Alessandro Reali. "Isogeometric Kirchhoff–Love shell formulations for general hyperelastic materials". In: *Computer Methods in Applied Mechanics and Engineering* 291 (2015), pp. 280–303.

[29] Yuri Bazilevs, Victor M Calo, Thomas JR Hughes, and Yongjie Zhang. "Isogeometric fluid-structure interaction: theory, algorithms, and computations". In: *Computational mechanics* 43.1 (2008), pp. 3–37.

[30] Les Piegl and Wayne Tiller. *The NURBS book*. Springer Science & Business Media, 2012.

[31] J Kiendl, Y Bazilevs, M-C Hsu, R Wüchner, and K-U Bletzinger. "The bending strip method for isogeometric analysis of Kirchhoff–Love shell structures comprised of multiple patches". In: *Computer Methods in Applied Mechanics and Engineering* 199.37-40 (2010), pp. 2403–2416.

[32] Martin Ruess, Dominik Schillinger, Ali I Oezcan, and Ernst Rank. "Weak coupling for isogeometric analysis of non-matching and trimmed multi-patch geometries". In: *Computer Methods in Applied Mechanics and Engineering* 269 (2014), pp. 46–71.

[33] Laurens Coox, Francesco Greco, Onur Atak, Dirk Vandepitte, and Wim Desmet. "A robust patch coupling method for NURBS-based isogeometric analysis of non-conforming multipatch surfaces". In: *Computer Methods in Applied Mechanics and Engineering* 316 (2017), pp. 235–260.

[34] Yuri Bazilevs, Victor M Calo, John A Cottrell, John A Evans, Thomas Jr R Hughes, S Lipton, Michael A Scott, and Thomas W Sederberg. "Isogeometric analysis using T-splines". In: *Computer Methods in Applied Mechanics and Engineering* 199.5-8 (2010), pp. 229–263.

[35] E. Stein, R. de Borst, and T. J. R. Hughes. *Encyclopedia of computational mechanics. Volume 2: Solids and Structures*. Springer, 2004.

[36] Yavuz Basar and Wilfried B Krätzig. "Theory of shell structures". In: *Fortschritts-Berichte VDI, series* 18 (2000).

[37] Ralph Echter. “Isogeometric analysis of shells”. In: (2013).

[38] Dieter Jaeger and Ranu Jung. *Encyclopedia of computational neuroscience*. Springer Publishing Company, Incorporated, 2015.

[39] Thomas JR Hughes, Alessandro Reali, and Giancarlo Sangalli. “Efficient quadrature for NURBS-based isogeometric analysis”. In: *Computer methods in applied mechanics and engineering* 199.5-8 (2010), pp. 301–313.

[40] Jintai Chung and GM Hulbert. “A time integration algorithm for structural dynamics with improved numerical dissipation: the generalized-α method”. In: *Journal of applied mechanics* 60.2 (1993), pp. 371–375.

[41] Yuri Bazilevs, Kenji Takizawa, and Tayfun E Tezduyar. *Computational fluid-structure interaction: methods and applications*. John Wiley & Sons, 2013.

[42] David Kamensky, Ming-Chen Hsu, Dominik Schillinger, John A Evans, Ankush Aggarwal, Yuri Bazilevs, Michael S Sacks, and Thomas JR Hughes. “An immersogeometric variational framework for fluid–structure interaction: Application to bioprosthetic heart valves”. In: *Computer methods in applied mechanics and engineering* 284 (2015), pp. 1005–1053.

[43] Woojin Kim, Injae Lee, and Haecheon Choi. “A weak-coupling immersed boundary method for fluid–structure interaction with low density ratio of solid to fluid”. In: *Journal of Computational Physics* 359 (2018), pp. 296–311.

[44] Klaus-Jürgen Bathe. *Finite element procedures*. Klaus-Jurgen Bathe, 2006.

[45] Rajat Mittal and Gianluca Iaccarino. “Immersed boundary methods”. In: *Annu. Rev. Fluid Mech.* 37 (2005), pp. 239–261.

[46] J.E. Welch, F.H. Harlow, J.P. Shannon, and B.J. Daly. *The MAC method: a computing technique for solving viscous, incompressible, transient fluid-flow problems involving free surfaces*. Tech. rep. Los Alamos Scientific Lab., Univ. of California, Nov. 1965.

[47] Yuanxun Bao, Aleksandar Donev, Boyce E Griffith, David M McQueen, and Charles S Peskin. “An Immersed Boundary method with divergence-free velocity interpolation and force spreading”. In: *Journal of computational physics* 347 (2017), pp. 183–206.

[48] Man Mohan Rai and Parviz Moin. “Direct simulations of turbulent flow using finite-difference schemes”. In: *Journal of computational physics* 96.1 (1991), pp. 15–53.

[49] Alexandre Joel Chorin. “On the convergence of discrete approximations to the Navier-Stokes equations”. In: *Mathematics of computation* 23.106 (1969), pp. 341–353.

[50] Parviz Moin and John Kim. “Numerical investigation of turbulent channel flow”. In: *Journal of fluid mechanics* 118 (1982), pp. 341–377.

