ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

MASTER THESIS

Isogeometric Analysis of Electrophysiological Models on Surfaces

Advisor:
Prof. Alfio QUARTERONI
Supervisors:
Dr. Luca DEDE'
Dr. Toni LASSILA

Author:
Alessandro PATELLI

June 20, 2014

A vendi mia una vaca
per fa studia un asan.
—My Dad

Abstract

In this project we numerically simulate electrophysiological models for cardiac applications by means of Isogeometric Analysis. Specifically, we aim at understanding the advantages of using high order continuous NURBS (Non-Uniform Rational B-Splines) basis functions in the approximation of the traveling waves of the action potential. As application, we consider the numerical simulations on the human left atrium modelled as a surface. Firstly in our analysis, we consider a benchmark time-dependent diffusion-reaction problem describing a traveling front in a two dimensional domain, for which we aim at understanding the role of NURBS basis functions in the approximation of the conduction velocity. Then, we extend the analysis to more complex electrophysiological models, in particular to the numerical approximation of the monodomain equation. The latter is a Partial Differential Equation and a system of Ordinary Differential Equations. We consider the Aliev-Panfilov model and we analyze the different aspects related to its numerical approximation, including the role of high order continuous NURBS basis functions in the simulation of cardiac excitation models. Then, we consider realistic simulations of the Mitchell-Schaeffer model on the human left atrium represented as a surface for which the strong anisotropic behavior of the action potential, due to the fiber orientation of the cardiac tissue, is taken into account.

Contents

Introduction

The study of the electrical properties of biological cells and tissues is one of the main purposes of electrophysiology in cardiac modeling [1, 2]. The physical quantities involved are the action potential and the electric current, which span on a wide variety of scales, namely from single cells to the whole heart. Many cardiac diseases are directly related to the electrical activity of the heart. In particular, cardiac dysrhytmias such as bradycardias or Tachycardias, may be caused by a malfunction of the electrical part of the heart. Therefore, many researches have been made in order to mathematically model the propagation of the action potential in the cardiac tissue [3, 4] and to improve the understanding of the complex dynamics related to the electrical activity of the heart. In this context, the numerical simulations are a useful tool to study the electrophysiological models and describe the electrical activity of the heart from the action potential propagation to more complexes dynamics, such as spiral waves.

The classical Finite Element Method (FEM) [5, 6] with Lagrangian basis functions has been widely used for the simulation of cardiac excitation models [7]. The monodomain equations is a simple model used for the simulation of the propagation of the action potential, which represent a coupling between a time-dependent nonlinear reaction-diffusion Partial Differential Equations (PDE) and a system of Ordinary Differential Equations (ODEs). The numerical approximation of the monodomain model leads to the approximation of a fast traveling pulse. The principal issue related to the simulation of such excitable wave is to approximate the proper conduction velocity. It has been shown that the velocity of propagation spatially approximated by the Galerkin-FEM using linear polynomial basis functions, strongly depends on the mesh size used of the computation [8]. This finding is important because shows a limitation of the classical FEM in solving electrophysiological models for large scale computations.

In this work, the PDEs are spatially approximated by means of NURBS-based Isogeometric Analysis (IGA) in the framework of the Galerkin method, where NURBS (Non-Uniform Rational B-Splines) are one of the possible basis functions. IGA is a recent computational methodology, which integrates the FEM with the Computed Aided Design (CAD) [9, 10]. This methodology is based on the isogeometric paradigm, for which the same basis functions used to represent the known geometry are then used for the approximation of the PDEs. This aspect of representing exactly the computational domain is only one of the peculiar aspects of IGA. In fact, another property of NURBS basis functions is that their global continuity can be increased from C^0 to at most C^{p-1}, where $p > 0$ is the polynomial order of the basis functions. The latter property plays a central role for this work. Since we are interested in understanding the advantages of NURBS basis functions in the numerical approximation of electrophysiological models on surfaces.

Finally, we numerically approximate and simulate the propagation of the action potential on a realistic geometry of the human left atrium which we represent by means as a surface. We represent the atrium by using NURBS basis functions which are namely C^0- and C^1-continuous, simulate the Mitchell-Schaeffer model [2] by using IGA, and we critically discuss the numerical results.

This work is organized as follows. Chapter 1 introduces to the mathematical formulation of the cardiac electrophysiology while, Chapter 2 to the formulation of Isogeometrical analysis. In Chapter 3 we study a benchmark time-dependent reaction-diffusion problem describing a traveling front; this test problem represents a simplification of the monodomain model. The aim of this chapter is to

study the relation between the conduction velocity of the traveling front and the global continuity of the NURBS basis functions considered for the numerical approximation. In Chapter 4 we simulate the Aliev-Panfilov model [1] in a two dimensional domain with the goal to understand the advantages of high order continuous NURBS basis functions in the approximation of different electrophysiological models. In Chapter 5 we propose a realistic simulation of the Mitchell-Schaeffer model [2] on the human left atrium represented by means of a NURBS surface. A qualitative analysis between simulations obtained with a different total number of degrees of freedom is made. Conclusions follow.

Chapter 1

Mathematical Modeling of the Cardiac Electrophysiology

1.1 Introduction to Electrophysiology

1.1.1 The heart anatomy

The heart is the main component of the circulatory system and has the function of pumping blood throughout the body. It is composed by two atria (upper chambers) and two ventricles (lower chambers) and it is divided in two sides: right and left; see Figure (1.1), left. Each side consists of an atrium and a ventricle. The right atrium receives deoxygenated blood and transfers this blood to the right ventricle which pumps it, via the pulmonary artery, into the lungs. The oxygenated blood returns in the left atrium through the pulmonary veins and passes to the left ventricle. Finally the oxygenated blood is pumped throughout the body via the aorta. As shown in Figure (1.1), right, the heart tissue layer is divided into three parts, the endocardium (most internal part), the myocardium, and the epicardium (most external part). The sac containing the heart is called the pericardium. The contraction of the two atria and the two ventricles is due to an electrical stimulation started at the Sinoatrial node (**SA**), the pacemaker of the heart. From the SA node the electrical conduction passes from the right atrium to the left atrium (interatrial conduction).

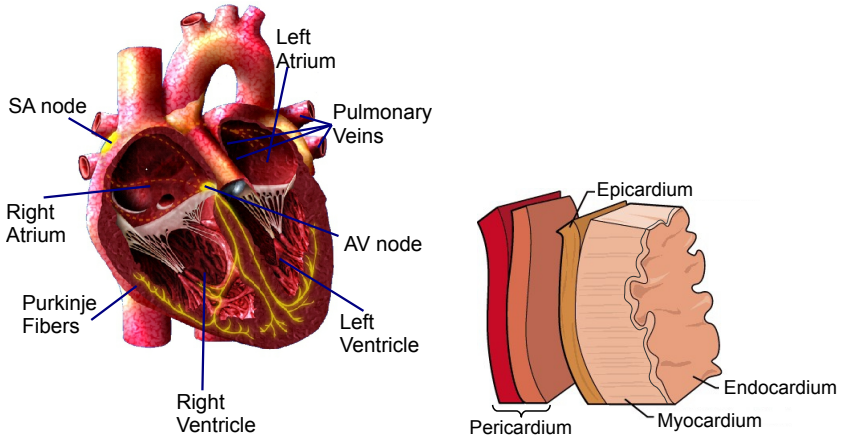


Figure 1.1: Left: The intern of the heart is shown with the mains parts names'. Right: The cardiac tissue wall layers.

As the atria is electrically isolated from the ventricles (in a healthy heart), the electrical conduction is filtered through the Atrioventricular node (**AV**). The function of this structure is to provide a delay in the conduction of the electrical impulse between the atria and the ventricles. This delay permits the atria to contract and allows the blood to fill completely the ventricles. Then, after the electrical

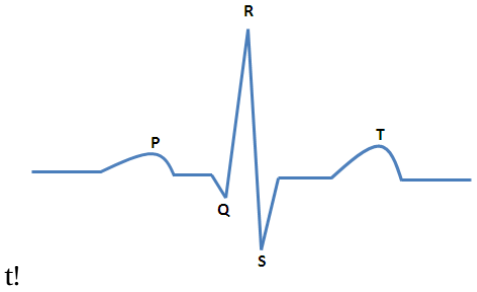


Figure 1.2: A ECG of a normal sinus cardiac rhythm

conduction passes the AV node, the Purkinje Fiber network delivery the electrical current to the ventricles. Finally, the contraction of the ventricles pumps the blood throughout the body via the aorta (left ventricle) and in the lungs via the pulmonary artery (right ventricle).

The normal beat rhythm of the heart is around 60-100 beats/minute (bpm) and is called sinus rhythm; in Figure (1.2) the ECG of the normal cardiac cycle is shown. The impulse generated by the SA generates the P waves. This phase reflects the depolarization of the atria as the wave goes toward the AV node. The phase called QRS complex represents the rapid depolarization of the ventricles. Finally, the repolarization of the ventricles is reflected by the T waves. In order to optimize the heart performance all the components must work effectively, both mechanically and electrically. The malfunction of the electrical part may lead to the malfunction of the heart and thus to an irregular heart beat. Such problems are called cardiac dysrhythmia or simply arrhythmia.

1.1.2 Cardiac cells and ionic channels

The primary cell types of cardiac cells are the SA cells, the AV cells, the Purkinje fiber cells, and the myocardial cells in both atria and ventricles. Each of the previous cited cells presents different characteristics and functions. In this chapter we focus our attention on the myocardiac or cardiomyocytes, that are muscle cells.

Cardiomyocytes are responsible for the primary function of the heart and they have a cylindrical shape with at least a factor of 10 of difference between their length and diameter. In particular the length of a cardiomyocytes is of approximately 100 μm and the diameter between 10 and 25 μm. Each cell is connected to its neighbors in an end-to-end fashion which implies that the global structure of the cardiac tissue is arranged in fiber-like structure. For example in the ventricle the fibers are organized in toroidal layers nested within the cardiac wall and from the epicardium to the endocardium they rotating counterclockwise. Moreover in [11, 12] evidence has been presented that the fiber structure of the ventricles have an additional laminar organization. As consequence of the fiber structure, the cardiac tissue is strongly anisotropic with wave speeds that have a strong dependence on the fiber direction. In fact in the myocardium the propagation of potential (see section 1.2) is about 0.5 ms^{-1} along the fibers and about 0.17 ms^{-1} transversely to the fibers.

Each cardiomyocyte is embedded in a membrane called sarcolemma which protects the cell. The membrane is made of phospholipid and is thickness varies between 4 and 5 μm. The semipermeability of the membrane limits the inflow of hydrosoluble molecules but allows the flow of ions through the ionic channels.

There are at least one hundred different types of ionic channels each of which carry on the flux of a single type of ion, or multiple types of ions, or other molecules. Moreover some channels are called **gates** due to the fact that they can be opened or closed. For the sake of simplicity, in this section we introduce only the channels related to three ions which play a central role in the bioelectrical activity of the heart: the sodium (Na^+), the potassium (K^+) , and the calcium (Ca^{2+}). The three channels related to the previous ions are gates and their opened-closed status is voltage-dependent.

1.2 Mathematical models of the bioelectric activity of the heart

1.2.1 Nonexcitable and excitable cells

In order to understand electrical signaling in cells it is helpful to characterize the cells into two groups: **excitable cells** and **nonexcitable cells**. The first group of cells has the characteristic that under the application of a strong enough current the membrane potential goes through a large excitation, called **action potential**, before returning to the rest. On the contrary, when a current is applied to a nonexcitable cells for a certain period of time, the potential returns immediately to the equilibrium status after the current stops. Cardiac cells are excitable cells. The major advantage for excitable cells is that noises are filtered because excitable cells could respond to a stimulus fully or not at all. In this manner stimulus not enough strong may be related to background noise.

The most important milestone in the study of generation and propagation of signals in cells is the work of Hodgkin and Huxley in 1952, who derived the first quantitative model of wave propagation of an electrical signal along a giant squid axon [13]. This model has an important impact on the modeling of the bioelectric activity of cardiac cellular membrane as it influences the work of FitzHugh who derived a simpler model which is very extremely suitable for theoretical and numerical study [14].

1.2.2 Action potential

In this section we briefly introduce the different phases describing the action potential dynamic of cardiac cells, shown in Figure (1.3).

Let us assume to stimulate a cardiac cell by applying a sufficiently high current. In general, the action potential has 5 characteristic phases. The first phase (called phase 0) of the action potential dynamics is a fast depolarization due to the influx of sodium ions (Na^+). Phase 1 is characterized by a small decay due to the outflow of the potassium. Phase 2; is called the *plateau* phase and results from the opening of voltage-sensitive calcium (Ca^{2+}) channels. Finally a slow phase of depolarization due to the ongoing outflow of potassium ions (K^+) characterizes phase 3. The final phase, phase 4, is the resting phase when the ions concentrations values are reset.

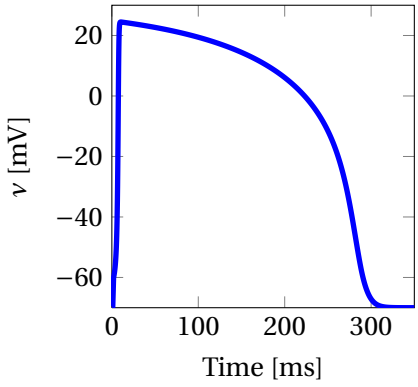


Figure 1.3: Action potential shape

As said in the section 1.2.1 excitable cells only respond to a sufficiently strong stimulus. A property of the action potential is that its shape does not depend on the intensity of the applied stimulus. Moreover the cardiac cells can not be continuously excited, they have a period, called **refractory period**, during which they do not respond to any stimulus applied to them.

1.2.3 Traveling front wave in excitable systems

In section 1.2.1 we introduced the concept of excitable and nonexcitable cells. By extending this phenomena from a cellular level to an entire tissue made of excitable cells, we come up with an

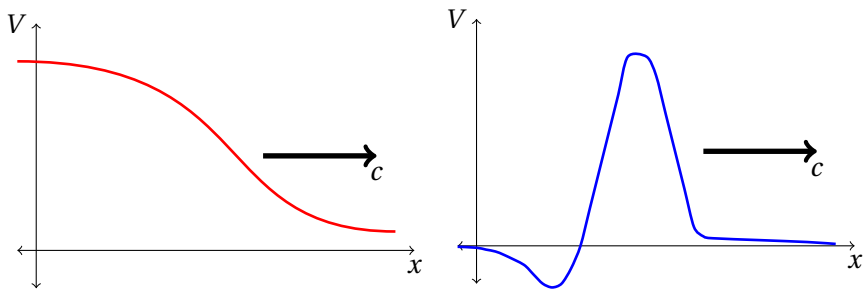


Figure 1.4: Left: A traveling front; Right: A traveling pulse.

excitable media, or **excitable system**. Thus, in this Section, we will refer to excitable media or system, as a group of individual cardiac cells placed one near each other forming part of cardiac tissue, with the property that each cell can communicate with its neighbors, i.e. each cell can pass information to its adjacent cells. In [15] the authors highlight three dynamical characteristics that are typical for excitable media; the medium (i) has a globally stable equilibrium, (ii) has an excited state which is attained temporarily in response to a stimulus which exceeds some threshold level, and (iii) has a refractory period, during which the medium does not respond to further stimulation and gradually returns to the resting state. The latter is the capacity of an excitable medium to propagate (excitable) waves.

From now on we will refer to excitable waves as traveling waves and we will distinguish between two different types of traveling waves: we call **traveling waves** the waves that look like a moving plateau, or transition between different level; we will instead call traveling pulse the waves that begin and terminate at the same value. In Figure (1.4) the two different types of waves are shown.

The one dimensional mathematical formulation of a traveling wave is possible via the **bistable equation** which reads:

$$\frac{\partial V}{\partial t} = \frac{\partial^2 V}{\partial x^2} + f(V), \quad \text{in } \Omega \times (0,T), \tag{1.1}$$

together with suitable initial and boundary conditions. In Eq. (1.1), $\Omega \subseteq \mathbb{R}$, $T > 0$, $f(V)$ is such that it possesses has three zeros at the values 0, α, and 1, where $0 < \alpha < 1$. An example of a possible function for f is the following cubic polynomial,

$$f(V) = aV(V-1)(V-\alpha),\ 0 < \alpha < 1.$$

In this case Eq. (1.1) is a nonlinear diffusion-reaction parabolic problem. Eq. (1.1) can be generalized to higher dimensional problems; in those cases the one-dimensional second derivative in space would be replaced by the Laplace operator.

1.2.4 Ionic current membrane model

The transmembrane potential, or **action potential**, is the jump of the potential V across the cellular membrane surface and it is the physical quantity of interest in the study of the bioelectric activity of a myocyte. We will refer to the internal region of the cell as intracellular (i) and to the external region as extracellular (e).

Citing [7] chapter 5: "*the electrical behavior of the membrane is represented by a circuit consisting of a capacitor connected in parallel with a resistor, modeling the various ionic channels regulating the selective and independent ionic fluxes through the membrane*". The transmembrane current I_m reads

$$I_m = C_m \frac{dV}{dt} + I_{ion} + I_{app}, \tag{1.2}$$

where I_{ion} denote the ionic membrane current, C_m the membrane capacity and I_{app} the applied

current per unit of the membrane surface. The general form of the ionic current is given by

$$I_{ion}(V,\mathbf{w},\mathbf{c}) = \sum_{k=1}^{P} g_k(\mathbf{c}) \prod_{j=1}^{M} w_j^{p_{j_k}} (V - V_k(c)) + I_0(V,\mathbf{c}), \tag{1.3}$$

where $g_k(\mathbf{c})$, $V_k(\mathbf{c}) = (RT/zF)\log\dfrac{c_k^e}{c_k^i}$, and $c_k^{i,e}$ are the maximal conductance of the ion channel, the Nersnt potential, and the intra- extra cellular ion concentrations for the $k-$th ionic species, respectively, with p_{k_j} integers. In Eq. (1.3) the current is split into the sum of quantities depending on ionic fluxes modulated by the gating dynamics and a time-independent term $I_0(V,\mathbf{c})$. The variables $\mathbf{w} := (w_1, w_2, \ldots, w_M)$ are the gating variables and regulate the conductance of the various ionic fluxes and $\mathbf{c} := (c_1, c_2, \ldots, c_Q)$ are variables regulating the intracellular concentration of the various ions. More details of gating variables and ionic channels are given in [3].

1.2.5 The Monodomain equation

By taking into account the intra- and the extracellular media, the cardiac tissue is represented as superposition of two continuous media coexisting at every point of the tissue and separated by a membrane. Thus, the cardiac tissue is considered as a *bidomain*. The so called bidomain model is a coupling between a system of reaction-diffusion parabolic equations and a system of ODEs. We recall that the cardiac tissue is strongly anisotropic due to the fiber structure (see section 1.1.2), which implies that the propagation of the potential is strongly influenced by the fiber direction and orientation. The first system of partial differential equations describes the two different cellular potentials: u_i being the intracellular and u_e the extracellular potential. The second system of ordinary differential equations are related to the ionic gating variables $\mathbf{w}$ and the ionic concentrations $\mathbf{c}$; see Eq. (1.3). Thus the bidomain system of equations reads:

$$\begin{cases} c_m \dfrac{\partial V}{\partial t} - \nabla\cdot(\mathbf{D}_i \nabla u_i) + I_{ion}(V,\mathbf{w},\mathbf{c}) & = I_{app}^i, \quad \text{in } \Omega\times(0,T) \\ -c_m \dfrac{\partial V}{\partial t} - \nabla\cdot(\mathbf{D}_e \nabla u_e) - I_{ion}(V,\mathbf{w},\mathbf{c}) & = I_{app}^e, \quad \text{in } \Omega\times(0,T) \\ \dfrac{\partial \mathbf{w}}{\partial t} - R(V,\mathbf{w}) = 0, \ \dfrac{\partial \mathbf{c}}{\partial t} - S(V,\mathbf{w},\mathbf{c}) & = 0, \quad \text{in } \Omega\times(0,T), \end{cases} \tag{1.4}$$

complemented with suitable initial and boundary conditions. Respectively, $\mathbf{D}_i$ and $\mathbf{D}_e$ are the intra- and extracellular anisotropic conductivity tensors, $c_m = \chi C_m$, C_m is the membrane capacity, and χ is the ratio of membrane area per tissue volume. We recall that $V = u_i - u_e$ is the transmembrane. A complete presentation of the bidomain equations can be found in [4, 7].

Solving the system of PDEs coupled with the system of ODEs (1.4) requires a high computational cost. For this reason a relaxed model called **monodomain** model is widely used in large scale simulations. The monodomain model is described by a nonlinear time-dependent reaction-diffusion equation and a system of Ordinary Differential Equations. Thus the transmembrane potential $V(\mathbf{x},t)$ is governed by the following system of equations:

$$\begin{cases} \chi\left(C_m \dfrac{\partial V}{\partial t} + I_{ion}(\mathbf{u},V)\right) - \nabla\cdot(\mathbf{D}\nabla V) = I_{app}, \ \text{in } \Omega\times(0,T) \\ \dfrac{\partial \mathbf{u}}{\partial t} = f(\mathbf{u},V), \ \text{in } \Omega\times(0,T), \end{cases} \tag{1.5}$$

with suitable initial and boundary conditions, where $\mathbf{D}$ is the anisotropic conductivity tensor, C_m, χ, and I_{app} are the same quantities introduced in Eq. (1.2).

1.2.6 Two variable model of cardiac excitation

Ionic models describe most of the basic properties of the cardiac tissue, in particular the depolarization and repolarization phase of the action potential and restitution properties. In the past decades several ionic models have been derived all with a different numbers of equations. For example, the Bueno-Orovio et al. ionic model counts 4 variables [16], while ten Tusscher et al. developed two different ionic models respectively with 17 and 19 equations [17, 18]. Clearly there is a relation between computational complexity and number of variables needed by the ionic models. Thus, in order to avoid numerical difficulties, researchers developed other models for cardiac excitation such as the two-variable FitzHugh-Nagumo model [14, 19]. These different types of models are more suitable for intensive two- or three-dimensional computations [20], and permit analytical analysis [3]. In this section we present two minimal models for the cardiac excitation dynamics: the Aliev-Panfilov model [1] and the Mitchell-Schaeffer model [2]. Both are a modification of the FitzHugh-Nagumo model [14, 19].

1.2.6.1 *Aliev-Panfilov model*

R.R. Aliev and A.V. Panfilov proposed a modification of the FitzHugh-Nagumo model [14, 19] which simulates the restitution property of cardiac tissue, adequately represents the shape of the action potential, can be efficiently implemented, and can be used in the simulation of the action potential the whole heart. The model does not try to simulate the internal dynamics of the cell, but to describe properly the characteristics of the propagation of the excitable wave in canine myocardium. The model consists of two equations: one nonlinear diffusion-reaction parabolic equation and an Ordinary Differential Equation. Let $\Omega \subseteq \mathbb{R}^d$, $d = 1,2,3$ be a open domain and $I \subset \mathbb{R}^+$ be the time interval, the Aliev-Panfilov model reads:

$$\begin{cases} \dfrac{\partial \overline{V}}{\partial \overline{t}} = \nabla \cdot (\mathbf{D} \nabla \overline{V}) - k\overline{V}(\overline{V}-a)(\overline{V}-1) - \overline{V} r + I_{app}, \text{ in } \Omega \times I, \\ \dfrac{dr}{d\overline{t}} = \left(\epsilon_0 + \dfrac{\mu_1 r}{\overline{V}+\mu_2}\right)(-r - k\overline{V}(\overline{V}-a-1)), \text{ in } \Omega \times I, \end{cases} \tag{1.6}$$

together with suitable initial and boundary conditions. In Eq. (1.6), $\mathbf{D}$ is the anisotropic conductivity tensor and the values of the parameters are given in Table (1.1). The variables $\overline{V}$, r, and $\overline{t}$ are dimensionless quantities. The actual potential V $[mV]$ and time t $[ms]$ are as:

$$V = \quad 100\overline{V} - 80, \tag{1.7}$$

$$t = \quad 12.9\overline{t}. \tag{1.8}$$

Parameter	Value
ϵ_0	0.002
k	8
a	0.15
μ_1	0.2
μ_2	0.3

Table 1.1: Aliev-Panfilov parameters model values

The variable $\overline{V}$ varies from 0 to 1, thus the real potential, V varies form -80 mV to the maximal value of 20 mV. The time has been scaled in order to have the action potential duration (APD) measured at the level of 90% of repolarization, equal to $APD_0 = 330$ ms.