[51] Jungwoo Kim, Dongjoo Kim, and Haecheon Choi. “An Immersed-Boundary Finite-Volume Method for Simulations of Flow in Complex Geometries”. In: *Journal of Computational Physics* 171.1 (2001), pp. 132–150. DOI: `10.1006/jcph.2001.6778`.

[52] Robert D Guy and David A Hartenstine. "On the accuracy of direct forcing immersed boundary methods with projection methods". In: *Journal of Computational Physics* 229.7 (2010), pp. 2479–2496.

[53] Jim Douglas and James E Gunn. "A general formulation of alternating direction methods". In: *Numerische Mathematik* 6.1 (1964), pp. 428–453.

[54] Eli Turkel. "Preconditioning techniques in computational fluid dynamics". In: *Annual Review of Fluid Mechanics* 31.1 (1999), pp. 385–416.

[55] Alfio Quarteroni, Riccardo Sacco, and Fausto Saleri. "Foundations of Matrix Analysis". In: *Numerical Mathematics*. Springer, 2007, pp. 1–32.

[56] Richard M Beam and Robert F Warming. "An implicit finite-difference algorithm for hyperbolic systems in conservation-law form". In: *Journal of computational physics* 22.1 (1976), pp. 87–110.

[57] Dale B Haidvogel and Thomas Zang. "The accurate solution of Poisson's equation by expansion in Chebyshev polynomials". In: *Journal of Computational Physics* 30.2 (1979), pp. 167–180.

[58] Billy L Buzbee, Gene H Golub, and Clair W Nielson. "On direct methods for solving Poisson equation". In: *SIAM Journal on Numerical analysis* 7.4 (1970), pp. 627–656.

[59] Fady M. Najjar and S. P. Vanka. "Simulations of the unsteady separated flow past a normal flat plate". In: *International Journal for Numerical Methods in Fluids* 21.7 (1995), pp. 525–547. ISSN: 10970363. DOI: `10.1002/fld.1650210702`.

[60] R. Verzicco and P. Orlandi. "A finite-difference scheme for three-dimensional incompressible flows in cylindrical coordinates". In: *Journal of Computational Physics* (1996). ISSN: 00219991. DOI: `10.1006/jcph.1996.0033`.

[61] I_ Orlanski. "A simple boundary condition for unbounded hyperbolic flows". In: *Journal of computational physics* 21.3 (1976), pp. 251–269.

[62] R Kahawita and P Wang. "Numerical simulation of the wake flow behind trapezoidal bluff bodies". In: *Computers & fluids* 31.1 (2002), pp. 99–112.

[63] M Kiya and M Matsumura. "Incoherent turbulence structure in the near wake of a normal plate". In: *Journal of Fluid Mechanics* 190 (1988), pp. 343–356.

[64] Woojin Kim and Haecheon Choi. "Immersed boundary methods for fluid-structure interaction: A review". In: *International Journal of Heat and Fluid Flow* 75 (2019), pp. 301–309.

[65] Andrew A Johnson and Tayfun E Tezduyar. "Parallel computation of incompressible flows with complex geometries". In: *International Journal for Numerical Methods in Fluids* 24.12 (1997), pp. 1321–1340.

[66] Tayfun E Tezduyar and Sunil Sathe. “Modelling of fluid–structure interactions with the space–time finite elements: solution techniques”. In: *International Journal for Numerical Methods in Fluids* 54.6-8 (2007), pp. 855–900.

[67] Charles S Peskin. “Flow patterns around heart valves: a numerical method”. In: *Journal of computational physics* 10.2 (1972), pp. 252–271.

[68] Yoichiro Mori and Charles S Peskin. “Implicit second-order immersed boundary methods with boundary mass”. In: *Computer methods in applied mechanics and engineering* 197.25-28 (2008), pp. 2049–2067.

[69] Yongsam Kim and Charles S Peskin. “Penalty immersed boundary method for an elastic boundary with mass”. In: *Physics of Fluids* 19.5 (2007), p. 053103.

[70] Haoxiang Luo, Rajat Mittal, Xudong Zheng, Steven A Bielamowicz, Raymond J Walsh, and James K Hahn. “An immersed-boundary method for flow–structure interaction in biological systems with application to phonation”. In: *Journal of computational physics* 227.22 (2008), pp. 9303–9332.

[71] Xiaolei Yang, Xing Zhang, Zhilin Li, and Guo-Wei He. “A smoothing technique for discrete delta functions with application to immersed boundary method in moving boundary simulations”. In: *Journal of Computational Physics* 228.20 (2009), pp. 7821–7836.

[72] Markus Uhlmann. “An immersed boundary method with direct forcing for the simulation of particulate flows”. In: *Journal of Computational Physics* 209.2 (2005), pp. 448–476.

[73] Marcos Vanella and Elias Balaras. “A moving-least-squares reconstruction for embedded-boundary formulations”. In: *Journal of Computational Physics* 228.18 (2009), pp. 6617–6628.

[74] Wim-Paul Breugem. “A second-order accurate immersed boundary method for fully resolved simulations of particle-laden flows”. In: *Journal of Computational Physics* 231.13 (2012), pp. 4469–4498.

[75] Tobias Kempe and Jochen Fröhlich. “An improved immersed boundary method with direct forcing for the simulation of particle laden flows”. In: *Journal of Computational Physics* 231.9 (2012), pp. 3663–3684.

[76] EA Fadlun, Roberto Verzicco, Paolo Orlandi, and J Mohd-Yusof. “Combined immersed-boundary finite-difference methods for three-dimensional complex flow simulations”. In: *Journal of computational physics* 161.1 (2000), pp. 35–60.

[77] Gianluca Iaccarino and Roberto Verzicco. “Immersed boundary technique for turbulent flow simulations”. In: *Appl. Mech. Rev.* 56.3 (2003), pp. 331–347.

[78] Jung Hee Seo and Rajat Mittal. “A sharp-interface immersed boundary method with improved mass conservation and reduced spurious pressure oscillations”. In: *Journal of computational physics* 230.19 (2011), pp. 7347–7363.

[79] Stephan Schwarz, Tobias Kempe, and Jochen Fröhlich. “A temporal discretization scheme to compute the motion of light particles in viscous flows by an immersed boundary method”. In: *Journal of Computational Physics* 281 (2015), pp. 591–613.

[80] Marcos Vanella, Patrick Rabenold, and Elias Balaras. “A direct-forcing embedded-boundary method with adaptive mesh refinement for fluid–structure interaction problems”. In: *Journal of Computational Physics* 229.18 (2010), pp. 6427–6449.

[81] Dionysios Angelidis, Saurabh Chawdhary, and Fotis Sotiropoulos. “Unstructured Cartesian refinement with sharp interface immersed boundary method for 3D unsteady incompressible flows”. In: *Journal of Computational Physics* 325 (2016), pp. 272–300.

[82] Antonio Posa, Marcos Vanella, and Elias Balaras. “An adaptive reconstruction for Lagrangian, direct-forcing, immersed-boundary methods”. In: *Journal of Computational Physics* 351 (2017), pp. 422–436.

[83] Antonio Cristallo and Roberto Verzicco. “Combined immersed boundary/large-eddy-simulations of incompressible three dimensional complex flows”. In: *Flow, turbulence and combustion* 77.1-4 (2006), pp. 3–26.

[84] Jongho Lee, Jungwoo Kim, Haecheon Choi, and Kyung-Soo Yang. “Sources of spurious force oscillations from an immersed boundary method for moving-body problems”. In: *Journal of computational physics* 230.7 (2011), pp. 2677–2695.

[85] Haoxiang Luo, Hu Dai, Paulo JSA Ferreira de Sousa, and Bo Yin. “On the numerical oscillation of the direct-forcing immersed-boundary method for moving boundaries”. In: *Computers & Fluids* 56 (2012), pp. 61–76.

[86] Wei-Xi Huang, Soo Jai Shin, and Hyung Jin Sung. “Simulation of flexible filaments in a uniform flow by the immersed boundary method”. In: *Journal of Computational Physics* 226.2 (2007), pp. 2206–2228.

[87] Alfredo Pinelli, IZ Naqavi, Ugo Piomelli, and Julien Favier. “Immersed-boundary methods for general finite-difference and finite-volume Navier–Stokes solvers”. In: *Journal of Computational Physics* 229.24 (2010), pp. 9073–9091.

[88] Vamsi Spandan, Valentina Meschini, Rodolfo Ostilla-Mónico, Detlef Lohse, Giorgio Querzoli, Marco D de Tullio, and Roberto Verzicco. “A parallel interaction potential approach coupled with the immersed boundary method for fully resolved simulations of deformable interfaces and membranes”. In: *Journal of computational physics* 348 (2017), pp. 567–590.

[89] Diogo MC Martins, Duarte MS Albuquerque, and José CF Pereira. “Continuity constrained least-squares interpolation for SFO suppression in immersed boundary methods”. In: *Journal of Computational Physics* 336 (2017), pp. 608–626.

[90] Juwon Jang and Changhoon Lee. “An immersed boundary method for nonuniform grids”. In: *Journal of Computational Physics* 341 (2017), pp. 1–12.

[91] Serge Piperno, Charbel Farhat, and Bernard Larrouturou. “Partitioned procedures for the transient solution of coupled aroelastic problems Part I: Model problem, theory and two-dimensional application”. In: *Computer methods in applied mechanics and engineering* 124.1-2 (1995), pp. 79–112.

[92] EH Van Brummelen. “Added mass effects of compressible and incompressible flows in fluid-structure interaction”. In: *Journal of Applied mechanics* 76.2 (2009), p. 021206.

[93] Giovanna Guidoboni, Roland Glowinski, Nicola Cavallini, and Suncica Canic. “Stable loosely-coupled-type algorithm for fluid–structure interaction in blood flow”. In: *Journal of Computational Physics* 228.18 (2009), pp. 6916–6937.

[94] Jianming Yang and Frederick Stern. “A non-iterative direct forcing immersed boundary method for strongly-coupled fluid–solid interactions”. In: *Journal of Computational Physics* 295 (2015), pp. 779–804.

[95] Bruce M Irons and Robert C Tuck. “A version of the Aitken accelerator for computer iteration”. In: *International Journal for Numerical Methods in Engineering* 1.3 (1969), pp. 275–277.

[96] Ulrich Küttler and Wolfgang A Wall. “Fixed-point fluid–structure interaction solvers with dynamic relaxation”. In: *Computational mechanics* 43.1 (2008), pp. 61–72.