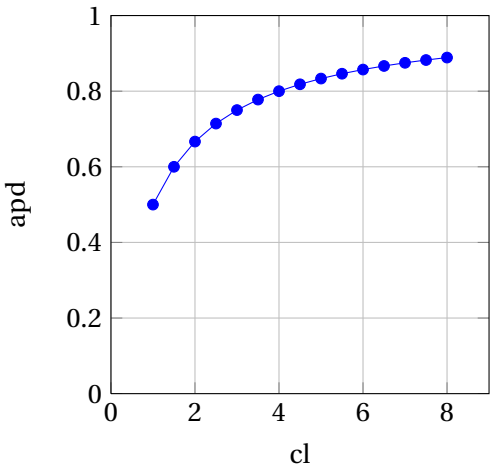


Figure 1.5: Restitution property of the medium represented by the Eq. (1.9).

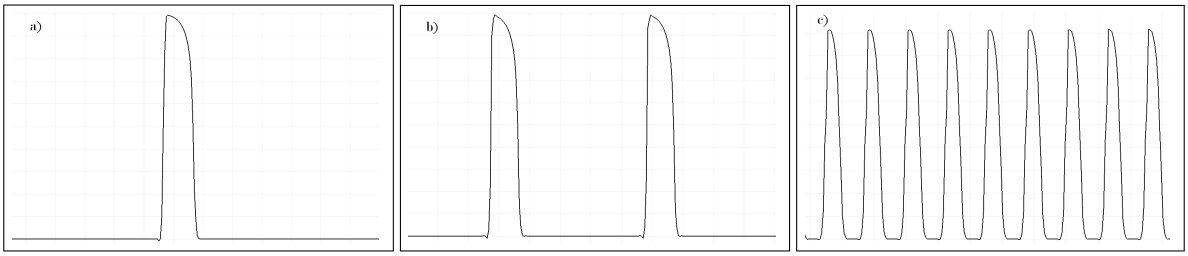


Figure 1.6: Action potential shape and duration depending on the cycle length, T: (a) $cl = 8$; (b) $cl = 4$; (c) $cl = 1$.

Experiments on canine myocardium [21] show that there is a relation between the APD and the cycle length (CL) of stimulation, i.e., the time interval between two stimuli. In [1] a relation between the APD and CL is given by the following relation:

$$\frac{1}{apd} = 1 + \frac{b}{cl}, \tag{1.9}$$

where $apd = APD/APD_0$ and $cl = CL/APD_0$ and b is a suitable constant. In Figure (1.5) we show the behavior of the apd for different values of cl using Eq. (1.9). The behavior of the action potential duration for three different periods of stimulation is shown in Figure (1.6). This behavior of the potential means that the closer are the stimuli, the shorter is the action potential duration; this property of the cardiac tissue allows the heart to accelerate the beats when the body is under a physical effort. In [1] a comparison between the experimental data presented in [21] and the result of the simulation obtained with the model (1.6) is shown and the major remark is that the simulated apd, computed for different cl, lies on averaged lines obtained from the experimental data.

Finally, the Aliev-Panfilov model permits also to simulate vortex dynamics just by applying an extra stimulus in the wake of the passing wave; for more details see section 4.3.

1.2.6.2 *Mitchell-Schaeffer model*

The Mitchell-Schaeffer model is another example of minimal model for describing the dynamics of the cardiac tissue. In [2] the model is presented as a system of two ODEs, but we introduce the model as a coupled system of a PDE and an ODE as done for the Aliev-Panfilov model. The variables of the model are the transmembrane potential $\overline{V}$ and a gating variable h. Let us assume that $\Omega \subseteq \mathbb{R}^d$, $d = 1,2,3$, is an open domain and $I \subseteq \mathbb{R}^+$ is the time interval. The Mitchell-Schaeffer model coupled

Parameter	Value
τ_{in}	0.3 ms
τ_{out}	6 ms
τ_{open}	120 ms
τ_{close}	150 ms
V_{gate}	0.13

Table 1.2: Mitchell-Schaeffer parameters model values

with the monodomain equation reads:

$$\begin{cases} \dfrac{\partial \overline{V}}{\partial t} = \nabla \cdot (\mathbf{D} \cdot \nabla \overline{V}) + \dfrac{h}{\tau_{in}} \overline{V}^2 (1 - \overline{V}) - \dfrac{\overline{V}}{\tau_{out}} + I_{app}, \text{ in } \Omega \times I, \\ \dfrac{dh}{dt} = \begin{cases} \dfrac{1-h}{\tau_{open}}, \text{ if } \overline{V} < V_{gate} \\ \dfrac{-h}{\tau_{close}}, \text{ if } \overline{V} > V_{gate} \end{cases}, \text{ in } \Omega \times I, \end{cases} \tag{1.10}$$

together with suitable initial and boundary conditions. In Eq. (1.10), $\mathbf{D}$ is the anisotropic conductivity tensor and I_{app} is the applied current. The other parameters value are reported in Table (1.2). The variables $\overline{V}$ and h are dimensionless quantities varying between 0 and 1. The relation to scaled back the voltage to the real physiological values is the following:

$$V = V_{min} + \overline{V}(V_{max} - V_{min}),$$

where we set $V_{min} = -70\ mV$ and $V_{max} = 30\ mV$.

Finally, the simplicity of Eq. (1.10) makes easy the derivation of an explicit formula for the restitution curve, thus the Mitchell-Schaeffer model can describe qualitatively the same behavior previously shown in Figure (1.5).

Chapter 2

Isogeometric Analysis

The idea behind Isogeometric Analysis (IGA) is the integration of the Computer Aided Design (CAD) technology and the classical Finite Element Analysis (FEA) [9]. For the numerical solution of Partial Differential Equations (PDEs) using FEA, designers generate CAD models and translate them into analysis-suitable geometries. The geometries are then meshed and ready to be imported in FEA codes. This approach has many consequences. Firstly, the introduction of a numerical error between the computational geometry and the Finite Element mesh which could lead to accuracy issues. Secondly, the computational cost required to construct the Finite Element mesh is quite high, and represents one of the more time consuming steps in the FEA, [9]. The last consequence we want to focus on is related to the communications issues between CAD system and Finite Element solvers. In general, when performing mesh refinement, the geometrical approximation of the mesh does not improve if automatic communicators between CAD and the mesh generator are not established. IGA, as FEA, is based on a *isoparametric paradigm*: the same basis functions used for the representation of the known geometry are also used to represent the unknown solution field. However, the paradigm presents a significant difference between the two methodologies. In fact, in FEA, firstly the basis representing the PDEs unknown solution field is chosen and then the same basis is used for the representation and approximation of the known geometry. IGA, on the contrary, firstly chooses the basis to represent the known geometry which is later used to approximate the unknown solution field of the PDEs. Many different computational geometry technologies may be adopted to represent geometries. In this work, we choose to present and use the Non Uniform Rational B-Splines (NURBS) due to their large use and development in CAD. In particular, NURBS can **exactly** represent conic section, e.g. circles, spheres, etc. They also possess remarkably useful mathematical properties, such as C^{p-1} continuity for p-th degree NURBS. In this chapter the main properties and features of Isogeometric Analysis are presented. The first part will focus on the geometric representation of shapes using NURBS, while the second part is dedicated to the use of NURBS as basis for the analysis. We especially present IGA in the framework of the Galerkin method and solve different equations: the heat equation and a non-linear time-dependent diffusion-reaction problem.

2.1 B-Splines and NURBS

2.1.1 Geometrical representation with B-splines and NURBS

The first difference between classical FEA and IGA based on NURBS is that the **parameter space** is local to elements for the former while local to **patches** for the latter. More precisely, in FEA each element has its own mapping into the physical space, while the B-Splines, or NURBS, maps a patch of multiple elements from the parameter space into the physical space. Thus, the mapping is global to the whole paths rather than to the elements. Patches are seen as **subdomains** within which element types and material models are assumed to be uniform. In this work we consider domains represented only with a single patch. In the next section we introduce the B-Splines.

2.1.2 Knot vectors

In one dimension a **knot vector** is non-decreasing set of coordinates in the parameter space, written $\Xi = \{\xi_1, \xi_2, \ldots, \xi_{n+p+1}\}$, where $\xi_i \in \mathbb{R}$ is the i^{th} **knot** and the index i, $i = 1, 2, \ldots, n+p+1$, where p is the polynomial degree and n is the number of basis functions used to represent the B-Splines basis. The partition of the parameter space into elements is done by the knots. Knots vectors may be **uniform** or **non-uniform** if the knots are equally or unequally spaced in the parameter space. Moreover knot values may be repeated more then once. The latter property is called **multiplicities of knots** and has important implications for the properties of the basis functions. A knot is said to be **open** if its first and last knots values are repeated $p+1$ times. The peculiarity of basis functions formed from open knot vectors is that the basis functions are interpolators at the ends of the parameter interval, $[\xi_1, \xi_{n+p+1}]$, which in not in general true at the interior knots. This is one of the major differences between knots and nodes in the FEA. The same observation still holds when considering patches in multiple dimensions, where the basis functions are interpolators at the corners of the patches. Another important consideration in the multiple dimensional case is about the boundary of the domain. In fact the boundary of B-spline object with d parametric dimension is itself a B-spline object of dimension $d-1$. Let define $\hat{\Omega} = [0,1] \subseteq \mathbb{R}$ the one dimensional parameter domain. Given a knot vector $\Xi = \{\xi_1, \xi_2, \ldots, \xi_{n+p+1}\}$ with $\xi_1 = 0$ and $\xi_{n+p+1} = 1$, the mesh Ω_h over the domain $\hat{\Omega} = [0,1]$ is the partition into elements induced by the knots on the parameter domain without taking into account their multiplicity. We will denote $\hat{\Xi} = \{\hat{\xi_1}, \hat{\xi_2}, \ldots, \hat{\xi_q}\}$ the vector containing all distinct knots of Ξ. The mesh over $\hat{\Omega}$ is defined by the subdomain $\hat{\Omega_j} := (\hat{\xi}_j, \hat{\xi}_{j+1})$, for $j = 1, 2, \ldots, q$. The **knot span** is instead defined by two consecutive knots: (ξ_i, ξ_{i+1}), for $i = 1, 2, \ldots, n+p$. The knot spans has null size when $\xi_i = \xi_{i+1}$.

2.1.3 Basis functions

Let us consider the knot vector $\Xi = \{\xi_1, \xi_2, \ldots, \xi_{n+p+1}\}$ and the corresponding parameter domain $\hat{\Omega}$. The one dimensional B-spline basis functions are defined recursively by the **Cox-de Boor recursion formula** starting from $p = 0$:

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

For $p > 0$ they reads:

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

Thus, $N_{i,p} : \hat{\Omega} \to \mathbb{R}$, for $i = 1, \ldots, n$ are piecewise polynomials of degree p. By convention we assume that $\frac{0}{0} = 0$.

The number n of B-Splines basis functions $\{N_{i,p}\}_{i=1}^{n}$ and the polynomial degree p could be easily deduced from the knot vector Ξ. Moreover, the regularity of the basis functions across the knots, ξ_i, is such that they are $\mathscr{C}^{p-m_i}-$continuous, where m_i is the multiplicity of the i^{th} knot. In particular, if the multiplicity m_i is equal to the polynomial degree p, the basis function is interpolatory in the i^{th} knot. There are several more important features of the basis functions. The basis constitutes a partition of unity, which is, $\forall \xi \in \hat{\Omega}$,

$$\sum_{i=1}^{n} N_{i,p}(\xi) = 1.$$

Another important observation is that each basis function is pointwise non-negative in the domain, which implies that all the entries of the mass matrix are positive. One of the major properties is that each function of order p possess $p-1$ continuous derivatives across the boundaries elements and

is the one of the most distinctive features of IGA. The last observation is about the support of the B-spline functions. Assuming that B-Splines basis functions of degree p are given, their knot spans are always $p+1$, thus high-order functions have support over much larger portion of the domain than classical FEA functions do. Finally, we introduce the notion of smoothness integer indexes which is basically a collection of integers $\mathbf{K} = \{k_1, \dots, k_q\}$ where $k_j = p + 1 - m_j$ and for which we define $k_{min} := \min_{j=2,\dots,q-1}\{k_j\}$ and $k_{max} := \max_{j=2,\dots,q-1}\{k_j\}$. Thus the continuity of the basis functions at the knot $\hat{\xi}_j$ are $\mathscr{C}^{k_j-1}-$continuity, for $j = 1, \dots, n$.

The extension of the definition of univariate B-Splines to $\kappa-$dimensional B-Splines can be easily done by using the tensor product structure, where $\kappa \geq 1$. Let us consider κ knot vectors $\Xi_\alpha = \{\xi_1^\alpha, \xi_2^\alpha, \dots, \xi_{n_\alpha+p\alpha+1}^\alpha\}$, where $\alpha = 1, \dots, \kappa$. A **mesh** $\mathscr{T}_h$ over the parametric domain $\hat{\Omega} = [0,1]^\kappa$ is defined by its partition into $\kappa-$dimensional elements:

$$\mathscr{T}_h := \{\Omega_{\mathbf{j}} := \otimes_{\alpha=1}^{\kappa}(\hat{\xi}_{j_\alpha}^\alpha, \hat{\xi}_{j_\alpha+1}^\alpha),\ 1 \leq j_\alpha \leq q_\alpha - 1\}, \tag{2.3}$$

where $\mathbf{j} = (j_1, \dots, j_\kappa) \in \{\mathbf{i} = (i_1, \dots, i_\kappa) : 0 \leq i_\alpha \leq n_\alpha, 1 \leq \alpha \leq \kappa\}$ is a multi-index. For all the elements $\Omega_{\mathbf{j}}$ of the mesh $\mathscr{T}_h$ the element size is computed as $h_{\Omega_{\mathbf{j}}} := \text{diam}(\Omega_{\mathbf{j}})$ for all multi-index $\mathbf{j}$ and the **global mesh size** will be set as $h = \max_{\mathbf{j}}\{h_{\Omega_{\mathbf{j}}}\}$. Finally, we can define the multivariate B-Splines basis function for $j_\alpha = 1, \dots, n_\alpha$ with $\alpha = 1, \dots, \kappa$ as:

$$N_{\mathbf{j}} : \hat{\Omega} \to \mathbb{R}, \quad N_{\mathbf{j}}(\xi) := \prod_{\alpha=1}^{\kappa} N_{j_\alpha}^\alpha(\xi_\alpha). \tag{2.4}$$

Thanks to the tensor product structure, the multivariate B-Splines are piecewise polynomials, pointwise non-negative, and constitute a partition of the unity, all this being properties that are inherited from univariate B-Splines basis functions. The tensor product structure helps us also to deduce some other properties of the multivariate B-Splines. In fact, their support is still compact and their regularity could be seen as directional regularity and is deduced by the one of the univariate functions used in the tensor product. As in the univariate case the basis functions are $\mathscr{C}^\infty-$continuous in all elements. We can extend the definition of the smoothness indexes for each parametric direction: for all $\alpha = 1, 2, \dots, \kappa$ define $\mathbf{K}_\alpha = \{k_1^\alpha, \dots, k_{q_\alpha}^\alpha\}$ and the minimum and maximum regularities in each parametric direction are $k_{\min}^\alpha := \min_{j_\alpha=2,\dots,q_\alpha-1}\{k_{j_\alpha}^\alpha\}$ and $k_{\max}^\alpha := \max_{j_\alpha=2,\dots,q_\alpha-1}\{k_{j_\alpha}^\alpha\}$. Finally, the global regularities are $k_{min} := \min\{k_{\min}^\alpha : \alpha = 1, \dots, \kappa\}$ and $k_{max} := \max\{k_{\max}^\alpha : \alpha = 1, \dots, \kappa\}$.

Finally, we introduce the uni- and multivariate B-Splines space.

Definition 2.1. *The B-Splines space built by the univariate B-Splines basis functions is denoted as:*

$$\mathscr{S}^h := \text{span}(\{N_i\}_{i=1}^n). \tag{2.5}$$

The tensor product B-Splines space, spanned by multivariate B-Splines basis function, is defined by

$$\mathscr{S}^h := \text{span}(\{N_{\boldsymbol{i}}\}_{\boldsymbol{i}\in I}), \tag{2.6}$$

where $I = \{\boldsymbol{i} = (i_1, \dots, i_\kappa) : 0 \leq i_\alpha \leq n_\alpha, 1 \leq \alpha \leq \kappa\}$.

2.1.4 B-Splines geometries

Uni- and multivariate B-Splines are used for the construction of geometries such as curves, surfaces and solids in $\mathbb{R}^d$. We start by giving the definition of univariate B-splines curve and then we generalize the definition to a $\kappa-$dimensional B-Splines object in $\mathbb{R}^d$, for $1 \leq \kappa \leq d$.

Definition 2.2. *Given the univariate B-Splines space* $\mathscr{S}^h$ *over the parametric domain* $\hat{\Omega} = [0,1]$*, and a set of points* $\{\boldsymbol{P}_i\}_{i=1}^n \subset \mathbb{R}^d$*, called* ***control points****, a B-Splines curve in* $\mathbb{R}^d$ *is defined as the linear combination of B-Splines basis functions with the control points assumed as coefficients, i.e:*

$$\boldsymbol{C} : \hat{\Omega} \to \Omega \subset \mathbb{R}^d, \quad \boldsymbol{C}(\xi) = \sum_{i=1}^{n} N_i(\xi)\boldsymbol{P}_i, \tag{2.7}$$

where Ω *represents the curve in* $\mathbb{R}^d$.

The generalization of Def. (2.2) is obtained by exploiting the tensor product nature of the multivariate B-Splines basis functions. By means of multivariate B-Splines basis function it is possible to define geometrical entities of dimension $\kappa \geq 1$ in the physical space $\mathbb{R}^d$, where $1 \leq \kappa \leq d$.

Definition 2.3. *Given an integer* κ *with* $1 \leq \kappa \leq d$ *and knot vectors* $\Xi_\alpha = \{\xi_1^\alpha, \xi_2^\alpha, \ldots, \xi_{n+p+1}^\alpha\}$ *for* $\alpha = 1, \ldots, \kappa$ *let us consider the multivariate B-Splines space* $\mathscr{S}^h$*, Equation 2.6, whose multivariate basis functions* $\{N_{\boldsymbol{i}}\}_{\boldsymbol{i}\in I}$ *are defined over the parametric domain* $\hat{\Omega} = [0,1]^\kappa$*, and a set of* $\prod_{\alpha=1}^{\kappa} n_\alpha$ *control points* $\{\boldsymbol{P_i}\}_{\boldsymbol{i}\in I} \subset \mathbb{R}^d$*. A* κ*-dimensional B-Splines geometrical entity in* $\mathbb{R}^d$ *is defined by the geometrical mapping:*

$$\boldsymbol{x}\colon \hat{\Omega} \to \Omega \subset \mathbb{R}^d, \quad \boldsymbol{x}(\boldsymbol{\xi}) = \sum_{\boldsymbol{i}\in I} N_{\boldsymbol{i}}(\boldsymbol{\xi})\boldsymbol{P_i}, \tag{2.8}$$

where Ω *indicates the geometry in* $\mathbb{R}^d$*. Two-dimensional B-splines entities* ($\kappa = 2$) *represent B-Splines surfaces in* $\mathbb{R}^d$ *for* $d = 2,3$*, while three-dimensional ones* ($\kappa = 3$) *B-Splines solids in* $\mathbb{R}^d$.

The B-Splines geometries directly inherit the properties of the basis functions. In particular the regularity of the B-Splines curve is given by the regularity of the basis functions and their support determines the fact that changing the position of a single control point locally affects the geometry. In the case of a B-Splines curve moving a control point affects the curve in at most $p+1$ elements. Moreover, the control points do not in generally lie on the geometry due to the non-interpolatory nature of the basis functions. Another property of the B-Splines geometry is the so called **convex hull property**, for which for which a B-Splines curve is contained in the union of the convex hulls defined by $p+1$ consecutive control points. Then, by Defs. (2.2-2.3), it follows that applying an affine transformation $\mathscr{A} : \mathbb{R}^d \to \mathbb{R}^d$ to a B-Splines is equivalent to apply it to the control points $\{\mathscr{A}(\mathbf{P}_i)\}_i$ which implies that no change of basis functions is need. The latter property in called **affine covariance**. All the above cited properties hold for $d = 1,2,3$, i.e, for B-Splines curves, surfaces or solids. B-Splines curves satisfy one last important properties named **variation diminishing property**. More precisely, consider the piecewise linear interpolation of the control points, named **control polygon**, the variation diminishing property states that no plane has more intersections with the B-Splines curve then it has with the control polygon. This property is peculiar only to B-Splines curve implies a monotone nature of them. This property has remarkable consequence on the interpolation of discontinuous data. The standard Lagrangian interpolation provides curve whose oscillating behavior increases when increase the degree of the basis functions. While, by considering the data set as control points, B-Splines curves behave monotonically due to the variation diminishing property. Lastly, in the multivariate case we refer to the piecewise multilinear interpolant of the control points $\mathbf{P}_i$ as the **control mesh** in the physical space $\mathbb{R}^d$, which does not coincide with the **physical mesh** in the physical domain Ω induced by the geometrical mapping $\mathbf{x}$, defined in the parameter space. In particular, the image of the elements in the parametric domain of Eq. (2.8) are the elements in the physical domain.

2.2 NURBS basis and geometries

2.2.1 NURBS basis functions

Non-Uniform Rational B-Splines (NURBS) basis function are an extension of the B-Splines polynomial functions. NURBS have been introduced and typically used in CAD system since B-Splines could not represent *exactly* conic sections. In this section we present uni- and multivariate NURBS basis functions.

Definition 2.4. *Given a knot vector* $\Xi = \{\xi_1, \xi_2, \ldots, \xi_{n+p+1}\}$*, a parametric domain* $\hat{\Omega} = [0,1]$*, a set of positive real numbers* $\{\omega_i\}_{i=1}^n$ *and the univariate B-Splines basis* $\{N_i\}_{i=1}^n$ *defined from the knot vector*

Ξ, the $i-th$ NURBS basis functions are defined as:

$$R_i : \hat{\Omega} \to \mathbb{R}, \quad R_i(\xi) = \frac{N_i(\xi)\omega_i}{W(\xi)}, \quad i = 1, \dots, n; \tag{2.9}$$

where W is a scalar piecewise polynomial function, called weighting function, defined as:

$$W : \hat{\Omega} \to \mathbb{R} \quad W(\xi) = \sum_{i=1}^{n} N_i(\xi)\omega_i. \tag{2.10}$$

In Def. (2.4) the hypothesis of positivity of the weights $\{\omega_i\}_{i=1}^n$ is not necessary but is a common assumption in literature and in most application of engineering interest. By convention we will refer to the **degree** of NURBS as to the degree of the B-Splines basis $\{N_i\}_{i=1}^n$. Most of the properties of NURBS basis can be directly deduced from the knot vector Ξ. In particular, the number of basis functions n and the degree p; the basis functions are $\mathscr{C}^\infty-$continuous in the interior of each element of the mesh, while in the knots they are $\mathscr{C}^{p-m_j}-$continuous, with m_j the multiplicity of the knot value $\hat{\xi}_j \in \hat{\Xi}$, for $j = 1, \dots, q$. The partition of unity still be satisfied by the basis $\{R_i\}_{i=1}^n$, and if it is defined from an open knot vector, $R_1(\xi_1) = R_n(\xi_{n+p+1}) = 1$. The support of each basis functions is still compact in $p+1$ knot spans and $p+1$ basis function have support in each knot span. Finally, the NURBS basis functions are pointwise non-negative. We define now the multivariate NURBS as a generalization of the univariate ones.