[97] Ivo Babuska and J Tinsley Oden. “Verification and validation in computational engineering and science: basic concepts”. In: *Computer methods in applied mechanics and engineering* 193.36 (2004), pp. 4057–4066.

[98] Sadatoshi Taneda. “Experimental investigation of the wake behind a sphere at low Reynolds numbers”. In: *Journal of the Physical Society of Japan* 11.10 (1956), pp. 1104–1108.

[99] TA Johnson and VC Patel. “Flow past a sphere up to a Reynolds number of 300”. In: *Journal of Fluid Mechanics* 378 (1999), pp. 19–70.

[100] Bengt Fornberg. “Steady viscous flow past a sphere at high Reynolds numbers”. In: *Journal of Fluid Mechanics* 190 (1988), pp. 471–489.

[101] Pedro Costa, Bendiks Jan Boersma, Jerry Westerweel, and Wim-Paul Breugem. “Collision model for fully resolved simulations of flows laden with finite-size particles”. In: *Physical Review E* 92.5 (2015), p. 053012.

[102] P Gondret, M Lance, and L Petit. “Bouncing motion of spherical particles in fluids”. In: *Physics of fluids* 14.2 (2002), pp. 643–652.

[103] I Eames and SB Dalziel. “Dust resuspension by the flow around an impacting sphere”. In: *J Fluid Mech* 403 (2000), pp. 305–328.

[104] Luca Heltai, Josef Kiendl, Antonio DeSimone, and Alessandro Reali. “A natural framework for isogeometric fluid–structure interaction based on BEM–shell coupling”. In: *Computer methods in applied mechanics and engineering* 316 (2017), pp. 522–546.

[105] Wei-Xi Huang and Hyung Jin Sung. “Three-dimensional simulation of a flapping flag in a uniform flow”. In: *Journal of Fluid Mechanics* 653 (2010), pp. 301–336.

[106] Injae Lee and Haecheon Choi. “A discrete-forcing immersed boundary method for the fluid–structure interaction of an elastic slender body”. In: *Journal of Computational Physics* 280 (2015), pp. 529–546.

[107] Fang-Bao Tian, Hu Dai, Haoxiang Luo, James F Doyle, and Bernard Rousseau. “Fluid–structure interaction involving large deformations: 3D simulations and applications to biological systems”. In: *Journal of computational physics* 258 (2014), pp. 451–469.

[108] Patrick J Roache. “Perspective: a method for uniform reporting of grid refinement studies”. In: (1994).

[109] Charles S Peskin and Beth Feller Printz. “Improved volume conservation in the computation of flows with immersed elastic boundaries”. In: *Journal of computational physics* 105.1 (1993), pp. 33–46.

[110] Ricardo Cortez and Michael Minion. “The blob projection method for immersed boundary problems”. In: *Journal of Computational Physics* 161.2 (2000), pp. 428–453.

[111] Boyce E Griffith. “On the volume conservation of the immersed boundary method”. In: *Communications in Computational Physics* 12.2 (2012), pp. 401–432.

[112] Daegyoum Kim, Julia Cossé, Cecilia Huertas Cerdeira, and Morteza Gharib. “Flapping dynamics of an inverted flag”. In: *Journal of Fluid Mechanics* 736 (2013), pp. 1–12. ISSN: 00221120. DOI: `10.1017/jfm.2013.555`.

[113] Kourosh Shoele and Rajat Mittal. “Energy harvesting by flow-induced flutter in a simple model of an inverted piezoelectric flag”. In: *J. Fluid Mech.* 790 (2016), pp. 582–606. ISSN: 14697645. DOI: `10.1017/jfm.2016.40`.

[114] Chao Tang, Nan Sheng Liu, and Xi Yun Lu. “Dynamics of an inverted flexible plate in a uniform flow”. In: *Phys. Fluids* 27.7 (2015). ISSN: 10897666. DOI: `10.1063/1.4923281`.

[115] Zhongwang Dou, Aaron Rips, Lauren Jacob, and Rajat Mittal. “Experimental Characterization of the Flow-Induced Flutter of a Suspended Elastic Membrane”. In: *AIAA Journal* (2019).

[116] Jaeha Ryu, Sung Goon Park, Boyoung Kim, and Hyung Jin Sung. “Flapping dynamics of an inverted flag in a uniform flow”. In: *Journal of Fluids and Structures* 57 (2015), pp. 159–169.

[117] M Glück, M Breuer, F Durst, A Halfmann, and E Rank. “Computation of wind-induced vibrations of flexible shells and membranous structures”. In: *Journal of Fluids and Structures* 17.5 (2003), pp. 739–765.

[118] Mitul Luhar and Heidi M Nepf. “Flow-induced reconfiguration of buoyant and flexible aquatic vegetation”. In: *Limnology and Oceanography* 56.6 (2011), pp. 2003–2017.

[119] NH Lovell, S Dokos, E Cheng, and GJ Suaning. “Simulation of parallel current injection for use in a vision prosthesis”. In: *Conference Proceedings. 2nd International IEEE EMBS Conference on Neural Engineering, 2005.* IEEE. 2005, pp. 458–461.