Definition 2.5. *For $\alpha = 1, \dots, \kappa$, given κ knot vectors $\Xi_\alpha = \{\xi_1^\alpha, \xi_2^\alpha, \dots, \xi_{n_\alpha+p_\alpha+1}^\alpha\}$ and the multivariate B-Splines basis $\{N_{\boldsymbol{i}}\}_{\boldsymbol{i}\in I}$ associated to the knots vectors $\Xi_\alpha, \forall \alpha$, where $I = \{\boldsymbol{i} = (i_1, \dots, i_\kappa) : 0 \le i_\alpha \le n_\alpha, 1 \le \alpha \le \kappa\}$. The $i-th$ multivariate NURBS basis function is defined as:*

$$R_i : \hat{\Omega} \to \mathbb{R}, \quad R_i(\boldsymbol{\xi}) = \frac{N_i(\boldsymbol{\xi})\omega_i}{W(\boldsymbol{\xi})} \quad \forall i \in I, \tag{2.11}$$

where $\{\omega_i\}_{\boldsymbol{i}\in I}$ are a set of weights and

$$W : \hat{\Omega} \to \mathbb{R}, \quad W(\boldsymbol{\xi}) = \sum_{\boldsymbol{i}\in I} \omega_i N_i(\boldsymbol{\xi}), \tag{2.12}$$

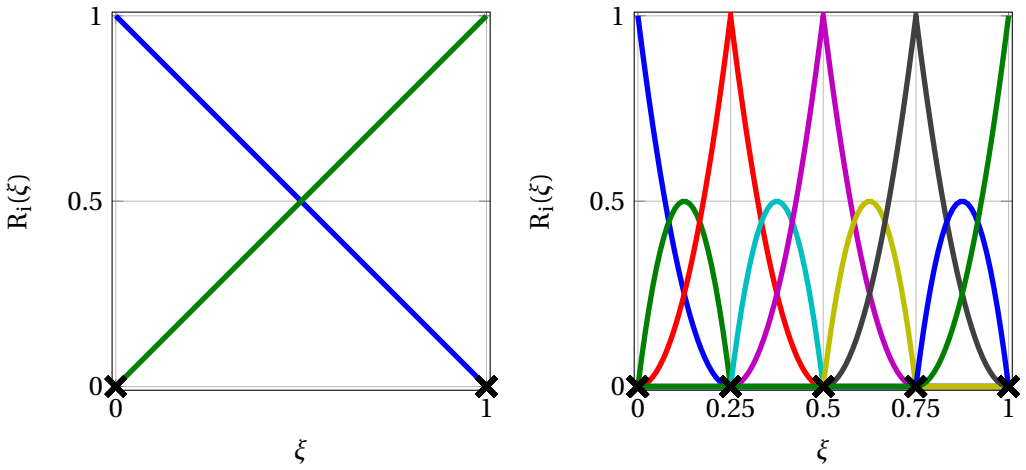
is the weighting function defined over the parametric domain $\hat{\Omega} = [0,1]^\kappa$.

We remark that if in Def. (2.5) the weights ω_i are all unitary (or constant), i.e $\omega_{\boldsymbol{i}} = k \ \forall \boldsymbol{i} \in I$ and $k \in \mathbb{R}$, $R_{\boldsymbol{i}} \equiv N_{\boldsymbol{i}}$ for all $\boldsymbol{i}$, due to the fact that B-Splines form a partition of unity for all $\xi \in \hat{\Omega}$; which means that B-Splines represents a particular case of NURBS. The properties of multivariate NURBS can be deduced by the ones of the related multivariate B-Splines basis and the tensor product structure.

Definition 2.6. *The NURBS space over the parametric domain $\hat{\Omega}$ uni- and multivariate case, respectively, reads:*

$$\mathcal{N}^h := \text{span}(\{R_i\}_{i=1}^n), \tag{2.13}$$

$$\mathcal{N}^h := \text{span}(\{R_{\boldsymbol{i}}\}_{\boldsymbol{i}\in I}). \tag{2.14}$$



(a) $\Xi = \{\{0\}^2, \{1\}^2\}$ (b) $\Xi = \left\{\{0\}^3, \left\{\frac{1}{4}\right\}^2, \left\{\frac{1}{2}\right\}^2, \left\{\frac{3}{4}\right\}^2, \{1\}^3\right\}$

Figure 2.1: Examples of (a) first order basis functions, and (b) second order basis functions, associated to two different open knots vectors.

Example 2.7. *In Figure (2.1) we show two examples of basis functions associated to two different knot vectors. Basis functions (2.1(a)) are associated to the knot vector* $\Xi = \{\{0\}^2, \{1\}^2\}$*, while basis functions (2.1(b)) are associated to* $\Xi = \left\{\{0\}^3, \left\{\frac{1}{4}\right\}^2, \left\{\frac{1}{2}\right\}^2, \left\{\frac{3}{4}\right\}^2, \{1\}^3\right\}$.

2.2.2 NURBS geometries

As discussed in the latter section, NURBS basis have the ability to exactly represent a wide range of geometries that polynomials cannot. In this section we present two different point of views in order to understand NURBS entities: a **geometric point of view** and an **algebraic point of view**.

2.2.2.1 The geometric point of view

From the geometrical point of view, a NURBS geometry in $\mathbb{R}^d$ is a projection of B-Splines in $\mathbb{R}^{d+1}$ onto a $d-$dimensional hypersurface. In particular, conic sections, e.g. circles and ellipses, can be represented exactly by projective transformations of piecewise quadratic B-Splines. The obtained object will be denoted by $\Omega \subset \mathbb{R}^d$.

Let $\mathscr{C}(\xi)$ be a NURBS curve obtained by projective transformation of a B-Splines curve, $\mathscr{C}^\omega(\xi)$. In this context the B-Splines curve is called **projective curve** associated to **projective control points** $\{\mathbf{P}_i^\omega\}$, while the terms *curve* and *control points* are reserved for the NURBS $\mathscr{C}(\xi)$ and the points $\{\mathbf{P}_i\}$ respectively. The control points for the NURBS curve are obtained in the following manner:

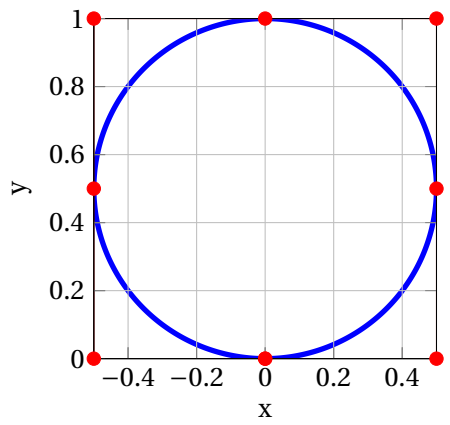
$$\begin{aligned} (\mathbf{P}_i)_j &= (\mathbf{P}_i^\omega)_j/\omega_i, \quad j = 1,2,\ldots,d, \\ \omega_i &= (\mathbf{P}_i^\omega)_{d+1}, \end{aligned} \tag{2.15}$$

where $(\mathbf{P}_i)_j$ is the $j-$th component of the vector $\mathbf{B}_i$ and ω_i is the $i-$th weight.

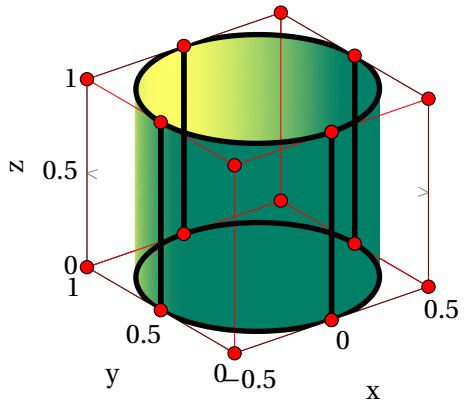
Definition 2.8. *Given the weighting function W, Eq. (2.10), The NURBS curve* $\mathscr{C} : \hat{\Omega} \to \Omega \subset \mathbb{R}^d$ *is given component-wise by:*

$$(\mathscr{C}(\xi))_j = \frac{(\mathscr{C}^\omega(\xi))_j}{W(\xi)} \quad \forall\, j = 1,2,\ldots,d. \tag{2.16}$$

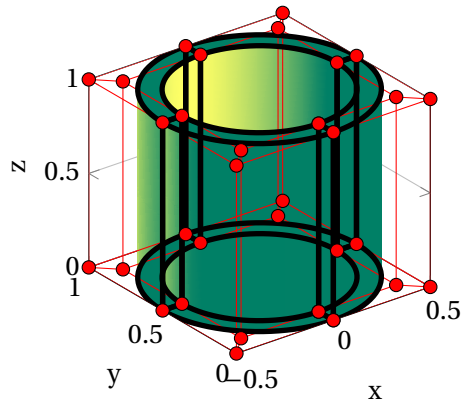
We remember that $\mathscr{C}(\xi)$ is a piecewise rational function within each element is the quotient of two polynomials.



(a) A circle



(b) A cylinder



(c) A tube

Figure 2.2: Examples of NURBS geometrical Objects. (a) A circle obtained using the basis functions shown in Figure (2.1(b)). (b) Cylinder obtained by extrusion of the circle, the multivariate basis functions is the tensor product between basis functions shown in Figure (2.1). (c) Solid tube obtained from the cylinder by adding a third parametric direction in order to obtain the thickness, the multivariate basis functions is the tensor product of the basis functions shown in Figure (2.1(b)) with (two times) the basis functions in Figure (2.1(a)) (once for the height and once for the thickness).

2.2.2.2 The algebraic point of view

In section 2.2.1 we constructed the NURBS basis functions from B-Splines basis functions associated to some knot vectors, Defs. (2.4-2.5). This way of constructing NURBS implies that every property of B-Splines basis functions, hold for NURBS basis functions. The definition of NURBS curve is done in the same way as for B-Splines curve, Def. (2.2).

Definition 2.9. *Let $\{R_i\}_{i=1}^n$ be a univariate NURBS as in Def. (2.4), and $\{\boldsymbol{P}_i\}_{i=1}^n$ the control points computed as in Eq. 2.15. The equation for a NURBS curve $\mathscr{C} : \hat{\Omega} \to \Omega \subset \mathbb{R}^d$ reads:*

$$\mathscr{C}(\xi) = \sum_{i=1}^{n} R_i(\xi)\boldsymbol{P}_i. \tag{2.17}$$

The Def. (2.9) implies that the manipulation of NURBS curve is equivalent to the manipulation of B-Splines curve, i.e, to change shape of the curve it need only to move the control points.

To extend Def. (2.9) to the general case we remark that the numbers of control points needed to construct a NURBS geometry defined in the parametric domain $\hat{\Omega} = [0,1]^\kappa$, is $\mathbf{n}_\kappa = \prod_{\alpha=1}^{n} n_\alpha$, where n_α is the number of basis functions in the direction α.

Definition 2.10. *Given $\{R_{\boldsymbol{i}}\}_{\boldsymbol{i}\in I}$ a multivariate NURBS and a set of control points $\{P_{\boldsymbol{i}}\}_{\boldsymbol{i}\in I}\subset\mathbb{R}^d$, a NURBS geometry in $\mathbb{R}^d$ is represented by the geometrical mapping:*

$$\boldsymbol{x}:\hat{\Omega}\to\Omega\subset\mathbb{R}^d,\quad \boldsymbol{x}(\boldsymbol{\xi})=\sum_{\boldsymbol{i}\in I}=R_{\boldsymbol{i}}(\xi)\boldsymbol{P_i} \tag{2.18}$$

An important observation is that NURBS geometries are defined as a **map** from the parametric domain $\hat{\Omega}=[0,1]^\kappa$ into the physical space $\mathbb{R}^d$, to which the control points belong. In other words, a NURBS geometry is a linear combination of basis functions and control points.

Example 2.11. *In Figure (2.2) we show respectively a NURBS curve, a NURBS surface, and a NURBS solid.*

2.2.2.3 *Jacobian of the mapping*

In view of the used of NURBS for the solution of PDEs, we introduce and analyze the **deformation tensor** and the **Jacobian** of the map $\boldsymbol{x}$, Eq. (2.18), since the latter map will be used to represent the physical domain Ω, while the associated NURBS basis function will be used in the analysis part. From now on we will adopt the following restriction: the dimension κ of the parametric domain is less then or equal to the dimension d of the physical space, i.e, $\kappa\leq d$.

Definition 2.12. *We define the deformation tensor of the geometrical mapping (2.18) as:*

$$T:\Omega\to\mathbb{R}^{d\times d},\quad T_{ij}(\boldsymbol{\xi}):=\frac{\partial \boldsymbol{x}_i}{\partial\xi_j}(\boldsymbol{\xi}),\ \ \forall i,j=1,2,\dots,d, \tag{2.19}$$

and the Jacobian of the mapping as:

$$J:\hat{\Omega}\to\mathbb{R},\quad J(\boldsymbol{\xi}):=\det(T(\boldsymbol{\xi})). \tag{2.20}$$

NURBS geometries defined by maps with non-constant sign Jacobian will be referred as **degenerate** geometries and it appears when some control points cross each others. From a geometrical point of view there will be non impediment in considering degenerate geometries, however we assume that the following condition must be satisfied by the NURBS geometries considered:

$$J(\boldsymbol{\xi})>0,\text{a.e in }\hat{\Omega}\text{ and }J(\boldsymbol{\xi})=0\text{ in }\hat{\Omega}_{J_0}, \tag{2.21}$$

where $\hat{\Omega}_{J_0}:=\{\boldsymbol{\xi}\in\hat{\Omega}:J(\boldsymbol{\xi})=0\}$ and $|\cdot|$ is the measure with respect to the topology of the parameter space)))). Moreover, we assume that the geometrical map given in Eq. (2.18) is invertible, with smooth inverse on each element of the physical mesh (see section 2.1.4). Finally, we can define the general space of NURBS in the domain Ω as push forward of the space $\mathcal{N}_h=\text{span}(\{R_{\boldsymbol{i}}\}_{\boldsymbol{i}\in I})$, Def. (2.6), by using the following relation:

$$\mathscr{R}_{\boldsymbol{i}}=R_{\boldsymbol{i}}\circ\boldsymbol{x}^{-1},\ \ \forall i\in I=(\boldsymbol{i}=(i_1,i_2,\dots,i_\kappa):0\leq i_\alpha)\leq n_\alpha,1\leq\alpha\leq\kappa\}. \tag{2.22}$$

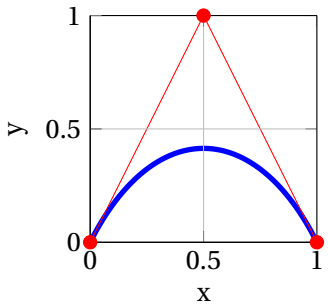
where $\{\mathscr{R}_{\boldsymbol{i}}\}_{\boldsymbol{i}\in I}$ denotes the NURBS basis in the physical domain.

Definition 2.13. *The NURBS space in the domain Ω is define as:*

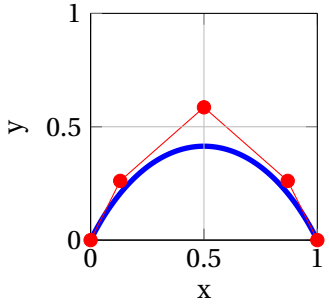
$$\mathscr{V}^h:=\text{span}(\{\mathscr{R}_{\boldsymbol{i}}\}_{\boldsymbol{i}\in I}).$$

2.2.3 *h*-,*p*- and *k*-refinements

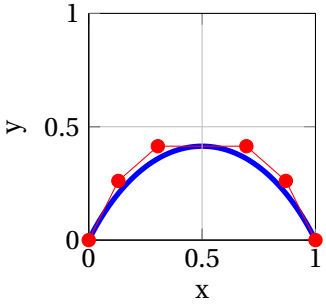
B-Splines and NURBS can be improved by refinement in order to build more detailed geometries. Three different procedures are possible, two, $h-$ and $p-$refinement, are similar with those used in FEA with Lagrangian basis. The third one, called k-refinement, is the one that make the use of B-Splines and NURBS appealing. In particular with the $k-$refinement one have the control not only over the element size and the basis order but also on the global continuity of the basis. Moreover, the enriched B-Splines or NURBS will leave the underlying geometry and its parametrization unchanged. In this section we present the three mentioned possible refinements for univariate B-Splines basis. The extension to the multivariate case is easily obtained due to the tensor product structure.



(a) Starting curve



(b) Curve obtained by $k-$refinement



(c) Curve obtained by inserting first a knot and then by elevation the order

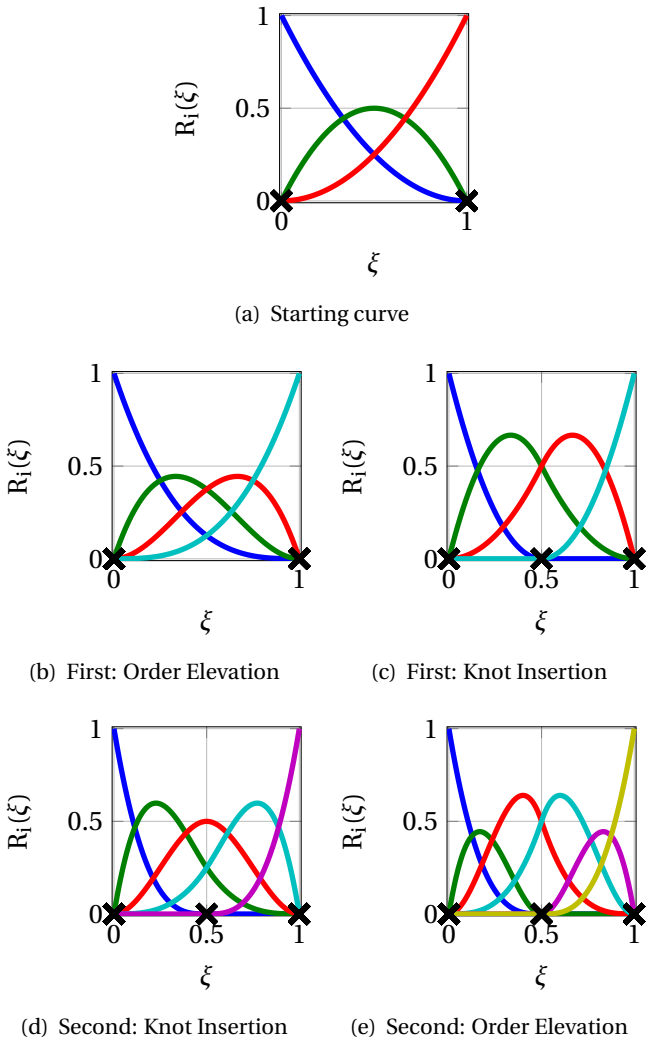
Figure 2.3: Starting from curve (a) we compare the resulting control points position after: (b) a $k-$refinement, and (c) a knot insertion followed by the order elevation. In Figure (2.4) we show the changes of the basis functions each steps of the two different refinements.

2.2.3.1 Knot insertion: h-refinement

Let $\{N_i\}_{i=1}^{n}$ be a B-Splines basis defined by the knot vector $\Xi = \{\xi_1, \xi_2, \ldots, \xi_{n+p+1}\}$. Let us consider a new knot $\overline{\xi} \in [\xi_j, \xi_{j+1})$ to be inserted in Ξ. The recursive formula (2.1-2.2) must now be applied to the *extended* knot vector $\overline{\Xi} := \{\xi_1, \xi_2, \ldots, \xi_j, \overline{\xi}, \xi_{j+1}, \ldots, \xi_{n+p+1}\}$ in order to obtain a set of $n+1$ basis functions. The resulting enriched functions space, $\overline{\mathscr{S}}^{\overline{h}} := \text{span}(\{\overline{N}_i\}_{i=1}^{n+1})$ is nested in $\mathscr{S}^h := \text{span}(\{N_i\}_{i=1}^{n})$. Now, in the case of a curve in $\mathbb{R}^d$ a new set of of control points and weights is defined for the new basis in order to obtain a curve geometrically and parametrically equivalent to the original one. Finally, we remark that $h-$refinement could reduce the regularity of the basis functions when inserting an already existing knot in Ξ. Moreover, with this kind of refinement not only the number of element increases but also the number of basis functions and consequently the number of control points.

2.2.3.2 Order elevation: p-refinement

Order elevation, or $p-$refinement, is made in such a manner to preserve the the regularity of the basis functions in whole domain. It follows that the multiplicity of each existing knot in Ξ has to be increased by one without inserting any additional knot. For details about algorithms, see [22]. As for $h-$refinement the order elevation procedure leads to the generation of nested function spaces, i.e, $\mathscr{S}^h \subset \overline{\mathscr{S}}^{\overline{h}}$.



(a) Starting curve

(b) First: Order Elevation

(c) First: Knot Insertion

(d) Second: Knot Insertion

(e) Second: Order Elevation

Figure 2.4: (a)-(b)-(d) Enrichment of the basis functions (a) by means of $k-$refinement. (a)-(c)-(e) Refinement of basis functions (a) by first inserting a new knot and then by elevating the polynomial order.

2.2.3.3 k-refinement

The last refinement call, $k-$refinement, is peculiar to B-Splines and NURBS. It consists to firstly elevating the order of the starting basis of degree p to a degree $q > p$ and secondly to insert new knot value $\overline{\xi}$ in such a way that the basis functions are $\mathscr{C}^{q-1}$-continuous in $\overline{\xi}$. Let us consider to perform first the knot insertion and only in a second time perform the order elevation. By inserting first a new knot $\overline{\xi}$ the basis function in $\overline{\xi}$ are only $\mathscr{C}^{p-1}-$continuous.

In Figure (2.3) we show an example of refinement of a curve using two different methods: the first method is to used the $k-$refinement explained in the previous section, while the second method is to use firstly the $h-$refinement, by inserting a new distinct knots, and secondly the $p-$refinement by augmenting by one the multiplicity of each already existing knots. We remark that with the second method of enrichment of the curve (2.3(a)) more knots are used to define the curve compared to the resulting curve obtained by means of the $k-$refinement, Figure (2.3(b)). This implies that more basis functions are used to define the curve (2.3(c)). Curve (2.3(a)) is obtained by using the basis functions shown in Figure (2.4(a)) and each steps of both refinements are shown in Figure (2.4).

2.3 Galerkin method for IGA

We present now the IGA-Galerkin method based on NURBS for the approximation of PDEs. In the following section we apply Galerkin method for IGA to the heat equation as introductory example to the method.