[120] Dalin Tang, Chun Yang, Tal Geva, and J Pedro. “Image-based patient-specific ventricle models with fluid–structure interaction for cardiac function assessment and surgical design optimization”. In: *Progress in Pediatric Cardiology* 30.1-2 (2010), pp. 51–62.

[121] Steven A Niederer, Eric Kerfoot, Alan P Benson, Miguel O Bernabeu, Olivier Bernus, Chris Bradley, Elizabeth M Cherry, Richard Clayton, Flavio H Fenton, Alan Garny, et al. “Verification of cardiac tissue electrophysiology simulators using an N-version benchmark”. In: *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 369.1954 (2011), pp. 4331–4351.

[122] Alan Lloyd Hodgkin and Andrew Fielding Huxley. “Propagation of electrical signals along giant nerve fibres”. In: *Proceedings of the Royal Society of London. Series B-Biological Sciences* 140.899 (1952), pp. 177–183.

[123] Flavio H Fenton and Elizabeth M Cherry. “Models of cardiac cell”. In: *Scholarpedia* 3.8 (2008), p. 1868.

[124] Alfio Quarteroni, Toni Lassila, Simone Rossi, and Ricardo Ruiz-Baier. “Integrated Heart—Coupling multiscale and multiphysics models for the simulation of the cardiac function”. In: *Computer Methods in Applied Mechanics and Engineering* 314 (2017), pp. 345–407.

[125] Janna C Nawroth, Hyungsuk Lee, Adam W Feinberg, Crystal M Ripplinger, Megan L McCain, Anna Grosberg, John O Dabiri, and Kevin Kit Parker. “A tissue-engineered jellyfish with biomimetic propulsion”. In: *Nature biotechnology* 30.8 (2012), p. 792.

[126] Go W Beeler and H Reuter. “Reconstruction of the action potential of ventricular myocardial fibres”. In: *The Journal of physiology* 268.1 (1977), pp. 177–210.

[127] Ching-hsing Luo and Yoram Rudy. “A dynamic model of the cardiac ventricular action potential. I. Simulations of ionic currents and concentration changes.” In: *Circulation research* 74.6 (1994), pp. 1071–1096.

[128] Denis Noble. “A modification of the Hodgkin—Huxley equations applicable to Purkinje fibre action and pacemaker potentials”. In: *The Journal of physiology* 160.2 (1962), p. 317.

[129] Richard FitzHugh. “Impulses and physiological states in theoretical models of nerve membrane”. In: *Biophysical journal* 1.6 (1961), pp. 445–466.

[130] Jinichi Nagumo, Suguru Arimoto, and Shuji Yoshizawa. “An active pulse transmission line simulating nerve axon”. In: *Proceedings of the IRE* 50.10 (1962), pp. 2061–2070.

[131] Flavio Fenton and Alain Karma. “Vortex dynamics in three-dimensional continuous myocardium with fiber rotation: Filament instability and fibrillation”. In: *Chaos: An Interdisciplinary Journal of Nonlinear Science* 8.1 (1998), pp. 20–47.

[132] James P Keener and James Sneyd. *Mathematical physiology*. Vol. 1. Springer, 1998.

[133] Wolfram-Hubertus Zimmermann, Michael Didié, Stephan Döker, Ivan Melnychenko, Hiroshi Naito, Christina Rogge, Malte Tiburcy, and Thomas Eschenhagen. “Heart muscle engineering: an update on cardiac muscle replacement therapy”. In: *Cardiovascular research* 71.3 (2006), pp. 419–429.

[134] Von Howard Ebron, Zhiwei Yang, Daniel J Seyer, Mikhail E Kozlov, Jiyoung Oh, Hui Xie, Joselito Razal, Lee J Hall, John P Ferraris, Alan G MacDiarmid, et al. “Fuel-powered artificial muscles”. In: *Science* 311.5767 (2006), pp. 1580–1583.

[135] John DW Madden, Nathan A Vandesteeg, Patrick A Anquetil, Peter GA Madden, Arash Takshi, Rachel Z Pytel, Serge R Lafontaine, Paul A Wieringa, and Ian W Hunter. “Artificial muscle technology: physical principles and naval prospects”. In: *IEEE Journal of oceanic engineering* 29.3 (2004), pp. 706–728.

[136] John O Dabiri. “Landmarks and frontiers in biological fluid dynamics”. In: *Physical Review Fluids* 4.11 (2019), p. 110501.

[137] Kenneth V Kardong et al. *Vertebrates: comparative anatomy, function, evolution*. QL805 K35 2006. McGraw-Hill New York, 2006.

[138] William J Krause. *Krause’s essential human histology for medical students*. Universal-Publishers, 2005.

[139] GW Jenkins, CP Kemnitz, GJ Tortora, and GJ Tortora. *Anatomy and physiology: from science to life. 2, International udent version ed. Hoboken*.

[140] Piero Colli Franzone, Luca Franco Pavarino, and Simone Scacchi. *Mathematical cardiac electrophysiology*. Vol. 13. Springer, 2014.

[141] Richard Klabunde. *Cardiovascular physiology concepts*. Lippincott Williams & Wilkins, 2011.

[142] Sebastian Brandstaeter, Alessio Gizzi, Sebastian L Fuchs, Amadeus M Gebauer, Roland C Aydin, and Christian J Cyron. “Computational model of gastric motility with active-strain electromechanics”. In: *ZAMM-Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik* 98.12 (2018), pp. 2177–2197.