Given a NURBS basis $\{R_{\mathbf{i}}\}_{\mathbf{i}\in I}$ in the parameter domain $\hat{\Omega}\subseteq\mathbb{R}^d$, a real valued function $\hat{u}_h$ can be represented in $\hat{\Omega}$ as:

$$\hat{u}_h:\hat{\Omega}\to\mathbb{R};\quad \hat{u}_h(\boldsymbol{\xi}):=\sum_{\boldsymbol{i}\in I}R_{\boldsymbol{i}}(\boldsymbol{\xi})d_i, \tag{2.23}$$

where $\{d_{\mathbf{i}}\}_{\mathbf{i}\in I}\subset\mathbb{R}$ are the control variables. The subscript $_h$ is referring to the global mesh size and is used here to remark that the functions are defined over the finite dimensional NURBS space, $\mathcal{N}^h$. introduced in Def. (2.6). The inverse of the geometrical mappings (2.18) composed with a function defined in $\hat{\Omega}$, allows us to define a function in the physical domain Ω as:

$$u_h:\Omega\to\mathbb{R},\ \ u_h(\boldsymbol{x})=\hat{u}_h\circ\mathbf{x}^{-1}(\boldsymbol{x})=\sum_{\boldsymbol{i}\in I}R_{\boldsymbol{i}}(\mathbf{x}^{-1}(\boldsymbol{x}))d_i=\sum_{\boldsymbol{i}\in I}\mathscr{R}_{\boldsymbol{i}}d_i. \tag{2.24}$$

The space $\mathcal{V}^h$ (Def. (2.13)) is the space of NURBS functions defined in Ω. The Galerkin method based on NURBS provides an approximated solution of a well posed boundary value problem, in the form (2.24), where the finite dimensional space is assumed to be the NURBS space $\mathcal{V}^h$. In the following section the IGA-Galerkin method for the solution of the heat equation is presented.

2.3.1 Galerkin method for IGA applied to the heat equation

Let $\Omega\subseteq\mathbb{R}^2$ be endowed with a regular boundary $\partial\Omega$. In this section we consider the following parabolic problem :

$$\begin{cases}\dfrac{\partial u}{\partial t}-\nabla\cdot(\mu\nabla u)=f(t)\ \text{ in }\Omega\times(0,T)\\ u=0\ \text{ on }\partial\Omega\times(0,T)\\ u(0)=u_0\ \ \text{in }\Omega.\end{cases} \tag{2.25}$$

In view of the numerically approximation of Eq. (2.25) we need to introduce the weak formulation. The classical procedure is to multiply for each $t>0$ the differential equation by a test function $v=v(\mathbf{x})\in V$ and integrating on Ω. We set $V=H_0^1(\Omega)$ and the weak formulation of the Eq. (2.25) reads:

For each $t>0$, find $u(t)\in V$ s.t.:

$$\left(\frac{\partial u(t)}{\partial t},v\right)_{L^2(\Omega)}+a(u(t),v)=F(v;t)\quad\forall v\in V, \tag{2.26}$$

where $(\cdot,\cdot)_{L^2(\Omega)}$ is the L^2-inner product and the form $a(\cdot,\cdot)$, and $F(\cdot;t)$ are defined as follows:

$$a(v,w)=\int_\Omega\mu\nabla v\nabla w\,d\Omega \tag{2.27}$$

$$F(w;t)=\int_\Omega f(t)w\,d\Omega. \tag{2.28}$$

We recall that a sufficient condition for the existence and uniqueness of the solution of Eq. (2.26) is that the bilinear form $a(\cdot,\cdot)$ is continuous and weakly coercive, i.e,

$$\exists\lambda\geq0,\ \exists\alpha\geq0\ :\ \ a(v,v)+\lambda||v||^2_{L^2(\Omega)}\geq\alpha||v||^2_V,\ \forall v\in V.$$

Moreover, the initial condition and the source term must be square-integrable functions,i.e. $u_0,f\in L^2(\Omega)$. Then the problem (2.25) admits a unique solution $u\in L^2(\mathbb{R}^+;V)\cap\mathscr{C}^0(\mathbb{R}^+;L^2(\Omega))$, with $V=H_0^1(\Omega)$.

We consider now the approximation of the Eq. (2.26) by means of Galerkin method based on NURBS. Thus, we consider $\mathcal{V}^h\subseteq V$ as finite dimensional function space spanned by NURBS basis

functions satisfying the boundary condition. The semi discretization of Eq. (2.26) reads:
for each $t > 0$, find $u_h(t) \in \mathcal{V}^h$ s.t

$$(\dot{u}_h(t), v_h)_{L^2(\Omega)} + a(u_h(t), v_h) = F(v_h; t), \ \forall v_h \in \mathcal{V}^h, \tag{2.29}$$

$$\text{with } u_h(0) = u_{0,h} \text{ in } \Omega, \tag{2.30}$$

where $u_{h,0}$ is the L^2 projection of the function u_0 onto $\mathcal{V}^h$ and the following notation has been adopted: $\dfrac{\partial u}{\partial t} = \dot{u}$.

Given a NURBS basis $\{\mathscr{R}_{i \in I}\}$ for the n_{eq}-dimensional space $\mathcal{V}^h = V \cap \mathcal{N}^h$, the test function v_h, the trial solution of Eq. (2.29) u_h and its time derivative as well, can be written as:

$$v_h = \sum_{A=1}^{n_{eq}} \mathscr{R}_A v_A,$$

$$u_h(t) = \sum_{A=1}^{n_{eq}} \mathscr{R}_A d_A(t),$$

$$\dot{u}_h(t) = \sum_{A=1}^{n_{eq}} \mathscr{R}_A \dot{d}_A(t),$$

where the coefficients $\{d_A(t)\}_{A=1}^{N_h}$ represent the control variables and the unknowns of the problem. Using the bilinearity of the L^2-inner product and of the form $a(\cdot,\cdot)$, and the linearity of $F(\cdot; t)$, the discrete Eq. (2.29) is equivalent to the following discrete problem:

for each $t > 0$, find $\{d_B(t)\}_{B=1}^{N_h}$:

$$\sum_{B=1}^{N_h} \dot{d}_B(t)(\mathscr{R}_B, \mathscr{R}_A)_{L^2(\Omega)} + d_B(t)\, a(\mathscr{R}_B, \mathscr{R}_A) = F(\mathscr{R}_A; t), \quad \forall A = 1, 2, \ldots, n_{eq}. \tag{2.31}$$

The $n_{eq}-$dimensional linear system (2.31) can be rewritten in the compact matrix form as:

$$\mathbf{M}\dot{\mathbf{d}}(t) + \mathbf{A}\mathbf{d}(t) = \mathbf{F}, \tag{2.32}$$

where $\mathbf{M}, \mathbf{A} \in \mathbb{R}^{n_{eq} \times n_{eq}}$ are the mass matrix and the stiffness matrix, respectively, $\mathbf{d} \in \mathbb{R}^{n_{eq}}$ is the vector of unknowns, and $\mathbf{F} \in \mathbb{R}^{n_{eq}}$ is the right hand side vector.

For the time discretization we use the generalized$-\alpha$ method which is a fully implicit discretization scheme with controllable numerical dissipation through the parameter $\rho_\infty \in [0, 1]$, for detail about the time-stepping scheme we refer the reader to [23, 24].

2.3.1.1 *A priori error estimate*

By introducing the finite element space $\mathcal{V}^h$ we introduce an approximation error on the computed solution. In this section we briefly discuss the a prior error estimate for the IGA-Galerkin method.

In section 2.2.3 we present the three possible refinements available for enrich the NURBS basis function. With the $h-$refinement, or knot insertion, one has the possibility to decrease the element size h by introducing suitable new knots without increasing the multiplicity of the existing ones. This kind of refinement is equivalent to the mesh refinement used for the classical finite element method. Moreover under this $h-$refinement the rate of convergence of both IGA based on NURBS and FEA, is the same and it depends only on the order p and not on the global continuity of the NURBS basis functions. For example by considering the Poisson problem with a sufficiently smooth boundary domain and input data for which the solution satisfies $u \in H^r(\Omega)$, with $r \geq p + 1$ ($p \geq 1$) we have:

$$||u - u_h||_{H^1(\Omega)} \leq Ch^p,$$
$$||u - u_h||_{L^2(\Omega)} \leq Ch^{p+1}.$$

Details of the proof and complements can be founds in [6, 5].

2.3.1.2 *Numerical Results*

In this section we show a numerical result for the heat equation in order to verify the previous a priori error estimates. We decided to computed the spatial approximation error at the finial time of the simulation. Let us consider the Eq. (2.25) in the domain Ω represented by the unitary square, i.e. $\Omega = (0,1)^2$. We considering the setting of the problem in such a manner that the exact solution is

$$u(x,y,t) = \sin(\pi x)\sin(\pi y)(\sin(10t))^2\exp(x^2+y^2).$$

In Figure (2.5) we report the errors in the H^1- and L^2-norm and the plot of the exact solution.

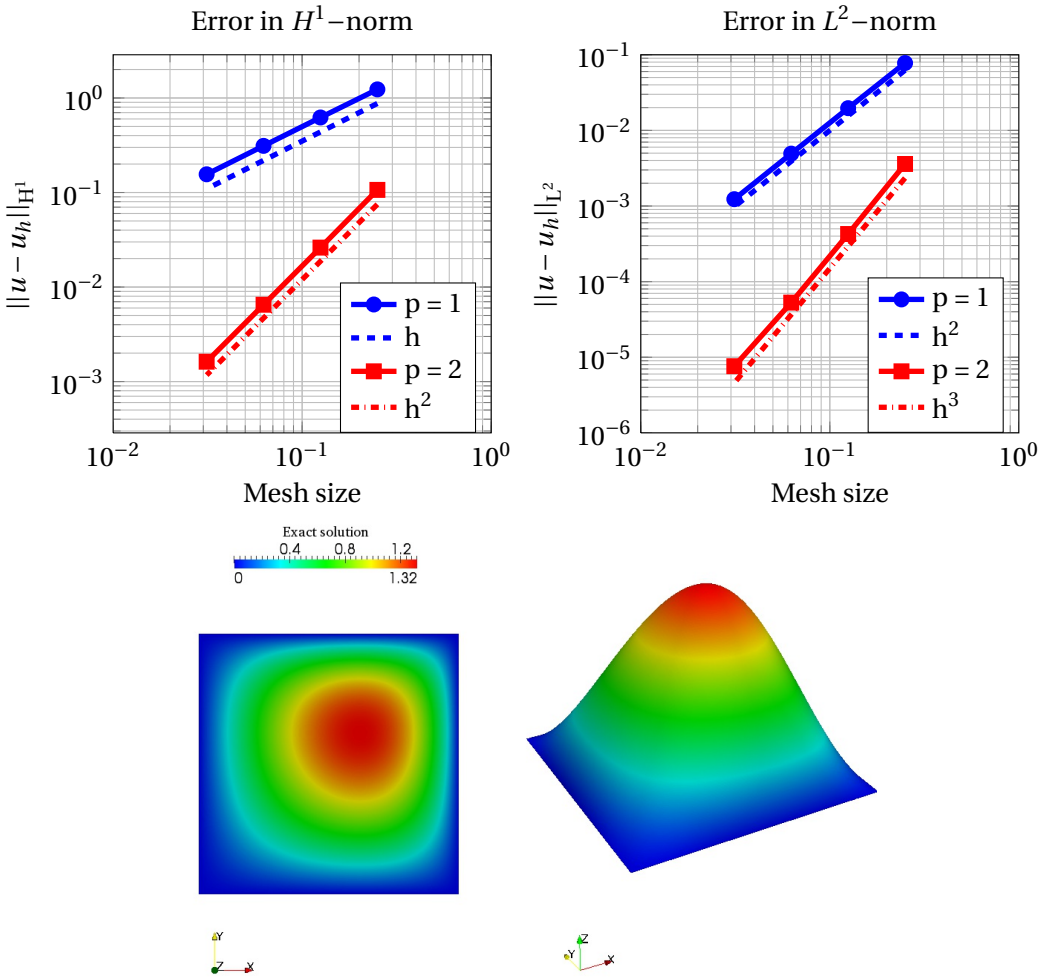


Figure 2.5: Heat equation: Top Left: plot of the errors in norm $H^1(\Omega)$ for $p = 1$(●) and $p = 2$(■). Top Right: plot of the errors in norm $L^2(\Omega)$ for $p = 1$(●) and $p = 2$(■). Bottom: exact solution from different views.

Chapter 3

Benchmark Problem for a Time-Dependent Nonlinear Reaction-Diffusion Equation

3.1 Motivation

In section 1.2 we introduced the monodomain equations as a model for the action potential dynamics of cardiac tissue. This model is a coupling between a nonlinear reaction-diffusion equation and a system of Ordinary Differential Equations. The PDE describes the action potential V and models the propagation of the excitable wave in the cardiac tissue. The ODEs system describes the dynamics of the ions channels and the ions concentrations, characterizing the shape of the action potential and assuring the excitable wave to be a pulse, see section 1.2.3.

The numerical approximation of electrophysiology models by means of the classical Finite Element method (FEM) is widely used, firstly because FEM is the method of choice for solid mechanics and secondly because classical FEM is a suitable method to develop coupling models between the electrophysiology and the mechanics of the heart [25]. In this section we present an issue related to the numerical approximation of the monodomain model, with the focus being on the spatial approximation by means of classical finite element method. Let first recall that the monodomain equations reads:

$$\begin{cases} \chi C_m \dfrac{\partial V}{\partial t} - \nabla \cdot (\mathbf{D}\nabla V) = I_m(\mathbf{u}, V), \text{ in } \Omega \times (0,T) \\ \dfrac{\partial \mathbf{u}}{\partial t} = f(\mathbf{u}, V), \text{ in } \Omega \times (0,T) \\ (\mathbf{D}\nabla V)\cdot \mathbf{n} = 0, \text{ on } \partial\Omega \times (0,T), \end{cases} \tag{3.1}$$

together with initial conditions. In Eq. (3.1) $I_m = I_{app} - \chi I_{ion}$, $\mathbf{n}$ denotes the directed outward unit normal on $\partial\Omega$, and $T > 0$ is the final time. Let Eq. (3.1) be defined in Ω, an open domain in $\mathbb{R}^d$, $d = 1,2,3$. Moreover, we assume that the potential V satisfies a no-flux boundary condition on the boundary of Ω, denoted $\partial\Omega$.

The variational form of Eq. (3.1), for some admissible weight function $w \in \mathcal{W}$ and for a prescribed value of $\mathbf{u}$, reads:

$$\int_\Omega \left(\chi C_m \dot{V} w - I_m(\mathbf{u}, V)\, w + (\mathbf{D}\nabla V)\cdot \nabla w\right) \mathrm{d}\Omega = 0, \ \forall\, w \in \mathcal{W}, \tag{3.2}$$

where $\dot{V} = \frac{\partial V}{\partial t}$. As we are interested in numerically solving Eq. (3.1) using FEM, the Galerkin method with Lagrangian piecewise polynomials will be used for the discretization in space of Eq. (3.2). For this reason let us define a piecewise polynomial basis $\{\varphi_i\}_{i=1}^N$ spanning the finite dimensional space $\mathcal{W}^h \subset \mathcal{W}$. The basis $\{\varphi_i\}_{i=1}^N$ is such that the voltage V is interpolated at the nodal values. We denote with the subscript $_h$ the elements of the finite dimensional space $\mathcal{W}^h$.

The discretized form of Eq. (3.2) in the compact matrix form reads:

$$\mathbf{M}\dot{\mathbf{V}}(t) + \mathbf{K}\mathbf{V}(t) = \mathbf{I},$$

where $\mathbf{V}(t) = \{V_a(t)\}_{a=1}^{N}$, $\mathbf{M}, \mathbf{K} \in \mathbb{R}^{N\times N}$ are the mass matrix and the stiffness matrix, respectively, and $\mathbf{I} = \{I_a\}_{a=1}^{N} = \left\{\int_\Omega I_m(V(t), \mathbf{u})\varphi_a\right\}_{a=1}^{N}$. For now on let us drop the dependence of V on t. The components of the vector $\mathbf{I}$ are computed by performing a numerical integration by means of the Gauss quadrature formula, i.e.,

$$I_a = \sum_{p=1}^{Q} (I_m\varphi_a)|_{s_p} w_p, \ \forall a = 1, \dots, n, \tag{3.3}$$

where $\{s_p\}_{p=1}^{Q}$ and $\{w_p\}_{p=1}^{Q}$ are the quadrature points and weights respectively. Our interest is focus on the current term I_m, because it depends on several state variables (voltage, ionic concentration, gating variable, etc.). The voltage variable is stored in the nodal values, while for the ionic variables one has the choice to treat this variables in different ways.

The first way, called **Nodal integration** is to define each ionic variables as nodal variables, and then to evaluate the current at Gauss points. This approach is made in two different manners. One is called *State Variable Interpolation* (SVI) and consists in interpolating the state variables on the Gauss points and then to evaluate the ionic current using interpolated state variable and voltage as:

$$I_a^{SVI} = \sum_{p=1}^{Q} I_m\Big(V_h(s_p), \mathbf{u}_h(s_p)\Big)\varphi_a(s_p) w_p, \ \forall a = 1, \dots, n.$$

Another way, called *Ionic current interpolation* (ICI), is to compute the current density from the nodal state variables and then to interpolate these to the Gauss point:

$$I_a^{ICI} = \sum_{p=1}^{Q} I_{h,m}(s_p)\varphi_a(s_p) w_p, \ \forall a = 1, \dots, n.$$

A completely different way, called **Gauss integration**, is to store directly the ionic variables in the quadrature points, then the current at a Gauss point s_p is computed using the internal variables $\mathbf{u}_p$ stored at that Gauss point and the voltage interpolated at that Gauss point:

$$I_a^{Gauss} = \sum_{p=1}^{Q} I_m\Big(V_h(s_p), \mathbf{u}_p\Big)\varphi_a(s_p) w_p, \ \forall a = 1, \dots, n.$$

In [8] a comparison between the Nodal Integration and Gauss Integration has been made with particular interest on the effect of the two different approaches in approximating the conduction velocity. The result of this comparison is that the the two different ways of integration have consequences on the convergence of the conduction velocity, with respect to the mesh size, that are one the opposite of the other. More precisely, with Gauss integration the conduction velocity is converging to the exact solution from above while, with the nodal integration the convergence is from below. Another important observation is that the conduction velocity converges for a small value of the mesh size, in fact in [8] the approximated conduction velocity converges to the proper one, for both methods of integration, for a mesh size of 20 μm; the simulation were computed on a $2D$ block of dimension 2.5×0.5 cm. Thus, between the conduction velocity and the mesh size there is a strong relation which is independent on the choice of the way the current term is interpolated (3.3). In Figure (3.1) we show the graph displaying the result of the comparison between the state variable interpolation and the ionic current interpolation; the data have been taken from [26]. For both the methodologies, the conduction velocity converges to the true one, from above and by neglecting coarse meshes, the convergence speed is the same.

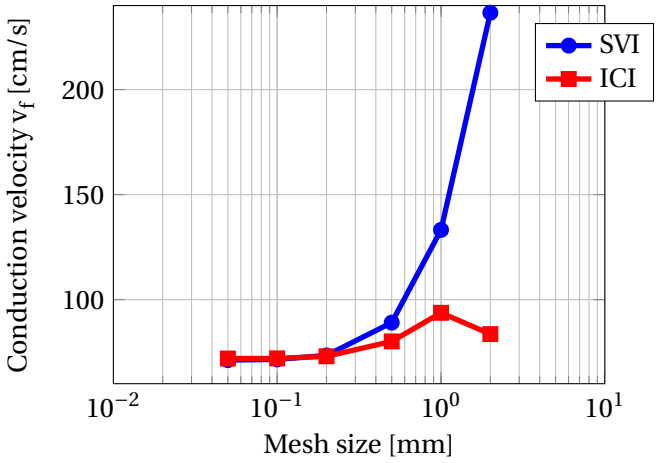


Figure 3.1: Relation between mesh size and conduction velocity approximated by means of classical FEM. ICI (■) approach compare to SVI (●) approach. The data have been taken from [26].

Another possible technique used to solve the system (1.5) is to consider an operator splitting scheme for the time discretization. For example in [27] the second order Strang's splitting [28] is used in order to solve the monodomain equation. The splitting is applied to the strong form of the differential operator of the PDE of the monodomain equations, which reads:

$$\chi C_m \frac{\partial V}{\partial t} - \nabla \cdot (\mathbf{D} \nabla V) = 0, \text{ in } \Omega \times (0, T), \tag{3.4}$$

$$\chi C_m \frac{\partial V}{\partial t} - I_m(\mathbf{u}, V) = 0, \text{ in } \Omega \times (0, T), \tag{3.5}$$

together with initial and boundary conditions. We remark that by adopting the latter operator splitting the diffusion term (3.4) does not depend on the ionic term thus, this splitting naturally decoupled the ionic equations, which become ordinary differential equations that do not need spatial discretization. This implies that the ionic variables do not need to be interpolated. For example, by spatial approximation with the classical finite element method with Lagrangian basis functions, the semi-discretized problem associated to the differential operator (3.4), in the compact matrix form, reads:

$$\mathbf{M}\dot{\mathbf{V}}(t) + \mathbf{K}\mathbf{V}(t) = 0, \tag{3.6}$$

where $\mathbf{V}(t) = \{V_a(t)\}_{a=1}^{N}$, $\mathbf{M}, \mathbf{K} \in \mathbb{R}^{N \times N}$ are the mass matrix and the stiffness matrix respectively. The ODE (3.5) together with the ionic equations, are applied to the finite element nodes, that is:

$$\chi C_m \frac{\mathrm{d}V_A}{\mathrm{d}t} - I_m(\mathbf{u}_A, V_A) = 0, \ \forall t \in (0, T), \tag{3.7}$$

$$\frac{\mathrm{d}\mathbf{u}_A}{\mathrm{d}t} - \mathbf{f}(\mathbf{u}_A, V_A) = 0, \ \forall t \in (0, T). \tag{3.8}$$

where they can be discretized in time and solved directly.

In this chapter we focus our attention on the numerical resolution of the PDE part of the monodomain model, thus, we are interested in the numerical approximation of a specific nonlinear time-dependent reaction-diffusion problem by means of IGA based on NURBS. In particular we study the role of the global continuity of the NURBS basis functions in the convergence of the velocity of the traveling front modeled by this benchmark problem.

3.2 IGA-NURBS for the test problem

The following test problem as been taken from [28]. The problem we consider is a nonlinear time-dependent reaction-diffusion problem whose exact solution, denoted u_{ex}, is known. Let $\Omega = (0,1)^2$

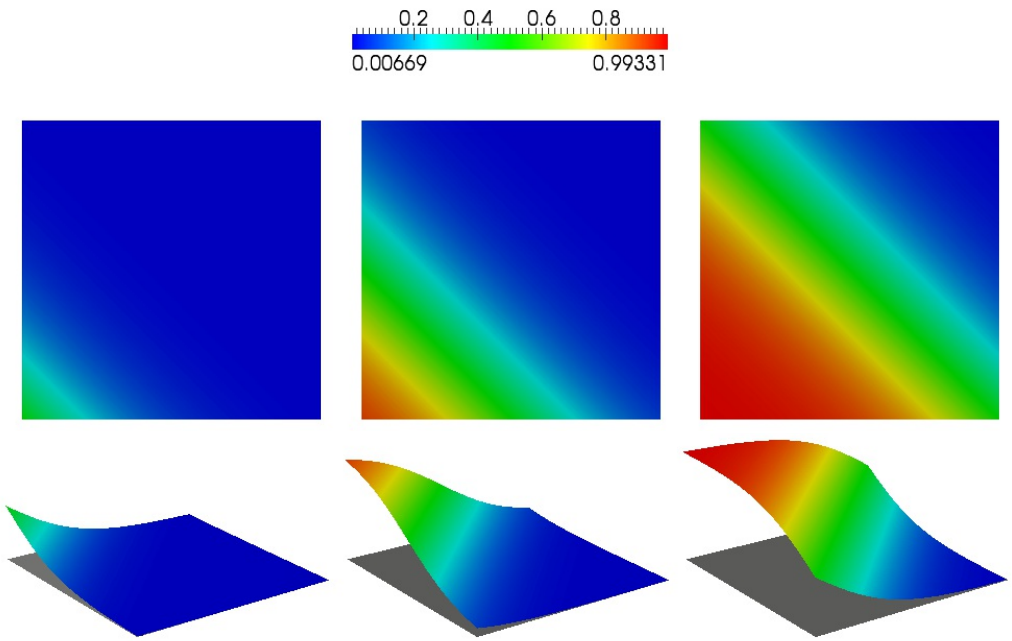


Figure 3.2: The exact solution (3.10) at different times. From left to right; $t = 0$, $t = 0.5$, and $t = 1$.

be the physical domain. The test problem reads:

$$\begin{cases} \dfrac{\partial u}{\partial t} - \mu \Delta u - \dfrac{1}{\mu} u^2 (1-u) = 0, \text{ in } \Omega \times (0,1), \\ \mu \nabla u \cdot \mathbf{n} = 0, \text{ on } \partial\Omega \times (0,1), \\ u_0 = u_{ex}(0), \text{ in } \Omega, \end{cases} \tag{3.9}$$

where $\mu = 0.1$ is the isotropic diffusion coefficient and $\partial\Omega$ is the boundary of the domain Ω. The exact solution u_{ex} is as follow:

$$u_{ex}(x,y,t) = \frac{1}{1 + e^{\frac{1}{2\mu}(x+y-t)}}. \tag{3.10}$$

The function u_{ex} describes a traveling front propagating diagonally in the plane. In Figure (3.2) we report the exact solution (3.10) at different times.

3.2.1 Space-time discretization

In this section we present the spatial discretization based on NURBS. We start by considering the weak form of Eq. (3.9) by multiplying the strong form of the problem by a weighting function $v \in \mathcal{V}$ and by integrating over Ω. The variational formulation of Eq. (3.9) reads:

find $u(t) \in \mathbf{V}$ such that, $u(0) = u_0$ in Ω and

$$(\dot{u}(t), v)_{L^2(\Omega)} + a(u(t), v) - \mathcal{N}(u(t), v) = 0, \ \forall v \in \mathbf{V}, t \in (0,1), \tag{3.11}$$

where $(\cdot,\cdot)_{L^2(\Omega)}$ is the L^2-inner product over the domain Ω, $\dot{u} = \frac{\partial u}{\partial t}$, and the variational forms $a(\cdot,\cdot)$ and $\mathcal{N}(\cdot,\cdot)$ are defined as follow:

$$a(u,v) := (\mu \nabla u, \nabla v)_{L^2(\Omega)}, \tag{3.12}$$

$$\mathcal{N}(u,v) := \left(\frac{1}{\mu} u^2 (1-u), v \right)_{L^2(\Omega)}, \tag{3.13}$$

where we drop the explicit dependence of u on t.

Let now $\mathcal{V}^h = \mathcal{V} \cap \mathcal{N}^h \subset \mathcal{V}$ be a finite dimensional function space, where $\mathcal{N}^h$ is a finite dimensional space defined by NURBS basis functions; we approximate the Eq. (3.11) in space by means of the Galerkin method as follows:

find $u_h(t) \in \mathcal{V}^h$ such that,

$$(\dot{u}_h(t), v_h)_{L^2(\Omega)} + a(u_h(t), v_h) - \mathcal{N}(u_h(t), v_h) = 0, \ \forall v_h \in \mathcal{V}^h, \ t \in (0,1), \tag{3.14}$$

for $u_h(0) = u_{h,0}$ in Ω, where $u_{h,0}$ is the L^2 projection of u_0 onto $\mathcal{V}^h$. We keep the same notation introduced in section 2.3.1 thus, the NURBS basis functions defining the discrete space $\mathcal{V}^h$ of dimension n_{eq} are denoted by $\mathscr{R}_A$, for $A = 1, \ldots, n_{eq}$, and the weighting function v_h, the trial solution u_h, and its time derivative can be written as:

$$v_h = \sum_{A=1}^{n_{eq}} \mathscr{R}_A v_A,$$
$$u_h(t) = \sum_{A=1}^{n_{eq}} \mathscr{R}_A d_A(t),$$
$$\dot{u}_h(t) = \sum_{A=1}^{n_{eq}} \mathscr{R}_A \dot{d}_A(t),$$

respectively, where the coefficients d_A represent the control variables. Moreover, we will denote by $\mathbf{M}$ the mass matrix and by $\mathbf{A}$ the stiffness matrix. The discrete nonlinear vector will be denoted by $\mathbf{N}(\mathbf{d}(t))$. The discrete problem (3.14) can be equivalently written in the compact form as:

$$\mathbf{M}\dot{\mathbf{d}}(t) + \mathbf{A}\mathbf{d}(t) - \mathbf{N}(\mathbf{d}(t)) = 0, \ t \in (0,1), \tag{3.15}$$

with $\mathbf{d}(0) = d_0 = u_{h,0}$. One possible way to solve the nonlinear diffusion-reaction equation is to treat separately the diffusion term and the nonlinear reaction term, i.e, by adopting an operator splitting technique. Detailed description of the operator splitting is given in [28]. In this work we solve the full problem (3.9) without splitting and, as for the heat equation (section 2.3.1), we use the generalized$-\alpha$ method [23, 24].

Let us divide the time interval into a n_{ts} time intervals of size $\Delta t^n := t^{n+1} - t^n$, with $t^0 = 0$. We denote the vector of control variables and its time derivatives at time step t^n by, $\mathbf{d}^n = \mathbf{d}(t^n) = \{d_A(t^n)\}_{A=1}^{n_{ts}}$ and $\dot{\mathbf{d}}^n = \dot{\mathbf{d}}(t^n) = \{\dot{d}_A(t^n)\}_{A=1}^{n_{ts}}$ respectively. The generalized$-\alpha$ method scheme for the problem (3.9) reads:

at each time t^n, given the vectors $\mathbf{d}^n$ and $\dot{\mathbf{d}}^n$, find the vectors $\mathbf{d}^{n+1}$ and $\dot{\mathbf{d}}^{n+1}$ such that:

$$\begin{aligned}
\mathbf{M}\dot{\mathbf{d}}^{n+\alpha_m} + \mathbf{A}\mathbf{d}^{n+\alpha_f} - \mathbf{N}(\mathbf{d}^{n+\alpha_f}) &= 0, \\
\mathbf{d}^{n+1} = \mathbf{d}^n + \Delta t^n \dot{\mathbf{d}}^n + \gamma \Delta t^n (\dot{\mathbf{d}}^{n+1} - \dot{\mathbf{d}}^n), & \\
\mathbf{d}^{n+\alpha_f} = \mathbf{d}^n + \alpha_f (\mathbf{d}^{n+1} - \mathbf{d}^n), & \\
\dot{\mathbf{d}}^{n+\alpha_m} = \dot{\mathbf{d}}^n + \alpha_m (\dot{\mathbf{d}}^{n+1} - \dot{\mathbf{d}}^n), &
\end{aligned} \tag{3.16}$$

where $t^{n+\alpha_f} := t^n + \alpha_f \Delta t^n$. The resulting nonlinear system (3.16) is then solved by using the Newton method [5]. The parameters α_f, α_m, and γ characterize the method. For example, the method is of second order if $\gamma = \frac{1}{2} + \alpha_m - \alpha_f$ and unconditionally absolutely stable stable if $\alpha_m \geq \alpha_f \geq \frac{1}{2}$. Moreover, the parameters α_m and α_f can be parametrized in term of ρ_∞, which denotes the limit value of the spectral radius of the amplification matrix for $\Delta t \to \infty$ as:

$$\alpha_m = \frac{1}{2}\left(\frac{3-\rho_\infty}{1+\rho_\infty}\right), \ \alpha_f = \frac{1}{1+\rho_\infty}, \ \gamma = \frac{1}{2} + \alpha_m - \alpha_f, \ \rho_\infty \in [0,1],$$

The key role of the parameter ρ_∞ is to control the high-frequency dissipation, as show in [29].In particular the method annihilates in one step the high numerical frequencies if $\rho = 0$ and preserves the high frequencies if $\rho = 1$. The latter value is in general not recommended because the preservation of high frequencies may spoil long time simulation of coupled systems. We adopted the value of $\rho_\infty = 0.5$ for all simulations, which has been shown to be effective in many problems, for example in turbulence computations [30].

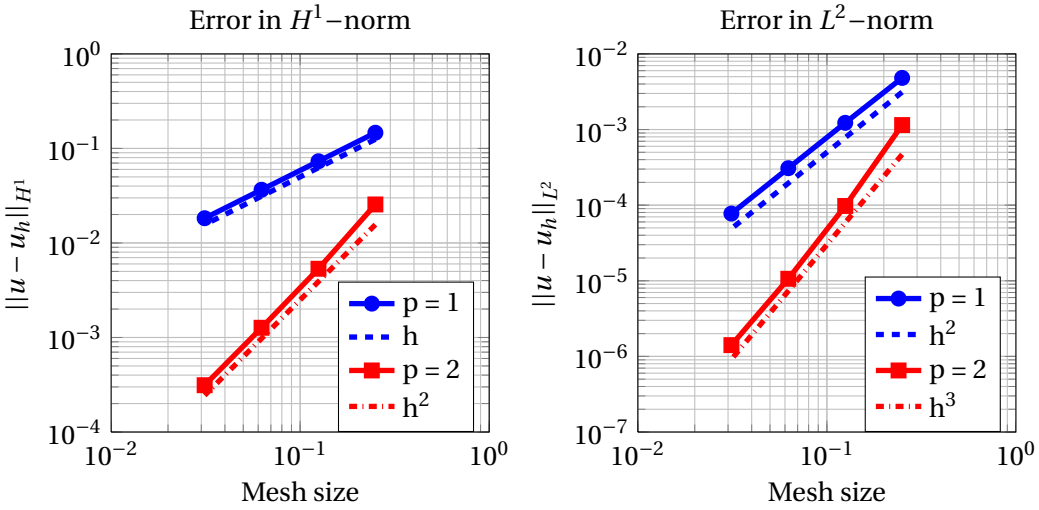


Figure 3.3: Test problem: Left: plot of the errors in norm $H^1(\Omega)$ for $p = 1$(●) and $p = 2$(■). Right: plot of the errors in norm $L^2(\Omega)$ for $p = 1$(●) and $p = 2$(■).

3.2.2 Convergence test

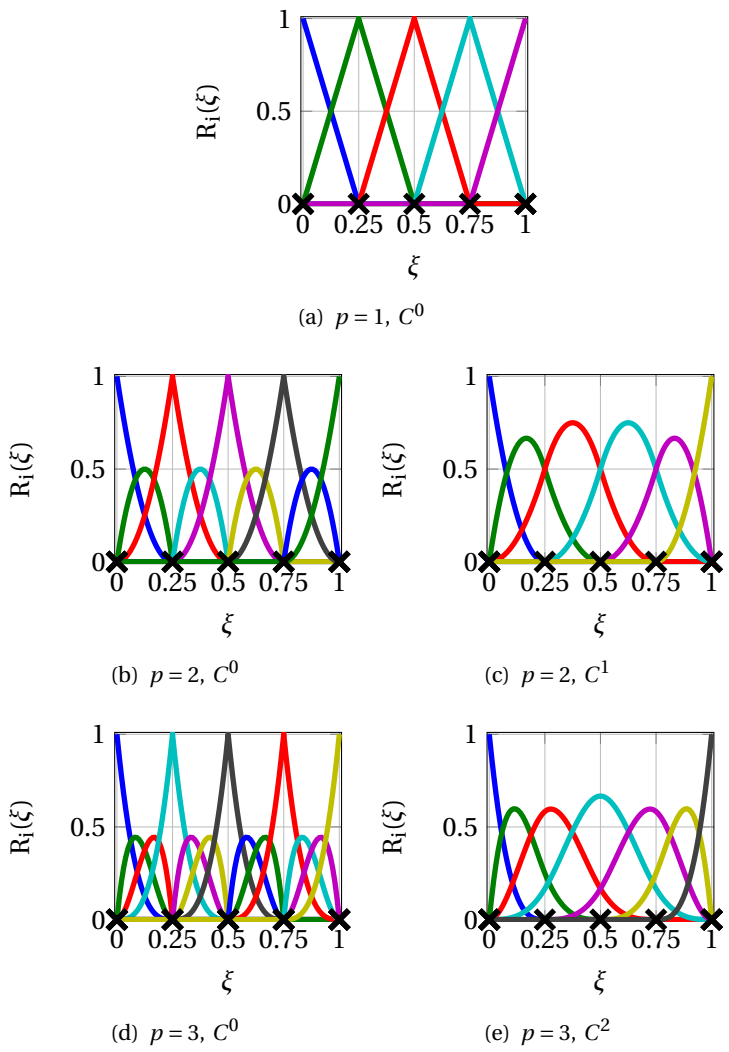
Before going further with the analysis, the *C++* code implementing the discrete problem (3.15) has been validated by computing the L^2- and H^1-norms of the spatial error between the approximated solution u_h and the exact solution (3.10) at the time of the simulation, $t = 0.5$, for different mesh sizes. In particular, we consider the polynomial degrees $p = 1$ and $p = 2$ with globally C^0-continuous NURBS basis functions and globally C^1-continuous NURBS basis functions respectively. We recall that the order of convergence depends on the polynomial order and is independent of the continuity of the basis functions. In Figure (3.3) we report the errors in H^1- and L^2-norms for the mesh size values: $0.25, 0.125, 0.0625$ and 0.03125. We observe that the convergence order is p in the H^1-norm and $p+1$ in the L^2-norm, which is the equivalent order observed for the heat equation problem, Figure (2.5).

3.3 Convergence of the conduction velocity

In this section we study the role of the global continuity of the NURBS basis functions in the approximation of the traveling front described by Eq. (3.9). In particular we are interested in the relation between the velocity of propagation of the front and the global continuity of the NURBS basis functions used in the computation. For this reason we will consider different meshes of the unit square Ω, and numerically solve Eq. (3.9) by means of NURBS basis functions with different polynomial degrees and different global continuities.

We begin the analysis firstly by approximating the solution of Eq. (3.9) with different polynomial orders and global continuities of the basis functions on a coarse mesh, $h = 0.25$. In particular, we aim at studying the approximation of the contour line of the solution corresponding to the value of 0.2. Because of the fact that the exact solution (3.10) describes a traveling front, the contour line corresponding to the selected value of 0.2, represents the forehead of the propagating front. In Table (3.1) we report the total number of basis functions used in the computation, i.e., the total number of degrees of freedom (DOFs) of the non linear system (3.15) for the chosen settings. In Figure (3.4) we plot the NURBS basis functions used for the corresponding settings of Table (3.1). We recall that the basis functions are the same for both the parametric directions.

This very first analysis shows how the different basis functions approximate the contour line of the solution corresponding to the selected value of 0.2, black solid line in Figure (3.5) top. All the basis functions approximate the exact front with an oscillatory behavior, thus the accuracy of the



(a) $p = 1,\ C^0$

(b) $p = 2,\ C^0$

(c) $p = 2,\ C^1$

(d) $p = 3,\ C^0$

(e) $p = 3,\ C^2$

Figure 3.4: NURBS basis functions used for the computation of the conduction velocity.

Polynomial order	Global continuity	Total number of DOFs
1	C^0	25 (5×5)
2	C^0	81 (9×9)
2	C^1	36 (6×6)
3	C^0	169 (13×13)
3	C^2	49 (7×7)

Table 3.1: Polynomial order, continuity and corresponding number of total degrees of freedom, in parenthesis we report the number of basis functions in both parametric directions.

approximation will be based on the maximal distance of the approximated front to the exact one.

The less accurate approximation of the front is provided by the linear polynomial basis functions, see Figure (3.5) bottom. While, the most accurate approximations are obtained by using high order polynomial basis functions globally C^0-continuous. On the other hand, by increasing the global continuities of the basis functions, the exact front is approximated with less accuracy compare to the one obtained with $p = 2,\ C^0-$ and $p = 3,\ C^0-$continuous basis functions.

Let us now compare the accuracy of the solutions in the approximation of the selected front with respect to the DOFs of the nonlinear system (3.15). In Table (3.1) we remark that the total numbers of DOFs related to quadratic polynomial basis functions globally C^0-continuous are larger by a factor

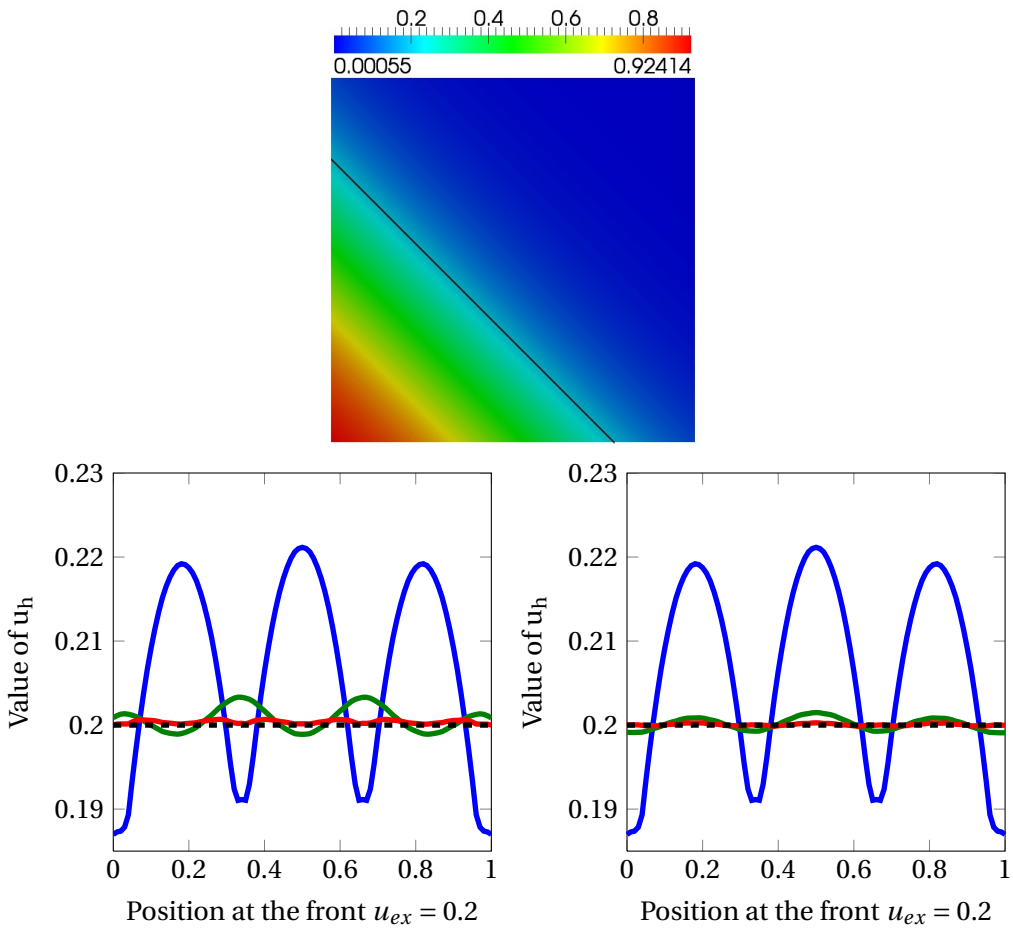


Figure 3.5: Top: Exact Solution (3.10) at time $t = 0.5$ with black solid line highlighting the front $u_{ex} = 0.2$. Bottom: value of the approximated solution u_h computed at the front $u_{ex} = 0.2$ for: $p = 1$, C^0 (—); Left: $p = 2$, C^1 (—) and $p = 2$, C^0 (—). Right: $p = 3$, C^2 (—) and $p = 3$, C^0 (—).

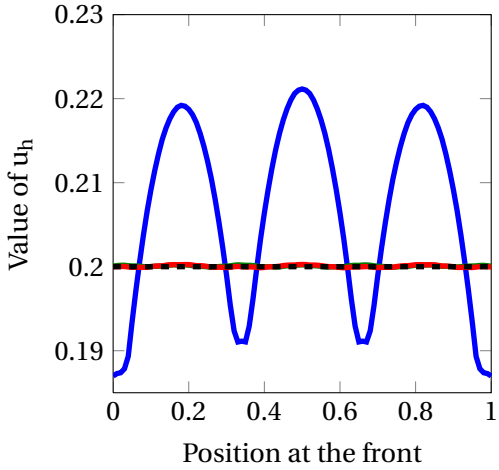


Figure 3.6: Value of the approximated solution u_h computed at the front $u_{ex} = 0.2$ for: $p = 1$, C^0, and $h = 0.25$ (—); $p = 2$, C^1, and $h = 0.125$ (—); $p = 3$, C^0, and 0.25 (—).

2.25 than basis functions with $p = 2$ globally C^1-continuous. Moreover, the basis functions globally C^0-continuous with $p = 3$ are related to a total number of DOFs which is larger by a factor 3.45 than basis functions with $p = 3$, globally C^2-continuous.

Thus, we solve the problem on a less coarse mesh, $h = 0.125$, using $p = 2$, C^1 NURBS basis function. Then, the accuracy obtained is the same compared to the one obtained by solving the problem on the coarse mesh ($h = 0.25$) with $p = 3$, C^0 basis functions, Figure (3.6). We remark that the total number of DOFs for $p = 2$, C^1 and $h = 0.125$ are 100 while for $p = 3$, C^0 and $h = 0.25$ are 169.

Thus, we remark that refining the basis functions (a), Figure (3.4) with a pure $p-$refinement leads to more accurate approximated solution at the front, in only two refinements. On the other hand, the approximation at the front is not as accurate if basis functions (Figure (3.4(a))) are refined by elevating both the polynomial order and the global continuity. Although the convergence of the approximated solution at the front is slower with the $k-$refinement, this method of refinement is more appealing because the total number of degrees of freedom increases considerably less faster in comparison of pure $p-$refinement. Thus, the approximation of the solution at the front may be obtained by optimizing the mesh size and the global continuity of the basis functions in such a manner that the accuracy of the approximated solution at the front is maximized and the dimension of the nonlinear system is minimized.

		Total number of DOFs			
Polynomial order	Global continuity	$h = 0.25$	$h = 0.125$	$h = 0.0625$	$h = 0.03125$
1	C^0	25	81	289	1089
2	C^0	81	289	1089	4225
2	C^1	36	100	324	1156
3	C^0	169	625	2401	9409
3	C^2	49	121	361	1225

Table 3.2: Details of the simulations

The next step of the analysis of test problem (3.9) is the study of the role of high order continuous NURBS basis functions in the convergence of the velocity of the traveling front. In section 3.1 we see that the mesh size is strongly related to the propagation velocity for the monodomain equation using classical finite element method with (Lagrangian) linear polynomial basis functions. In Figure (3.7) we report our findings about the relation between mesh size and conduction velocity, for polynomial orders $p = 1,2,3$ and different continuities. In Table (3.2) all details about the simulations are reported.