[143] Gary Tse, Eric TH Lai, Alex PW Lee, Bryan P Yan, and Sunny H Wong. “Electrophysiological mechanisms of gastrointestinal arrhythmogenesis: lessons from the heart”. In: *Frontiers in physiology* 7 (2016), p. 230.

[144] Alessandro Lucantonio, Paola Nardinocchi, Matteo Pezzulla, and Luciano Teresi. “Multiphysics of bio-hybrid systems: shape control and electro-induced motion”. In: *Smart materials and structures* 23.4 (2014), p. 045043.

[145] Fabian Pallasdies, Sven Goedeke, Wilhelm Braun, and Raoul-Martin Memmesheimer. “From Single Neurons to Behavior in the Jellyfish Aurelia aurita”. In: *arXiv preprint arXiv:1907.05060* (2019).

[146] Joakim Sundnes, Glenn Terje Lines, Xing Cai, Bjørn Frederik Nielsen, Kent-Andre Mardal, and Aslak Tveito. *Computing the electrical activity in the heart*. Vol. 1. Springer Science & Business Media, 2007.

[147] Alan L Hodgkin and Andrew F Huxley. “A quantitative description of membrane current and its application to conduction and excitation in nerve”. In: *The Journal of physiology* 117.4 (1952), pp. 500–544.

[148] Joakim Sundnes, Glenn Terje Lines, Xing Cai, Bjørn Frederik Nielsen, Kent-Andre Mardal, and Aslak Tveito. “On the computational complexity of the bidomain and the monodomain models of electrophysiology”. In: *Annals of biomedical engineering* 34.7 (2006), pp. 1088–1097.

[149] Leslie Tung. “A bi-domain model for describing ischemic myocardial dc potentials.” PhD thesis. Massachusetts Institute of Technology, 1978.

[150] EJ Vigmond, R Weber Dos Santos, AJ Prassl, M Deo, and G Plank. “Solvers for the cardiac bidomain equations”. In: *Progress in biophysics and molecular biology* 96.1-3 (2008), pp. 3–18.

[151] JC Neu and W Krassowska. “Homogenization of syncytial tissues.” In: *Critical reviews in biomedical engineering* 21.2 (1993), pp. 137–199.

[152] Piero Colli-Franzone, Luca F Pavarino, and Simone Scacchi. “Mathematical and numerical methods for reaction-diffusion models in electrocardiology”. In: *Modeling of Physiological flows*. Springer, 2012, pp. 107–141.

[153] Kirill Skouibine and Wanda Krassowska. “Increasing the computational efficiency of a bidomain model of defibrillation using a time-dependent activating function”. In: *Annals of Biomedical Engineering* 28.7 (2000), pp. 772–780.

[154] David E Clapham and Louis J DeFelice. “Small signal impedance of heart cell membranes”. In: *The Journal of membrane biology* 67.1 (1982), pp. 63–71.

[155] Simone Rossi and Boyce E Griffith. “Incorporating inductances in tissue-scale models of cardiac electrophysiology”. In: *Chaos: An Interdisciplinary Journal of Nonlinear Science* 27.9 (2017), p. 093926.

[156] A Gizzi, A Loppini, R Ruiz-Baier, A Ippolito, A Camassa, A La Camera, E Emmi, L Di Perna, V Garofalo, C Cherubini, et al. “Nonlinear diffusion and thermo-electric coupling in a two-variable model of cardiac action potential”. In: *Chaos: An Interdisciplinary Journal of Nonlinear Science* 27.9 (2017), p. 093919.

[157] Rubin R Aliev and Alexander V Panfilov. “A simple two-variable model of cardiac excitation”. In: *Chaos, Solitons & Fractals* 7.3 (1996), pp. 293–301.

[158] Alfonso Bueno-Orovio, Elizabeth M Cherry, and Flavio H Fenton. “Minimal model for human ventricular action potentials in tissue”. In: *Journal of theoretical biology* 253.3 (2008), pp. 544–560.

[159] Raimond L Winslow, Jeremy Rice, Saleet Jafri, Eduardo Marban, and Brian O'Rourke. "Mechanisms of altered excitation-contraction coupling in canine tachycardia-induced heart failure, II: model studies". In: *Circulation research* 84.5 (1999), pp. 571–586.

[160] Vivek Iyer, Reza Mazhari, and Raimond L Winslow. "A computational model of the human left-ventricular epicardial myocyte". In: *Biophysical journal* 87.3 (2004), pp. 1507–1525.

[161] Dominique Chapelle, Annabelle Collin, and Jean-Frédéric Gerbeau. "A surface-based electrophysiology model relying on asymptotic analysis and motivated by cardiac atria modeling". In: *Mathematical Models and Methods in Applied Sciences* 23.14 (2013), pp. 2749–2776.

[162] Annabelle Collin, Jean-Frédéric Gerbeau, Méléze Hocini, Michel Haıssaguerre, and Dominique Chapelle. "Surface-based electrophysiology modeling and assessment of physiological simulations in atria". In: *International Conference on Functional Imaging and Modeling of the Heart*. Springer. 2013, pp. 352–359.