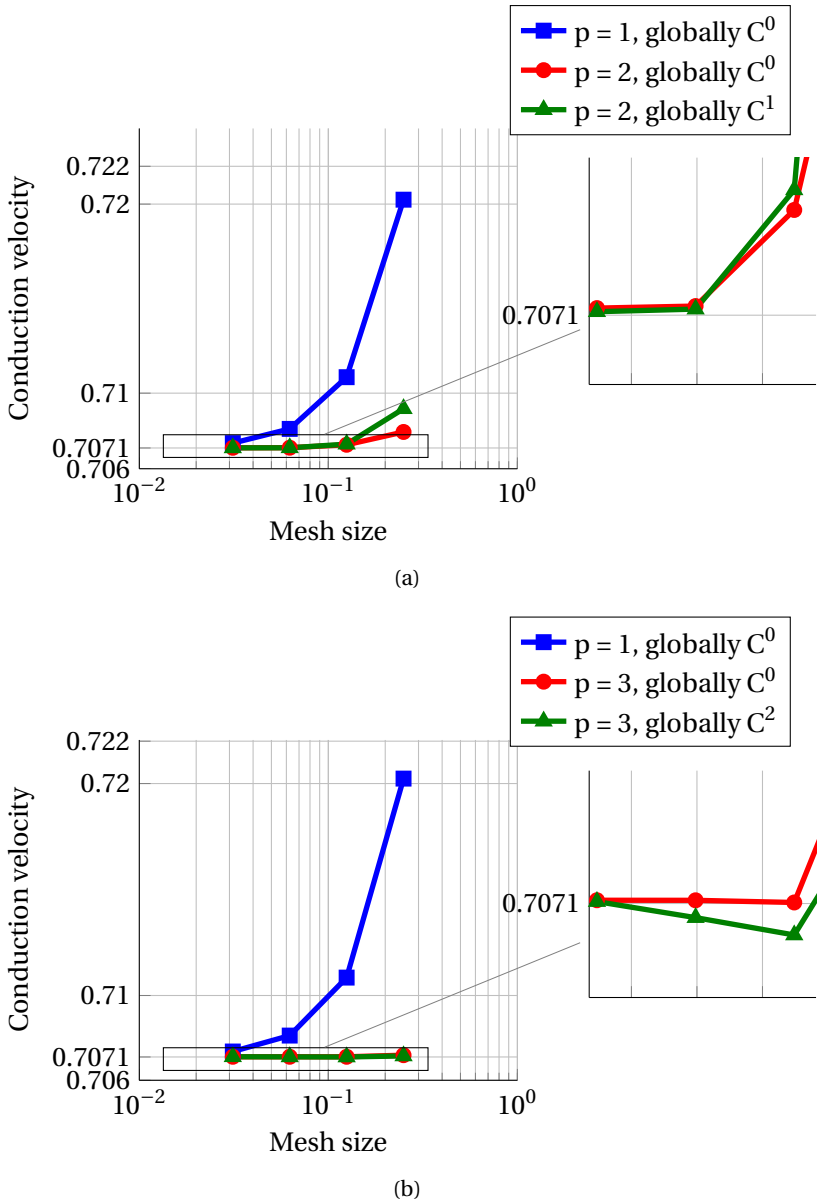


Figure 3.7: Convergence of the conduction velocity with respect to the mesh size: (a) $p=1, C^0$ (■), $p=2, C^0$ (●), and $p=2, C^1$ (▲). (b) $p=1, C^0$ (■), $p=3, C^0$ (●), and $p=3, C^2$ (▲).

We observe that the approximated traveling front obtained with the different basis functions, overestimate the real velocity value of $\frac{\sqrt{2}}{2}$, except for the basis function with $p=3, C^2-$ continuous for which the front is faster only for the coarser mesh. Let now compare the results obtained with $p=2$, C^1-continuous basis functions and with $p=2$, C^0 basis functions. We remark that for the two coarser meshes ($h=0.25$ and $h=0.125$) the conduction velocity obtained with globally C^0 basis functions, is closer to the real one, while for the finer mesh sizes ($h=0.0625$ and $h=0.03125$) the velocities obtained with globally C^1 basis functions are more accurate. The comparison between $p=3$, C^0 and $p=3$, C^2 are different with respect to the latter. Even for coarse meshes the value of the approximated conduction velocity obtained with $p=3$, C^0 and $p=3$, C^2-continuous basis functions is extremely close to the actual velocity. But the more remarkable finding is that the convergence of the value of the approximated conduction velocity to the true one is achieved with a smaller total number of degrees of freedom when the problem is solved using NURBS basis functions with C^{p-1} globally continuous, for $p=2,3$.

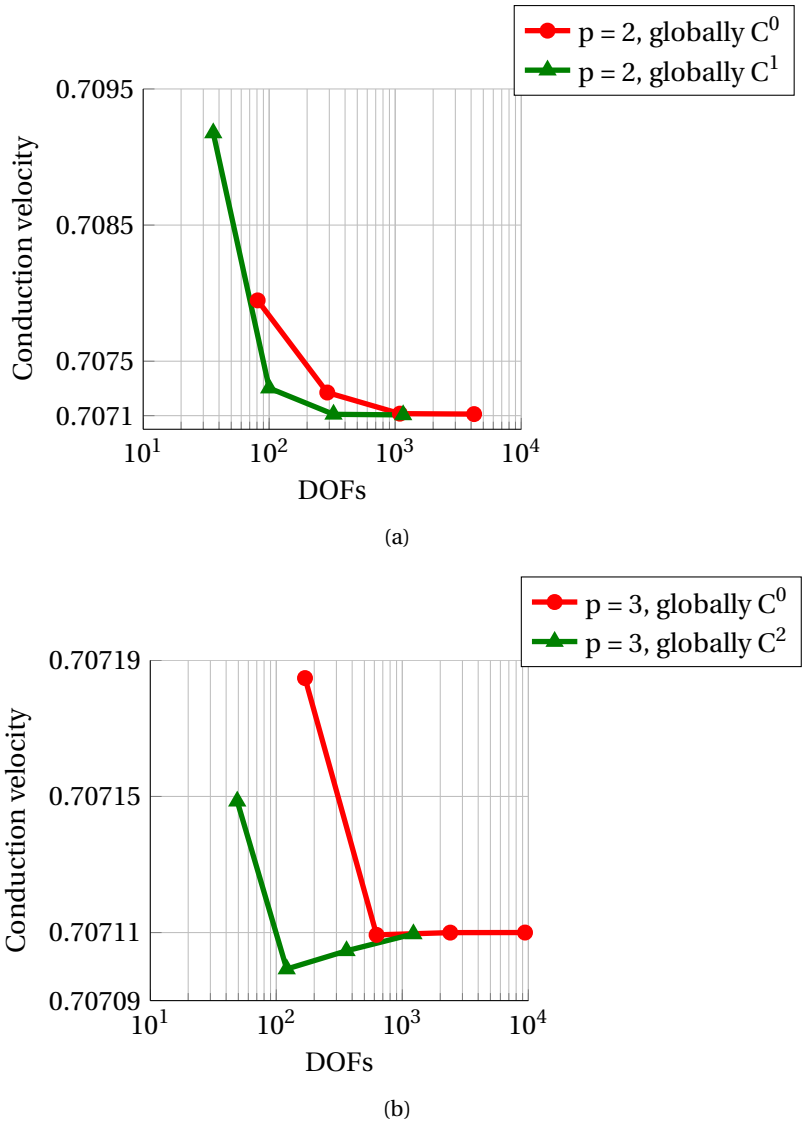


Figure 3.8: Relation between the conduction velocity and the total number of degrees of freedom (DOFs). (a) $p = 2$, C^0 (●) and $p = 2$, C^1 (▲). (b) $p = 3$, C^0 (●) and $p = 3$, C^2 (▲)

In fact, in Figure (3.8) we show the convergence of the value of the numerical conduction velocity with respect to the DOFs of the nonlinear system (3.15). We remark that by increasing the global continuity of the basis functions the approximation of the proper conduction velocity is achieved for a smaller numbers of DOFs with respect to high polynomial order basis functions globally C^0.

After the analysis we made on the convergence of the velocity of the traveling front we can conclude that the best settings of polynomial order and globally continuity is $p = 2$ with globally C^1- continuous NURBS basis functions which provides the the approximation of the true conduction velocity with the smaller total numbers of degrees of freedom of the non linear system (3.15).

Chapter 4

Isogeometric Analysis for the Monodomain Equation

In Chapter 3 we studied the benchmark problem (3.9) which describes a traveling front. The principal issue related to the numerical simulation of the Eq. (3.9) is to approximate the proper conduction velocity. We point out that the more efficiently way, in terms of DOFs of the system, is to use high order continuous basis functions to approximate the actual value of the propagating velocity.

Motivated by this finding, in this chapter we extend our study to the approximation of *traveling pulses* more precisely, our aim is to numerically solve the Monodomain equations by means of the Galerkin method based on NURBS basis functions. In particular, in this chapter we solve the Aliev-Panfilov model [1] presented in section 1.2.6.1 by means of IGA. This two-variable model permits to simulation the electrical activity of the cardiac tissue in a simple and fast way. We recall that the two equations describing the action potential and the recovery variable reads:

$$\frac{\partial V}{\partial t} = \nabla \cdot (D\nabla V) + I_m, \text{ in } \Omega \times I, \tag{4.1}$$

$$\frac{dr}{dt} = \left(\epsilon_0 + \frac{\mu_1 r}{V+\mu_2}\right)(-r - kV(V-a-1)), \text{ in } \Omega \times I, \tag{4.2}$$

where, $\Omega \subset \mathbb{R}^2$, $D = 10^{-4}$ is the *isotropic* conductivity tensor , $I_m = -kV(V-a)(V-1) - Vr + I_{app}$, and the other parameters' values are given in Table (1.1). As initial condition we set $V_0 = 0$ and $r_0 = 1$ in Ω and no-flux condition on $\partial\Omega \times I$ completes the model.

In order to numerical solve the system of equations (4.1-4.2), we adopt the operator splitting technique. The system we obtain reads:

$$\frac{\partial V}{\partial t} - \nabla \cdot (D \cdot \nabla V) = 0, \text{ in } \Omega \times I \tag{4.3}$$

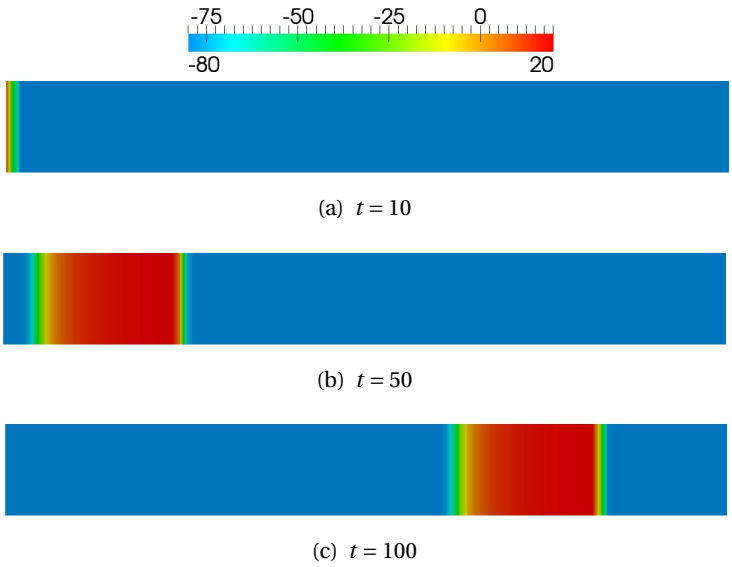
$$\frac{\partial V}{\partial t} - I_m = 0, \text{ in } \Omega \times I, \tag{4.4}$$

$$\frac{\mathrm{d}r}{\mathrm{d}t} - \left(\epsilon_0 + \frac{\mu_1 r}{V+\mu_2}\right)(-r - kV(V-a-1)) = 0, \text{ in } \Omega \times I, \tag{4.5}$$

where Eq. (4.3) is a PDE and Eqs. (4.4-4.4) are ODEs. Thus, by adopting the operator splitting we drop the spatial dependence on the recovery variable. Is this manner we avoid the issue of interpolating the gating variables, for details see Chapter 3. By discretizing Eq. (4.3) by means of the Galerkin method based on NURBS, we obtain the matrix system:

$$\mathbf{M}\dot{\mathbf{V}}(t) + \mathbf{K}\mathbf{V}(t) = 0, \tag{4.6}$$

where, $\dot{\mathbf{V}}(t) = \{V_A(t)\}_{A=1}^{n_{eq}}$, $\mathbf{M}, \mathbf{K} \in \mathbb{R}^{n_{eq} \times n_{eq}}$ are the mass matrix and the stiffness matrix, respectively. The ODEs (4.4-4.5) are applied to the degrees of freedom where are directly solved. More precisely,



(a) $t = 10$

(b) $t = 50$

(c) $t = 100$

Figure 4.1: Example of solution of the system (4.1-4.2) in three different time step.

for all A, $A = 1, \ldots, n_{eq}$ we solve using the explicit Euler method, the following equations:

$$\frac{\mathrm{d}V_A}{\mathrm{d}t} - I_m(r_A, V_A) = 0, \ \forall t \in I, \tag{4.7}$$

$$\frac{\mathrm{d}r_A}{\mathrm{d}t} - \left(\epsilon_0 + \frac{\mu_1 r_A}{V_A + \mu_2}\right)(-r_A - kV_A(V_A - a - 1)) = 0, \ \forall t \in I. \tag{4.8}$$

We recall that the NURBS are not interpolatory and thus the degrees of freedom are in general not on the geometry.

For the numerical scheme we implement the Strang's second order splitting [28]. At the discrete time t^n, the ODEs (3.7-3.8) are solved, then the obtained action potential is used to solve the Diffusion term (3.6). Finally the resulting action potential is used to solve again Eqs. (3.7-3.8).

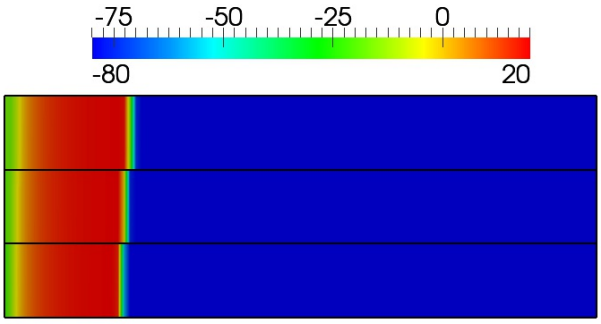
The initialization of the traveling pulse is done by applying a current on the left boundary of Ω. In Figure (4.1) we show an example of the propagation of the excitable wave simulated with $p = 1$, C^0-continuous basis functions and a mesh size of $h = 0.01563$.

In the next sections we are interested in understanding the advantages of using high continuous basis functions in solving the system (4.1-4.2). In particular we focus our attention on four aspects: the approximation of the conduction velocity, the approximation of the action potential duration, the simulation of spiral waves, and the simulation of other dynamics. In the next sections we will compare results obtained with different polynomial orders and different global continuities of the basis functions.

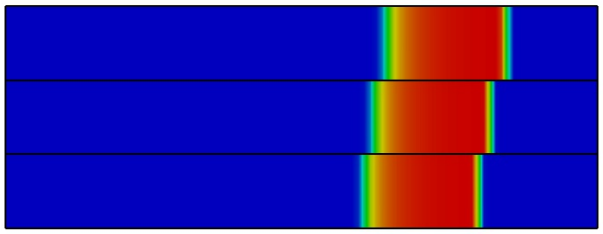
4.1 Convergence of the conduction velocity

In this section we consider the computational domain Ω as a 0.25×2 rectangle. In this section we study the approximation of the conduction velocity for $p = 1$, C^0, $p = 2$, C^1, and $p = 3$, C^2 basis functions. As mentioned in section 3.1, the conduction velocity is strongly related to the mesh size considered for the simulation. It can be easily seen in Figure (4.2) the impact that different mesh sizes have on the velocity of propagation of the traveling pulse. The latter solution have been approximated using $p = 1$, C^0-continuous basis functions and the considered mesh sizes are: $h = 0.01563$, $h = 0.00781$, $h = 0.00521$.

For this analysis we do not take into account high order polynomials basis functions which are globally C^0, because, as shown in Chapter 3, pure $p-$refinement increases the total numbers of de-



(a) $t = 25$



(b) $t = 100$

Figure 4.2: Comparison of the conduction velocity obtained using $p = 1$, C^0 basis functions for three different mesh sizes. In (a) and (b), from top to bottom: $h = 0.01563$, $h = 0.00781$, $h = 0.00521$.

grees of freedom but approximates the true value of the conduction velocity with the same rate of convergence obtained by adopting a $k-$refinement. Thus, by increasing the global continuity of the basis functions, the convergence of the value of the numerical conduction velocity to the actual velocity of propagation, is achieved with a smaller number of DOFs.

	$p = 1,\ C^0$		$p = 2,\ C^1$		$p = 3,\ C^2$	
Mesh size	DOFs	Velocity	DOFs	Velocity	DOFs	Velocity
0.01563	2193	0.0169	2340	0.0154	2489	0.0152
0.00781	8481	0.0164	8772	0.0158	9065	0.0156
0.00521	18865	0.0160	19300	0.0156	19737	0.0156
0.00391	33345	0.0159	33924	0.0156	34505	0.0158
0.00313	51921	0.0157	52644	0.0158	-	-
0.00272	68541	0.0158	-	-	-	-

Table 4.1: Details of the simulations

In Table (4.1) we report the details of the simulations run in order to obtain the graph shown in Figure (4.3). We first observe that the dimension of the domain Ω and as well as the time t are dimensionless quantities and thus the conduction velocity obtained is also dimensionless. The obtained results show that the conduction velocities simulated using $p = 1$, C^0 basis functions overestimate the true velocity of the traveling pulse. In fact by means of the $h-$refinement, Figure (4.3(a)) we observe that the value of numerical velocity of propagation decreases when the mesh size is decreased. For the two finer meshes ($h = 0.00313$ and $h = 0.00272$) the obtained velocities values are 0.0157 and 0.0158 respectively. On the other hand, the value of the conduction velocities simulated with high order continuous basis functions, have a different behavior under the $h-$refinement. For example, for $p = 2$, C^1 basis functions the value of the velocity increases when refining from $h = 0.015625$ to $h = 0.00781$. Then, for the successive refinements the value of the conduction velocity oscillates between 0.0156 and 0.0158. The same behavior is observed for the conduction velocities obtained with

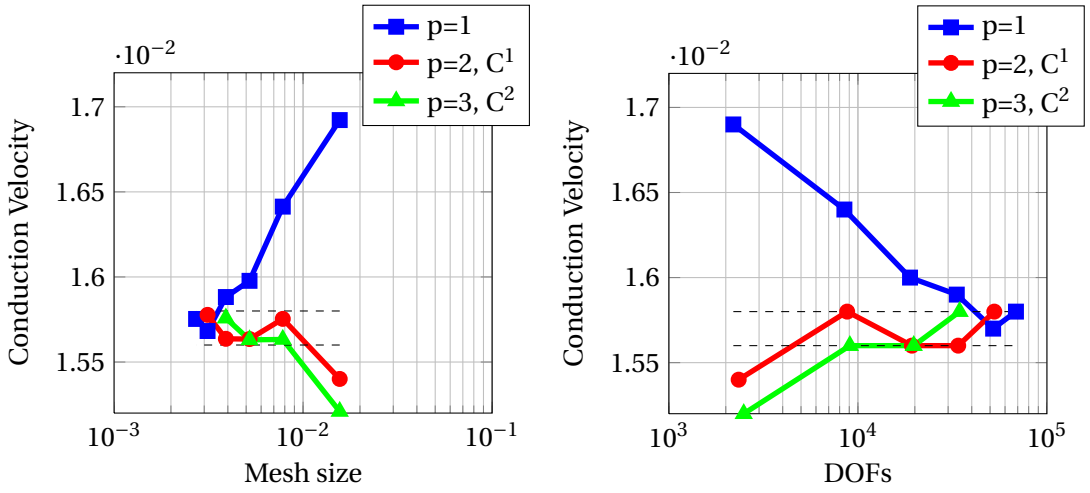


Figure 4.3: Convergence of the conduction velocity for different NURBS basis functions. (■) $p = 1,\ C^0$, (●) $p = 2,\ C^1$, (▲) $p = 3,\ C^2$, with respect to the mesh size (a), and to the total number of DOFs (b).

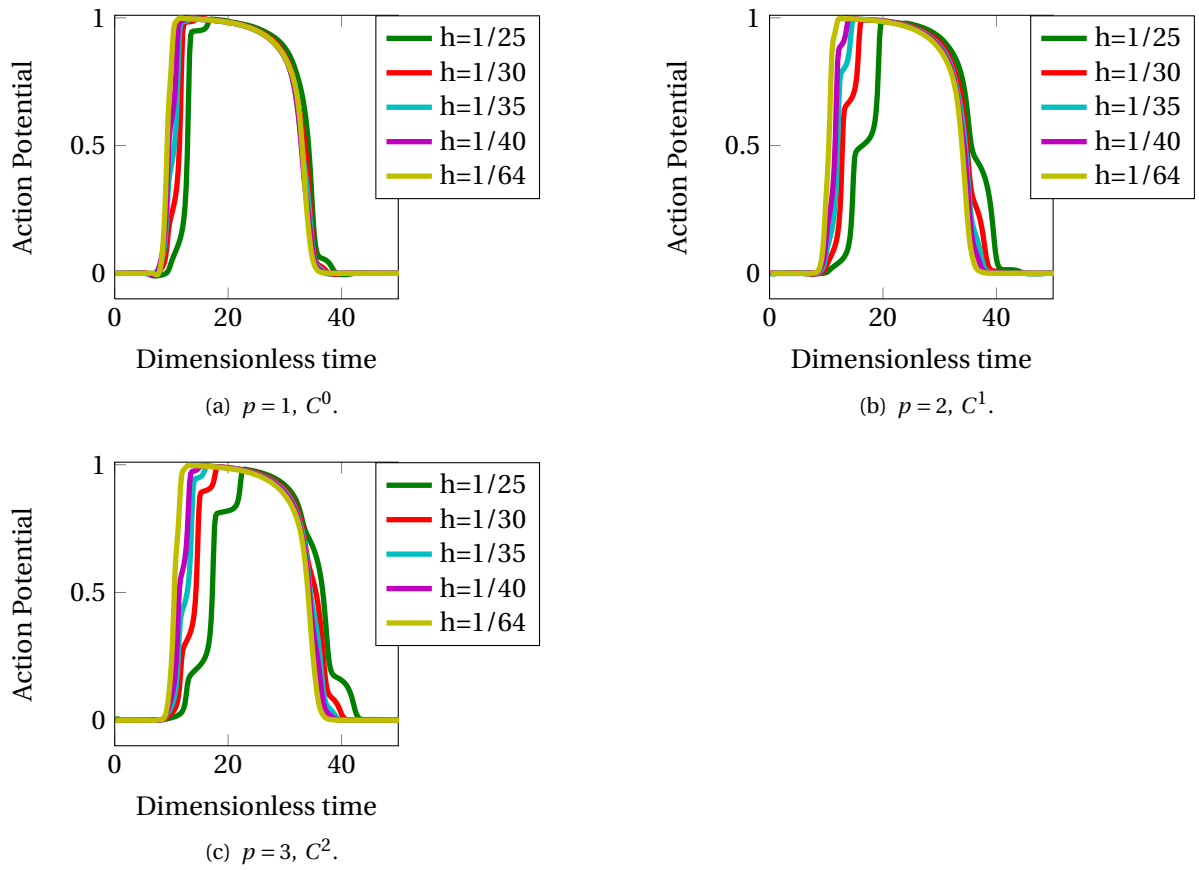
$p = 3,\ C^2$. The conduction velocity obtained with the simulation run on the coarser mesh considered, provides a values of 0.00152, then for the subsequent refinements the values of the conduction velocities obtained, are 0.0156 or 0.0158. Practically speaking, we were unable to run simulations on extremely fine meshes. For this reason, in this analysis we do not know the exact velocity of the propagating pulse, but we can infer from the obtained conduction velocities, shown in Table (4.1), that the expected true value of the velocity lies in the interval $I_{v_{ex}} = [0.0156, 0.0158]$, which are the values obtained for the finer meshes considered. By considering the total number of DOFs, Figure (4.3(b)), we remark that by using high order continuous basis functions, the value of the conduction velocity is in $I_{v_{ex}}$ for a smaller total number of degrees of freedom.