[163] Luca Pegolotti, Luca Dedè, and Alfio Quarteroni. "Isogeometric Analysis of the electrophysiology in the human heart: Numerical simulation of the bidomain equations on the atria". In: *Computer Methods in Applied Mechanics and Engineering* 343 (2019), pp. 52–73.

[164] Alessandro S Patelli, Luca Dedè, Toni Lassila, Andrea Bartezzaghi, and Alfio Quarteroni. "Isogeometric approximation of cardiac electrophysiology models on surfaces: An accuracy study with application to the human left atrium". In: *Computer Methods in Applied Mechanics and Engineering* 317 (2017), pp. 248–273.

[165] Maria Murillo and Xiao-Chuan Cai. "A fully implicit parallel algorithm for simulating the non-linear electrical activity of the heart". In: *Numerical linear algebra with applications* 11.2-3 (2004), pp. 261–277.

[166] Hüsnü Dal, serdar Goktepe, Michael kaliske, and Ellen Kuhl. "A fully implicit finite element method for bidomain models of cardiac electromechanics". In: *Computer Methods in Biomechanics and Biomedical Engineering* 253 (2013), pp. 323–336.

[167] John A Trangenstein and Chisup Kim. "Operator splitting and adaptive mesh refinement for the Luo–Rudy I model". In: *Journal of Computational Physics* 196.2 (2004), pp. 645–679.

[168] Jonathan P Whiteley. "An efficient numerical technique for the solution of the monodomain and bidomain equations". In: *IEEE Transactions on Biomedical Engineering* 53.11 (2006), pp. 2139–2147.

[169] Marc Ethier and Yves Bourgault. "Semi-implicit time-discretization schemes for the bidomain model". In: *SIAM Journal on Numerical Analysis* 46.5 (2008), pp. 2443–2468.

[170] Thomas Roy, Yves Bourgault, and Charles Pierre. "Analysis of time-stepping methods for the monodomain model". In: *arXiv preprint arXiv:1710.01106* (2017).

[171] Shankarjee Krishnamoorthi, Mainak Sarkar, and William S Klug. “Numerical quadrature and operator splitting in finite element methods for cardiac electrophysiology”. In: *International journal for numerical methods in biomedical engineering* 29.11 (2013), pp. 1243–1266.

[172] Yavuz Basar and Dieter Weichert. *Nonlinear continuum mechanics of solids: fundamental mathematical and physical concepts*. Springer Science & Business Media, 2013.

[173] Rodrigo Weber Dos Santos, G Plank, S Bauer, and EJ Vigmond. “Preconditioning techniques for the bidomain equations”. In: *Domain Decomposition Methods in Science and Engineering*. Springer, 2005, pp. 571–580.

[174] Pras Pathmanathan, Gary R Mirams, James Southern, and Jonathan P Whiteley. “The significant effect of the choice of ionic current integration method in cardiac electrophysiological simulations”. In: *International Journal for Numerical Methods in Biomedical Engineering* 27.11 (2011), pp. 1751–1770.

[175] Carla Manni, Francesca Pelosi, and M Lucia Sampoli. “Isogeometric analysis in advection–diffusion problems: Tension splines approximation”. In: *Journal of Computational and Applied Mathematics* 236.4 (2011), pp. 511–528.

[176] M Allessie, FI Bonke, and FJ Schopman. “Circus movement in rabbit atrial muscle as a mechanism of tachycardia. III. The” leading circle” concept: a new model of circus movement in cardiac tissue without the involvement of an anatomical obstacle.” In: *Circulation research* 41.1 (1977), pp. 9–18.

[177] Elizabeth M Cherry and Flavio H Fenton. “Visualization of spiral and scroll waves in simulated and experimental cardiac tissue”. In: *New Journal of Physics* 10.12 (2008), p. 125016.

[178] Niels F Otani. “A primary mechanism for spiral wave meandering”. In: *Chaos: An Interdisciplinary Journal of Nonlinear Science* 12.3 (2002), pp. 829–842.

[179] C Cherubini, S Filippi, P Nardinocchi, and L Teresi. “An electromechanical model of cardiac tissue: Constitutive issues and electrophysiological effects”. In: *Progress in biophysics and molecular biology* 97.2-3 (2008), pp. 562–573.

[180] Vlado A Lubarda. *Elastoplasticity theory*. CRC press, 2001.

[181] Carl Eckart. “The thermodynamics of irreversible processes. IV. The theory of elasticity and anelasticity”. In: *Physical Review* 73.4 (1948), p. 373.

[182] Erastus H Lee. “Elastic-plastic deformation at finite strains”. In: *ASME Journal of Applied Mechanics* 36 (1969), pp. 1–6.

[183] Edward K Rodriguez, Anne Hoger, and Andrew D McCulloch. “Stress-dependent finite growth in soft elastic tissues”. In: *Journal of biomechanics* 27.4 (1994), pp. 455–467.

[184] Alessio Gizzi, Christian Cherubini, Simonetta Filippi, and Anna Pandolfi. “Theoretical and numerical modeling of nonlinear electromechanics with applications to biological active media”. In: *Communications in Computational Physics* 17.1 (2015), pp. 93–126.

[185] Paola Nardinocchi and Luciano Teresi. “On the active response of soft living tissues”. In: *Journal of Elasticity* 88.1 (2007), pp. 27–39.

[186] Simone Rossi, Ricardo Ruiz-Baier, Luca F Pavarino, and Alfio Quarteroni. “Orthotropic active strain models for the numerical simulation of cardiac biomechanics”. In: *International journal for numerical methods in biomedical engineering* 28.6-7 (2012), pp. 761–788.

[187] Martyn P Nash and Alexander V Panfilov. “Electromechanical model of excitable tissue to study reentrant cardiac arrhythmias”. In: *Progress in biophysics and molecular biology* 85.2-3 (2004), pp. 501–522.

[188] Giulia Giantesio, Alessandro Musesti, and Davide Riccobelli. “A comparison between active strain and active stress in transversely isotropic hyperelastic materials”. In: *Journal of Elasticity* 137.1 (2019), pp. 63–82.

[189] D Ambrosi and S Pezzuto. “Active stress vs. active strain in mechanobiology: constitutive issues”. In: *Journal of Elasticity* 107.2 (2012), pp. 199–212.

[190] S GOVINDJEE. “The Symmetric Identity”. In: *Structural Engineering-Department of Civil Engineering Mechanics and Material–University of California Berkeley* (2010).

[191] Mikhail Itskov. “On the theory of fourth-order tensors and their applications in computational mechanics”. In: *Computer Methods in Applied Mechanics and Engineering* 189.2 (2000), pp. 419–438.

[192] Sven Klinkel and Sanjay Govindjee. “Using finite strain 3D-material models in beam and shell elements”. In: *Engineering Computations* (2002).

[193] Larry A Taber, Ming Yang, and W William Podszus. “Mechanics of ventricular torsion”. In: *Journal of biomechanics* 29.6 (1996), pp. 745–752.

[194] Alessandro Loppini, Alessio Gizzi, Ricardo Ruiz-Baier, Christian Cherubini, Flavio H Fenton, and Simonetta Filippi. “Competing mechanisms of stress-assisted diffusivity and stretch-activated currents in cardiac electromechanics”. In: *Frontiers in Physiology* 9 (2018), p. 1714.

[195] Viviane Timmermann, Lars A Dejgaard, Kristina H Haugaa, Andrew G Edwards, Joakim Sundnes, Andrew D McCulloch, and Samuel T Wall. “An integrative appraisal of mechano-electric feedback mechanisms in the heart”. In: *Progress in biophysics and molecular biology* 130 (2017), pp. 404–417.

[196] Serdar Göktepe and Ellen Kuhl. “Electromechanics of the heart: a unified approach to the strongly coupled excitation–contraction problem”. In: *Computational Mechanics* 45.2-3 (2010), pp. 227–243.

[197] S Pezzuto, D Ambrosi, and ALFIO Quarteroni. “An orthotropic active–strain model for the myocardium mechanics and its numerical approximation”. In: *European Journal of Mechanics-A/Solids* 48 (2014), pp. 83–96.

[198] Antonello Gerbi, Luca Dedè, and Alfio Quarteroni. “A monolithic algorithm for the simulation of cardiac electromechanics in the human left ventricle”. In: (2019).

[199] Francisco Sahli Costabal, Felipe A Concha, Daniel E Hurtado, and Ellen Kuhl. “The importance of mechano-electrical feedback and inertia in cardiac electromechanics”. In: *Computer methods in applied mechanics and engineering* 320 (2017), pp. 352–368.

[200] Ted Belytschko, Yun Yun Lu, and Lei Gu. “Element-free Galerkin methods”. In: *International journal for numerical methods in engineering* 37.2 (1994), pp. 229–256.

[201] Gui-Rong Liu and Yuan-Tong Gu. *An introduction to meshfree methods and their programming*. Springer Science & Business Media, 2005.

[202] A Gerhard Holzapfel. “Nonlinear solid mechanics II”. In: (2000).

[203] Lallit Anand and Sanjay Govindjee. *Continuum Mechanics of Solids*. Oxford University Press, USA, 2020.

List of related publications

- Nitti A., Kiendl J., Reali A., de Tullio, M. D. (2020). An immersed-boundary/isogeometric method for fluid–structure interaction involving thin shells. Computer Methods in Applied Mechanics and Engineering, 364, 112977.
- Nitti A., Kiendl J., Gizzi, A., Reali, A., de Tullio, M. D. A curvilinear Isogeometric framework for the electro-mechanical activation of thin cardiac tissues. *submitted to* Computer Methods in Applied Mechanics and Engineering (2021).

Declaration of Authorship

I, Alessandro Nitti, declare that this thesis titled, “Development of a multiphysics solver for complex coupled problems involving thin shells: fluid-structure-electrophysiology interaction” and the work presented in it are my own. I confirm that:

- This work was done mainly while in candidature for a PhD at the *Polytechnic University of Bari*. A portion of this work was done during a six-months visiting research period at the *Bundeswehr University Munich*.
- This work has benefited from the technical consultancy of Prof. J. Kiendl, Prof. A. Reali, and Prof. A. Gizzi, in addition to the supervision of Prof. M. D. de Tullio.
- Where I have consulted the published work of others, this is always clearly attributed.
- Any figure which is not referenced has been produced by myself.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Date: 31/10/2020