We observe that the oscillations of the approximated conduction velocity remarked in Figures (4.3(a)-4.3(b)) may could be related to the temporal discretization. However, there are no concrete mathematical explanations for that behavior.

In conclusion, in this section we show that the convergence of the conduction velocity is related to the global continuity of the basis function, and that the advantage of increasing the global continuity of the basis functions is that the total number of degrees of freedom needed in order to approximate the proper velocity of propagation of the action potential, is "small". Thanks to this analysis we understand that the $k-$refinement provides a new alternative methodology for the approximation of the conduction velocity. In fact, by increasing the global continuity of the basis functions the approximation of the conduction velocity depends less on the mesh size. This fact becomes clear by comparing the data in Table (4.1). For $p = 1,\ C^0$ is clear that the conduction velocities obtained strongly depends on the mesh sizes considered, while, for $p = 2,\ C^1$ and $p = 3,\ C^2$ basis functions the mesh sizes influence less the approximation of the conduction velocities of the action potential.

4.2 Approximation of the action potential

After showing the advantages of the C^1- and C^2-continuous NURBS basis functions for the approximation of the conduction velocity of the propagating pulse, we want to see the effect of high order continuous NURBS basis functions for the approximation of the action potential. For this we tested different basis functions for relatively coarse meshes in order to better understand the relation between the regularity of the basis functions and the approximation of the action potential.



(a) $p = 1$, C^0.

(b) $p = 2$, C^1.

(c) $p = 3$, C^2.

Figure 4.4: Action potential shape for different mesh size, polynomial order, and global continuity of the basis functions. The real time, in milliseconds, can be retrieved using Eq. (1.8).

For all the simulations the action potential has been obtained by selecting a single point of the mesh, and by extracting the value on that point of the approximated solution, over the time. In Figure (4.4) we report the obtained results.

We remark that for the mesh size $h = 1/64$ the action potentials approximated by the three considered basis functions have qualitatively the same shapes. In the following discussion, we will refer to shapes of the action potential obtained with the mesh size $h = 1/64$, as the *reference* action potential shape. An interesting fact is that using C^1- and C^2-continuous basis function, the action potential shape converges, under $k-$refinement, slower to the reference one compared to the action potential approximated using $p = 1$, C^0. In fact the duration of the fast depolarization phase of the solution approximated using higher order continuous basis functions, is longer compared to the duration of the same phase for the reference action potential. In fact, in Figures (4.4(b)-4.4(c)), it can be remarked that for the first 20-25 units of dimensionless time, the action potential value pass from 0 to 1 in a stepwise manner. Moreover, under $k-$refinement the (dimensionless) time needed to the action potential to accomplish the fast depolarization phase, decreases and the action potentials shape of the fast depolarization become almost vertical. Moreover, the higher is the continuity of the basis functions, the bigger is the number of steps between 0 and 1.

The accuracy of the approximated solution of a traveling pulse described by the system (4.1-4.2), involved the approximation of the conduction velocity and the approximation of the shape of the action potential duration. By combining the finding of this section and the conclusions of the previous one (section 4.1), we conclude that increasing the global continuity of the basis functions is still efficient.

The system of equations (4.1-4.2) is able to describe the fact that if the time between two stimuli

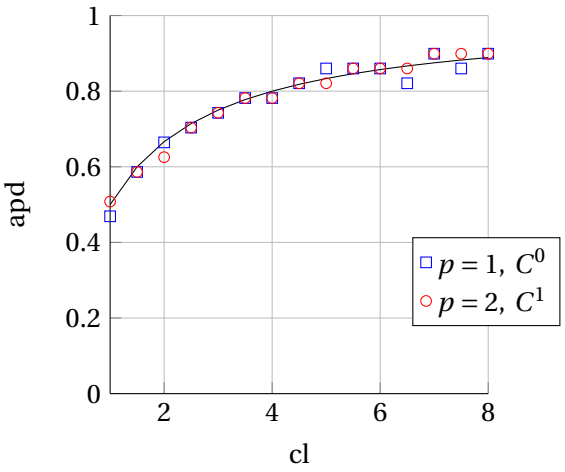
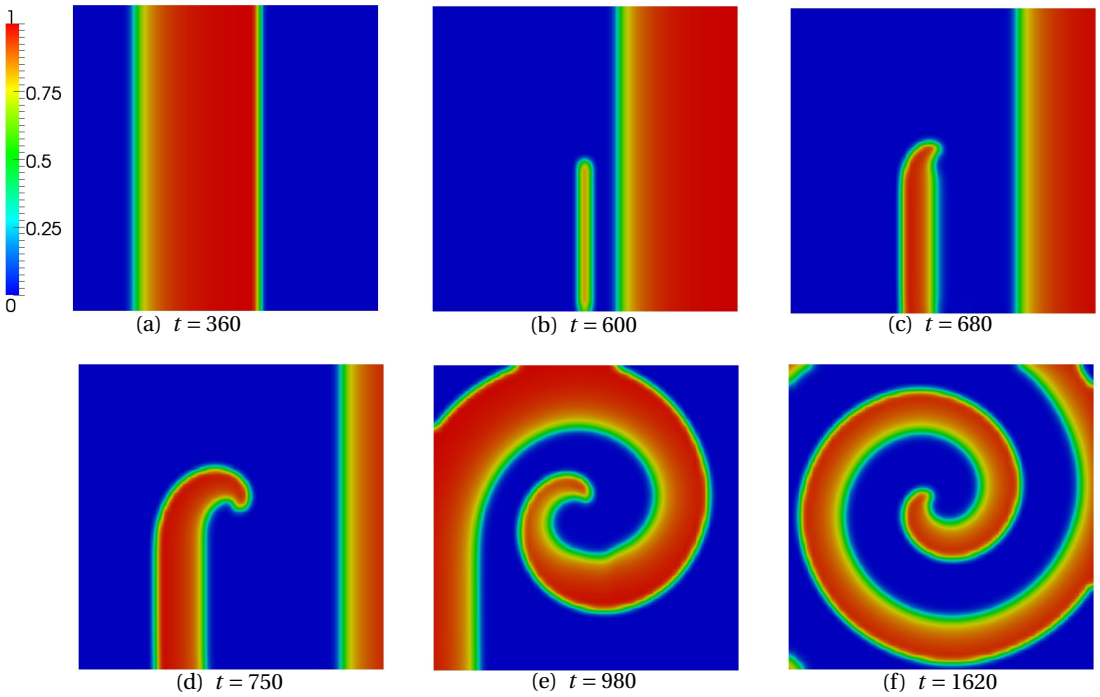


Figure 4.5: Comparison between $p=1,\ C^0$ basis function (□) and $p=2,\ C^1$ basis functions (○), in approximating the action potential duration (apd) for different cycles length (cl). In black solid line we plot the relation (1.9) for $b=1$. All details about the definitions of the apd and the cl can be found in section 1.2.6.1.

decreases, the action potential duration decreases, which means that when the heart is under effort the sinoatrial node increases the rhythm of the stimuli and the action potential duration decreases in order to be able to contract faster the cardiac muscles. In Figure (4.5) we show the results obtained by simulating the latter phenomena with $p=1,\ C^0-$ basis functions, and $p=2,\ C^1$ basis functions. The mesh size used was $h=0.01563$ and different cycles lengths (cl) have been considered. We observe that the action potential duration (apd) obtained with both basis functions,(□) and (○) in Figure (4.5), lay close to the theoretical value computed with the formula (1.9), and there is not remarkably differences between the two basis functions.



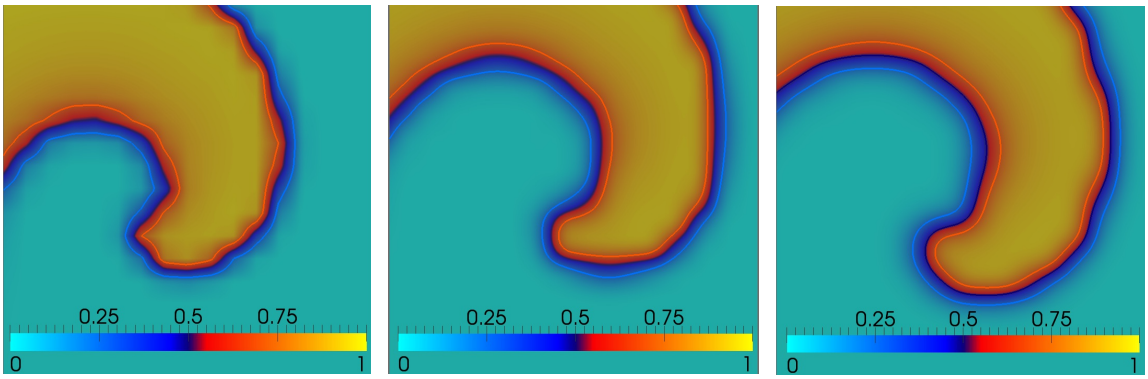
(a) $t=360$ (b) $t=600$ (c) $t=680$

(d) $t=750$ (e) $t=980$ (f) $t=1620$

Figure 4.6: Example of initiation of Spiral wave simulated with $p=1,\ C^0-$continuous basis functions, and mesh size $h=0.015625$.

4.3 Approximation of spiral waves

The Aliev-Panfilov model [1] can describe other dynamics different from the simple propagating pulse. One example of such dynamics is the propagation of the potential in a rotating way. In fact some pathological diseases of the heart, such as cardiac arrhythmia, can be related to the presence of such action potential spiral that leads to an irregular contraction of the cardiac muscles; for details see [31, 32, 33]. In the simulation, the initialization of the vortex is done by applying a stimulus in the wake of a propagating pulse. In Figure (4.6) we show a simulated rotating wave where the traveling pulse goes from the left boundary of the square, to the right. When the second stimulus is applied, the action potential can propagate only on the left and upper part of the domain, because the right part of the domain still be in a refractory phase, which means that in this phase the cells can not be excited. After the refractory phase is finished the action potential is free to propagate in all the domain. Numerically speaking, these dynamics require the approximation of a moving curved front, thus we are interested in comparing the simulations of these dynamics for different polynomials orders and continuities of the basis. For all simulations the unitary square has been taken as computational domain.



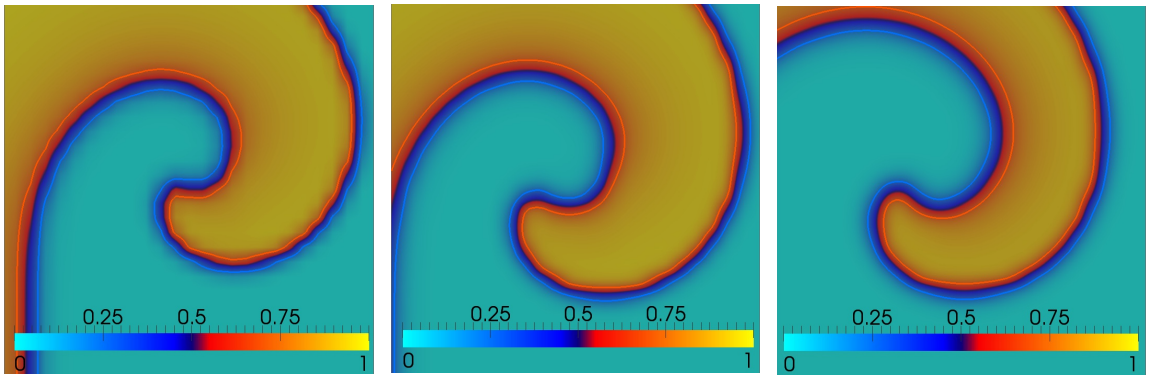
(a) $p = 1$, C^0. (b) $p = 2$, C^1. (c) $p = 3$, C^2.

Figure 4.7: Smoothing effect of high order continuous basis functions for mesh size $h = 0.025$. The contour lines of the approximated solution at the values of 0.7 (—) and 0.25 (—) are shown.

The first analysis we do is on the quality of the approximation of the curved front of the wave. For the sake of simplicity we focus our attention on the tip of the spiral. The following analysis is based on a qualitative comparison between the obtained result. In particular the term *quality* of the front will be used to classify the smoothness of the latter; the smoother the front will be, the higher the quality of the spiral wave will be. The latter definition of quality makes sense because we are considering an isotropic diffusion coefficient. Finally, this qualitative analysis will be used for a quantitative analysis based on the total number of degrees of freedom used in order to achieved a certain quality of the spiral wave.

We considered two mesh sizes, $h = 0.025$ and $h = 0.015625$, and initialized a spiral always in the same region for all the simulations considered. The different tips obtained with the computations are shown in Figures (4.7-4.8), where we compared simulation run with $p = 1$, C^0, $p = 2$, C^1 and $p = 3$, C^2 NURBS basis functions. In Figure (4.7) we observe that the quality of the front of the solution obtained with $p = 1$ can be classified as extremely low; this classification is based on the fact that along the front of the tips, the solution presents some peaks which made the front not smooth.

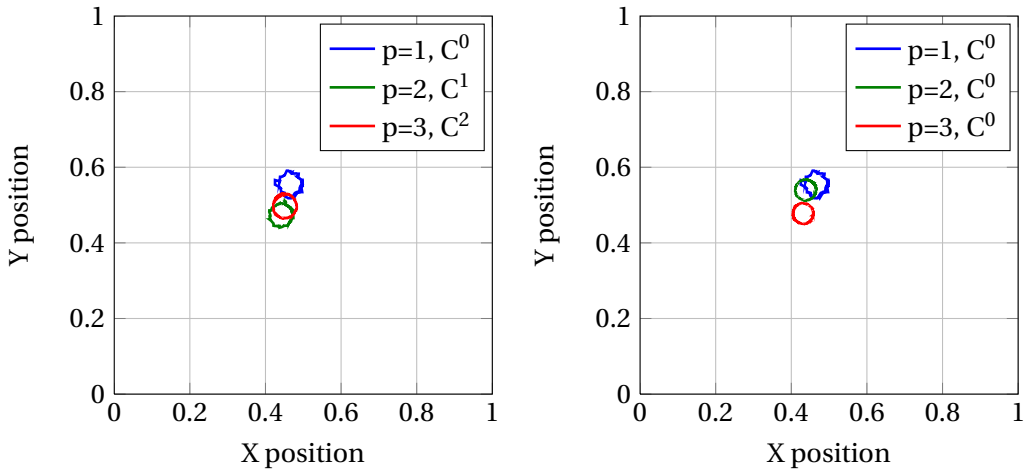
By increasing the continuity of the basis functions we remark that the quality increases and the peaks are smoothed out. By increasing the mesh size, Figure (4.8), we notice that the tip of the vortex obtained using $p = 1$ still have a peak while the tip of the vortexes simulated with C^1- and C^2-continuous basis functions are smoother and qualitatively much better. Thus the quality of the front for vortex simulated with higher continuous basis functions is better compared to the fronts



(a) $p = 1$, C^0. (b) $p = 2$, C^1. (c) $p = 3$, C^2.

Figure 4.8: Smoothing effect of high order continuous basis functions for mesh size $h = 0.015625$. The contour lines of the approximated solution at the values of 0.7 (—) and 0.25 (—) are shown.

simulated with $p = 1$, C^0-continuous basis functions. Moreover, increasing the global continuity of the basis function does not increase excessively the total number of degrees of freedom, in fact for the mesh size $h = 1/64$, the DOFs for $p = 1$ are 4425, while for $p = 3$, C^2, are 4489. Thus, by fixing the mesh size, the best quality of the front can be achieved by increasing the global continuity of the basis functions without increasing the system to solve.



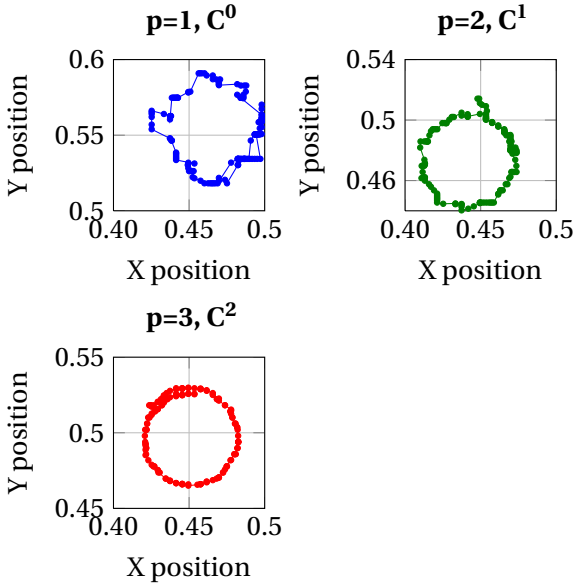
(a) Comparision between different continuities of the basis functions. (b) Comparision between different polynomials orders of the basis fucntions, globally C^0.

Figure 4.9: Positions, in Ω, of the trajectories of the tips of different spiral waves obtained with the mesh size $h = 0.015625$.

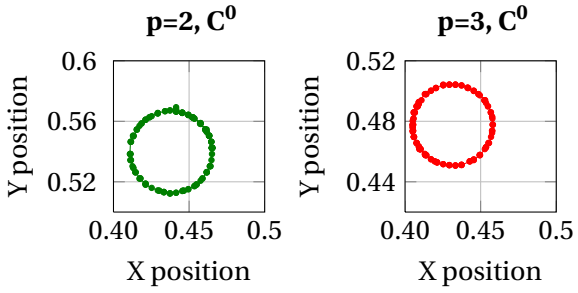
When simulating a 2D spiral on a coarse mesh, such as $h = 0.025$, with first order polynomial basis functions, the trajectory of the tip of the spiral wave is influenced by the orientation of the elements of the mesh. More precisely, the tip follows the elements of the mesh, which in IGA are squared for uniform meshes. This fact motivates the second analysis we have made on the numerical simulation of spiral wave. As for the previous qualitative analysis, we initialized all the spiral waves by applying the same stimulus in the same region, for the same duration, and of the same intensity.We want to point out that we compare different simulations run with the same mesh size $h = 0.015625$. In section 4.1 we learned that the velocity of the traveling pulse depends on the mesh size and on the basis functions used. This implies that the time (dimensionless) when the second stimulus is applied depends also on the mesh size and on the basis function used. Moreover, different times of initializa-

tion of the vortex leads to different trajectories of the tip of the latter. For example, in [33], different vortexes have been initialized with different times of application of the second stimulus, then the trajectories of the tips have been tracked. It has been shown that the behavior of the tips' trajectories strongly depend on the time when the spiral wave is initialized. They found out six different types of trajectories followed by the tip of the rotating wave: circular, epicycloidal, cycloidal, hypocycloidal, hypermeandering, and linear. We observe that in [33] they considered the 3-variable Fenton-Karma model [32], but the latter behavior of the spiral wave tip are valid for all other models describing the electrical activity of the cardiac tissue.

In our case we study the trajectory of the tip in the case of circular behavior, i.e., the tip rotates around a fixed center. In Figure (4.9(a)) we show the positions of the trajectories of the tips simulated using $p = 1$, C^0 (in blues), $p = 2$, C^1 (in green), and $p = 3$, C^2 (in red), with respect to the whole domain. We remark that the trajectories centers do not coincide. In Figure (4.10) we show the details of the trajectories of the tips. The trajectory obtained with $p = 1$ is far from being circular, in fact it presents some straight line because the tip follows, piecewise, the elements of the mesh. By increasing the continuity of the basis functions the trajectory become more circular, and for C^2-continuous basis functions the circular behavior of trajectory of the tip is achieved Figure (4.10(a)).



(a) Trajectories obtained by increasing the global continuity of the NURBS basis functions.



(b) Trajectories obtained by pure $p-$refinement

Figure 4.10: Detail of the circular movement of different spiral waves obtained with the mesh size $h = 0.015625$.

We also investigate the effect of pure $p-$refinement on the approximation of the movement of the spiral wave's tip. The result are reported in Figures (4.9(b)-4.10(b)) were we compare solution

obtained by using $p = 1,\ C^0$, $p = 2,\ C^0$, and $p = 3,\ C^0$ basis functions. We remark that by only increasing the polynomial order the circular movement of the spiral wave's tip is already achieved by using quadratic polynomial basis functions globally C^0-continuous. Thus, by refining one time by elevating the polynomial order, the circular behavior of the tip is achieved. For this reason we say that, under pure $p-$refinement, the convergence of the approximated trajectory of the tip to the real one is fast. But this gain in approximating speed of the convergence leads to a large total number of degrees of freedom of the system. In fact when using $p = 2,\ C^0$, the total number of DOFs are 16641, while for $p = 3,\ C^2$ the DOFs are only 4489. Thus, from a point of view of optimality between qualitative and quantitative aspects, the best choice of basis functions are the one with $p = 3$, globally C^2-continuous.

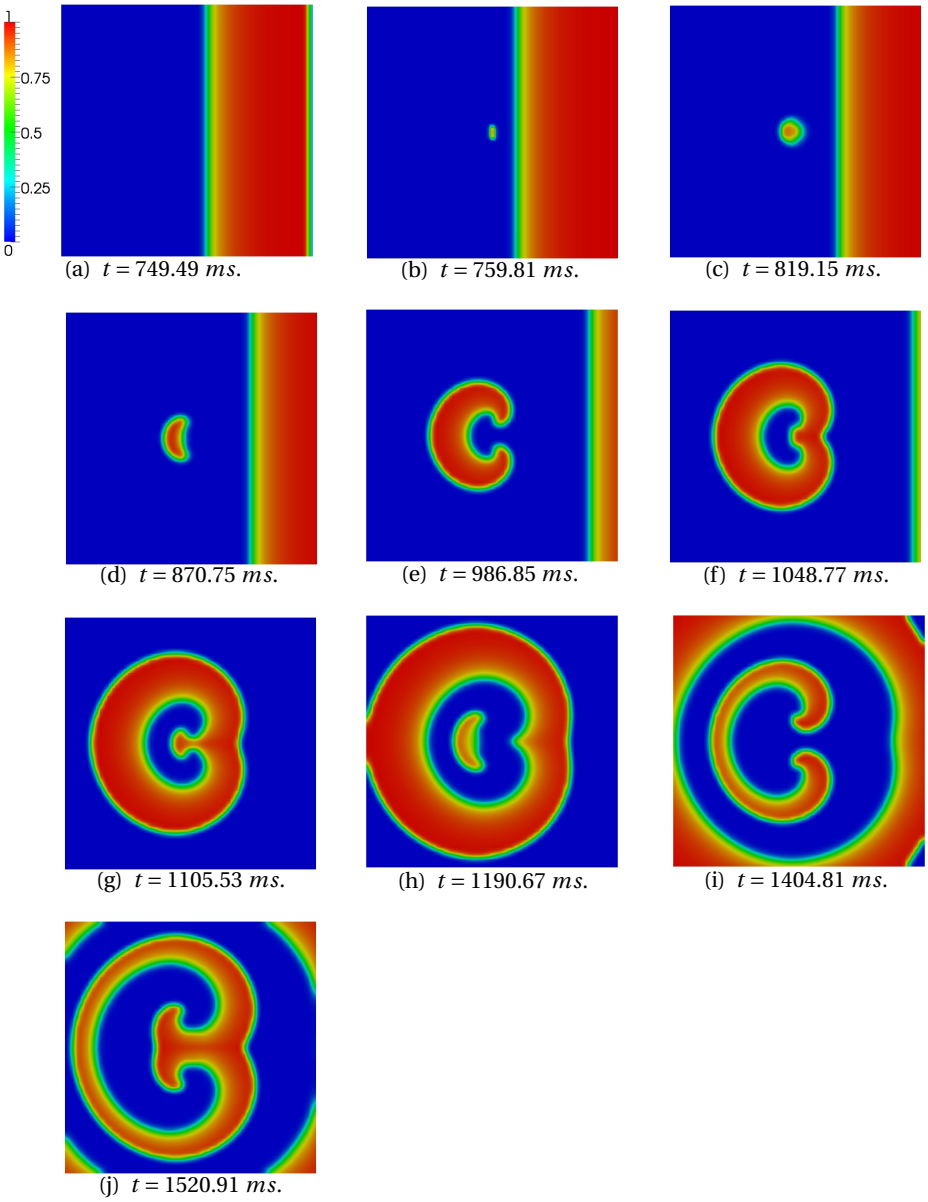
4.4 Simulation of spiral waves induced by “pacemaker” cells

The spiral wave is one type of dynamics that can be initialized by applying an extra stimulus in the wake of the propagating pulse. In this section we show a second type spiral wave simulated with the Aliev-Panfilov model [1]. In Figure (4.11) we show how this other spiral wave is induced. From a physiological point of view this kind of dynamics can occur in a human heart, if a small group of cardiac cells manages to stimulate themselves, i.e., they become “pacemaker” cells.

As for the spiral wave, the initialization of this dynamics is done by applying a extra stimulus on the wake of a propagating pulse. Just after the second stimulus is applied, the action potential can not propagate itself in the same direction of the propagation pulse, i.e., can not propagate itself to the right of domain. This is due by the fact that the cardiac tissue has a resting phase when its cells can not be excited. After the resting phase is over the action potential is allowed to propagate itself in the whole domain, and thus the dynamics is initialized.

The latter dynamics are more complicated then the spiral wave. In fact it involves two connected circulating tips (Figures (4.11(e)-4.11(i)) that periodically merge themselves in an unique annular ring like shape that freely propagates outwards in all directions, from its origin, Figures (4.11(h)-4.11(j)). During the merging process the two tips also come together and, moreover, the resulting shape detaches itself from the annular ring like shape; see Figures (4.11(f)-4.11(h)). The shape resulting from the merging of the two tips will then propagate itself in the way described above; see Figures (4.11(h)-4.11(j)).

From a numerical point of view the approximation of this dynamics is equivalent to the approximation of a spiral waves, in fact both dynamics leads to the approximation of a moving curved front. The analysis made on the quality of the front of spiral wave, can be applied also for this other dynamics. More precisely, the front can be smoothed by increasing the the global continuity of the basis functions, instead of elevating the polynomial order which will increase substantially the total number of degrees of freedom of the system.



(a) $t = 749.49\ ms.$ (b) $t = 759.81\ ms.$ (c) $t = 819.15\ ms.$

(d) $t = 870.75\ ms.$ (e) $t = 986.85\ ms.$ (f) $t = 1048.77\ ms.$

(g) $t = 1105.53\ ms.$ (h) $t = 1190.67\ ms.$ (i) $t = 1404.81\ ms.$

(j) $t = 1520.91\ ms.$

Figure 4.11: Dynamics reproducing the effect of cells becoming “pacemaker” cells and activate them self in the wake of a propagatin action potential pulse. Simulation obtained with $p = 1$, C^0 basis functions and mesh size $h = 0.015625$, at different time.

Chapter 5

Realistic Application: the Human Left Atrium

The aim of this section is to solve the Mitchell-Schaeffer model [2] presented in Chapter 1 by means of IGA, on a surface $\Omega \subset \mathbb{R}^3$, where Ω will be represented by means of NURBS basis functions and will be dimensionally and geometrically analogous to the human left atrium (LA). We recall that the two-variable model we want to solve reads as:

$$\begin{cases} \dfrac{\partial \overline{V}}{\partial t} = \nabla(\mathbf{D}\cdot\nabla\overline{V}) + \dfrac{h}{\tau_{in}}\overline{V}^2(1-\overline{V}) - \dfrac{\overline{V}}{\tau_{out}} + I_{app}, \text{ in } \Omega\times I, \\ \dfrac{dh}{dt} = \begin{cases} \dfrac{1-h}{\tau_{open}}, \text{ if } \overline{V} < V_{gate} \\ \dfrac{-h}{\tau_{close}}, \text{ if } \overline{V} > V_{gate} \end{cases}, \text{ in } \Omega\times I, \end{cases} \tag{5.1}$$

together with suitable initial and boundary conditions. In Eq. (5.1), $\mathbf{D}$ is the anisotropic conductivity tensor and I_{app} is the applied current. The values of the other parameters are reported in Table (1.2). The variables $\overline{V}$ and h are dimensionless quantities varying between 0 and 1.

In this section we will briefly discuss the methodology used in order to represent the human LA by means of NURBS and we discuss how we define the anisotropy of the LA tissue solving an auxiliary problem. Finally in the last part of this Chapter we present some numerical result obtained, with the focus being on the variation of the conduction velocity with respect to the total number of degrees of freedom of the discretized system. We compare simulations results obtained with different total numbers of degrees of freedom; see Table (5.1).

Ref. Level	DOFs	Time for a interation (s)
1 ($\mathcal{T}_{h,1}$)	10600	1.69
2 ($\mathcal{T}_{h,2}$)	22800	4.81
3 ($\mathcal{T}_{h,3}$)	39608	6.25
4 ($\mathcal{T}_{h,4}$)	61024	9.94

Table 5.1: Details of the simulation run. The time for a interation is the average over ten iterations and consider the assembly time plus the solution time.

5.1 Geometrical representation of the human left atrium by means of NURBS

The principal function of the left atrium of the heart is to pump the oxygenated blood into the left ventricle. Four pulmonary veins connect the lungs and the left atrium, the connection between

the left atrium and the left ventricle is controlled by the so called mitral valve. Thus, the left atrium have five holes.

In this section we want to briefly explain the major steps performed in order to represent the human left atrium by means of NURBS. The first step is to consider only the external surface of the left atrium. This simplification can be made due to the fact that the thickness of the atria wall is small and thus, transmural activation differences can be assumed to negligible. In Table (5.2) we report the sizes, in centimeters, of the anatomical features of the human LA taken into account, the data have been taken from [34].

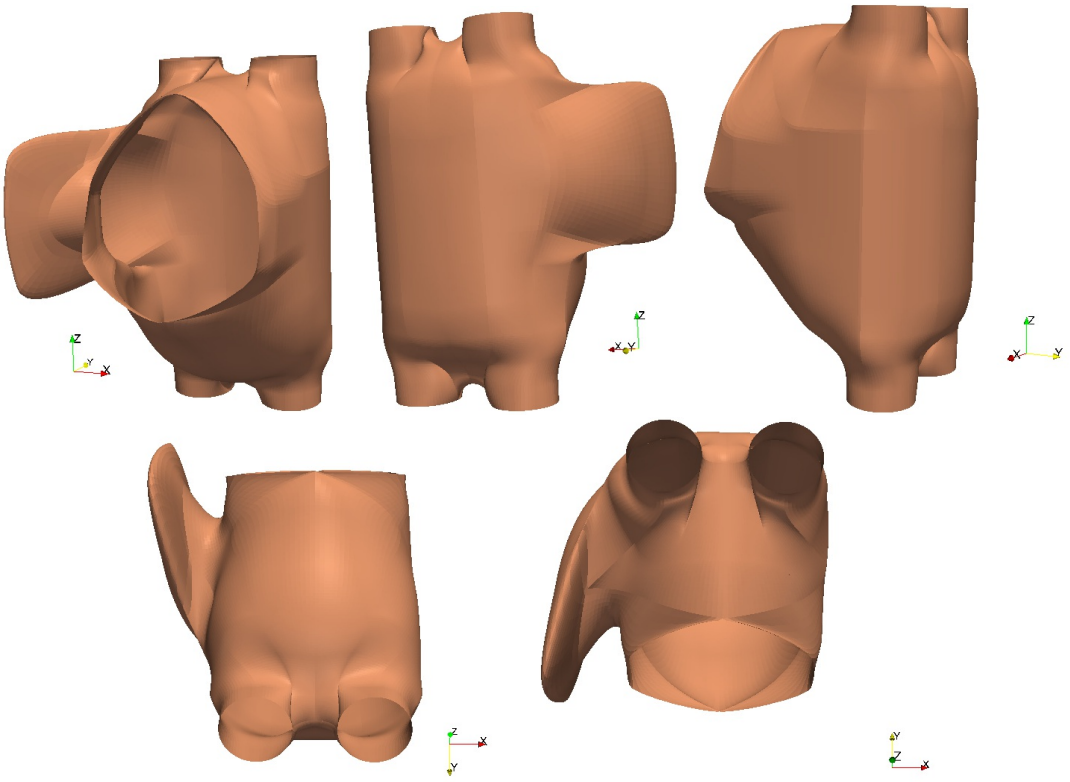


Figure 5.1: The computational domain represented wiht NURBS dimensionally and geometrically analogous to the human left atrium seen form different points of view.

The starting point in order to represent the LA by means of NURBS is the cylinder; see Figure (2.2(b)). For the sake of simplicity we will consider the parametric direction 1 as the one defining the circle, and the parametric direction 2 as the direction of the extrusion defining the height of the cylinder. The next step is to define the four pulmonary veins by inserting suitable knots in the parametric direction 1 in order to obtain enough control points to define two distinct circles at the base and at the top of the cylinder. We recall that in order to obtain an exact circle the knot vector, and consequently the basis, must be as shown in Figure (2.1(b)). Once the pulmonary veins are defined an extrusion of the latter is done. A second $h-$refinement then is done but in the parametric direction 2 in order to obtain the mitral valve. We observe that the mitral valve is not been approximated by a circle but by an oval like shape. The last major is the representation of the appendage as an extrusion of the surface. In Figure (5.1) we show the resulting representation of the human LA by means of NURBS.

As the NURBS used for the geometrical representation will be also used for the analysis part, is important to mention that for the geometrical representation we use $p = 2$, C^1-continuous NURBS, but in some lines of the mesh where the NURBS are only C^0-continuous. Thus, also for the analysis part the NURBS basis functions will not be globally C^1 in the whole computational domain. As last remark we want to point out that we denote Ω the computational domain and it representation by

LA anatomical feature	Reference values (cm)
Wall thickness	0.2 (mean)
Pulmonary veins diameter (inside)	1.1 (mean)
Mitral valve diameter (outside)	2.9 (mean)
Anterior-posterior extent	3.8 (mean)
Septal - lateral extent	3.9 (mean)
LA appendage	
Length	2.9 (mean)
Diameter (mid)	1.6 (mean)

Table 5.2: Sizes of the anatomical features of the human left atrium. Data have been taken from [34]

means of NURBS is exact. The meshes related to the considered refinements of Ω, will be denoted by $\mathcal{T}_{h,i}$, where i correspond to the level of refinement done; in Table (5.1) we report the details of the four levels of refinements that have been considered.

5.2 Laplace-Beltrami problem for the fiber direction

The cardiac tissue is strongly anisotropic due to the orientation of the fibers which the heart wall is made of. Many techniques have been used in order to reproduce a realistic conductivity of the cardiac tissue. For example in [35] the authors divided the left atrium into small regions, then using on specific studies on the anatomy of the atria, see for example [36], they defined the conduction tensor values in those regions. The same procedure has been adopted in [37].

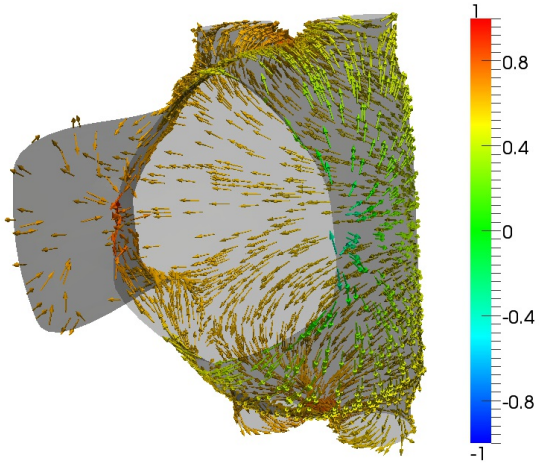


Figure 5.2: Solution of problem (5.2). The Glyph represents the normalized gradient **U** and the color map is the value of the approximated solution u_h.

We decided to adopt a different methodology with respect to [35] and [37]. Let us consider the mesh $\mathcal{T}_{h,i}$, $i = 1,\dots,4$, we first solve a Laplace-Beltrami problem on Ω (Figure (5.1)), with the mesh $\mathcal{T}_{h,i}$, where some suitable boundary conditions have been imposed. In details we solve the following problem:

$$\begin{cases} -\Delta u & = 0, \text{ in } \Omega, \\ \nabla u \cdot \mathbf{n} & = 0, \text{ on } \partial\Omega \setminus (\Gamma_1 \cup \Gamma_2), \\ u & = 1, \text{ on } \Gamma_1, \\ u & = -1, \text{ on } \Gamma_2, \end{cases} \tag{5.2}$$

where $\partial\Omega$ is the boundary of Ω, and $\Gamma_{1,2}$ are two regions of the boundary $\partial\Omega$ on the mitral valve. Moreover, we imposed the solution to be equal to 1 in two points in Ω, exactly between both couple of pulmonary veins. The boundary conditions and the imposed values in the domain allows to get the desired orientation of the gradient of the solution. In Figure (5.2) we show the normalized gradient of the solution. In Figure (5.3) we show in details the glyph of the normalized gradient in different regions of the LA mesh. Firstly we observe that our fibers field is similar to the fibers field used in [35, 37], and secondly we observe that the glyph of the normalized gradient between both couples of pulmonary veins is the consequence of prescribing the value of the solution in two internal points of the domain Ω.

Let now u_h be the approximated solution obtained by solving problem (5.2) with the mesh $\mathscr{T}_{h,i}$ by means of the Galerkin method based on NURBS basis functions. We define the normalized gradient of u_h locally for each element $K \in \mathscr{T}_{h,i}$, as:

$$\mathbf{U}(\mathbf{x}_{gp}) = \frac{\nabla u_h(\mathbf{x}_{gp})}{||\nabla u_h(\mathbf{x}_{gp})||}, \ \forall \mathbf{x}_{gp} \in K, \tag{5.3}$$

where $\mathbf{x}_{gp}$ are the quadrature Gauss points, $||\cdot||$ is the standard Euclidean norm in $\mathbb{R}^3$. Let us denote by $(\nabla u_h)_j$ the $j-$th component of the approximated solution's gradient. The evaluation of the normalized gradient (5.3) at the Gauss quadrature point $\mathbf{x}_{gp} \in K$ is done, component wise, as follows:

$$(\nabla u_h(\mathbf{x}_{gp}))_j = \sum_{A=1}^{n_{eq}} (\nabla R_A(\mathbf{x}_{gp}))_j d_A = \sum_{B=1}^{n_{K,func}} (\nabla R_B(\mathbf{x}_{gp}))_j d_B, \ \forall j = 1,2,3, \tag{5.4}$$

where $\{d_A\}_{A=1}^{n_eq}$ are the components of vector ∇u_h, $(\nabla R_A)_j$ is the jth component of the gradient of the NURBS basis function ∇R_A, n_{eq} is the total number of basis functions related to mesh $\mathscr{T}_{h,i}$, $n_{K,func}$ is the total number of NURBS basis functions which compact supports are in the element K. Thus, we can now define locally the components D_i of the anisotropic diffusion tensor as:

$$D_i(\mathbf{x}_{gp}) = \gamma D_{iso} + (1-\gamma) D_{iso} U_i(\mathbf{x}_{gp}), \ \forall \mathbf{x}_{gp} \in K, \tag{5.5}$$

where $\gamma \in [0,1]$ and $D_{iso} > 0$.

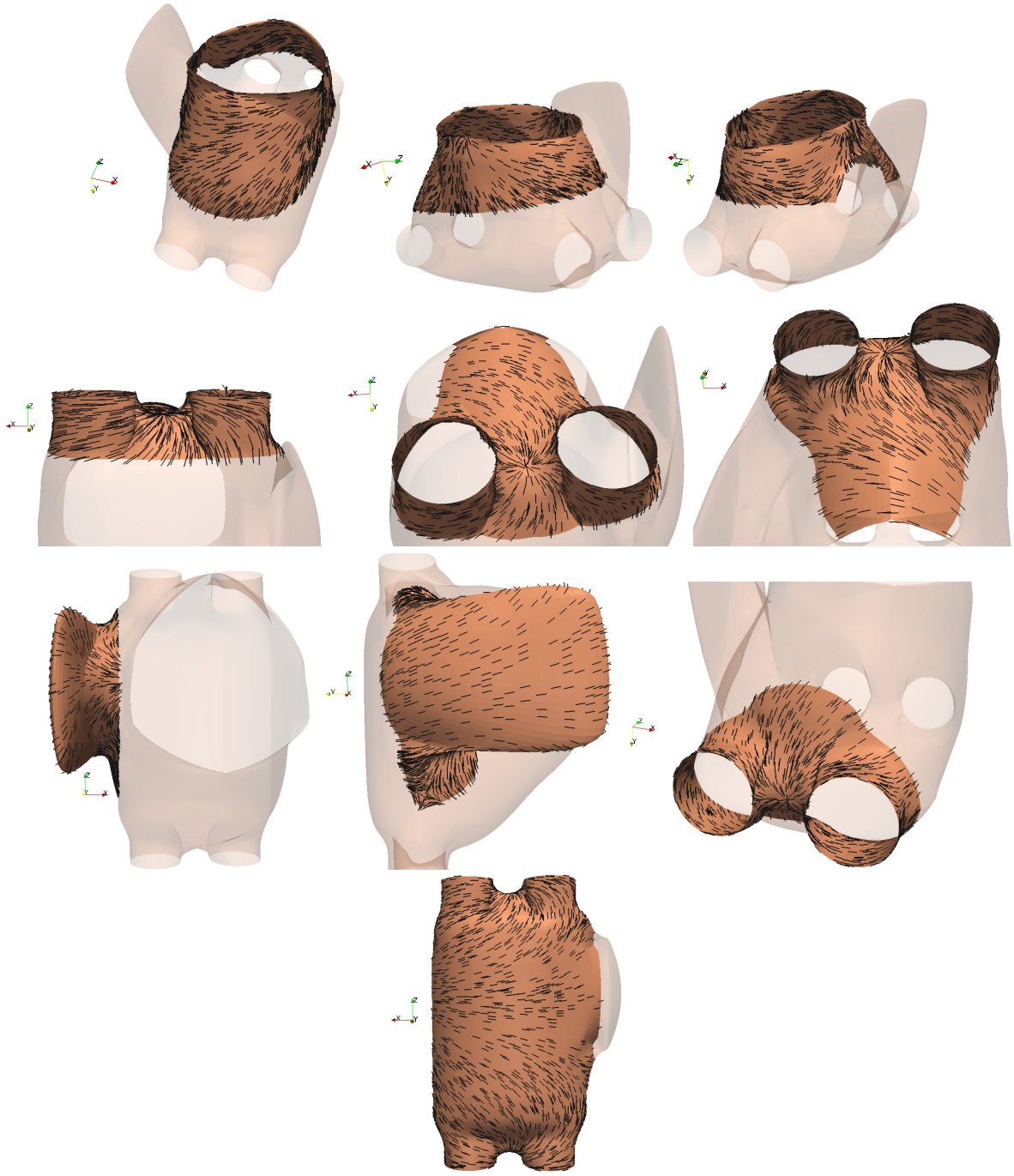
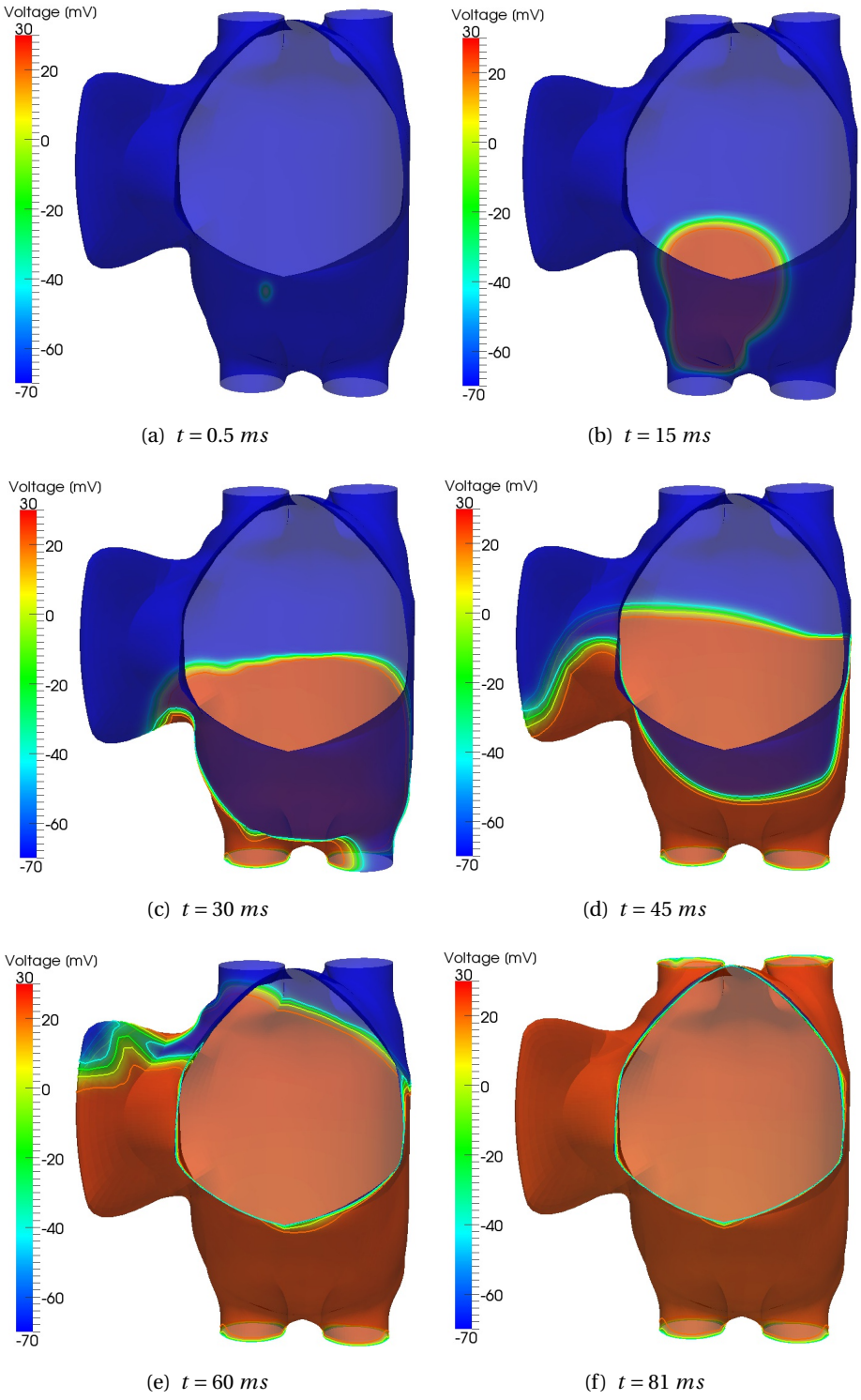


Figure 5.3: The normalized vector field defined by the gradient of the approximated solution of the Laplace-Beltrami problem (5.2).



(a) $t = 0.5\ ms$ (b) $t = 15\ ms$

(c) $t = 30\ ms$ (d) $t = 45\ ms$

(e) $t = 60\ ms$ (f) $t = 81\ ms$

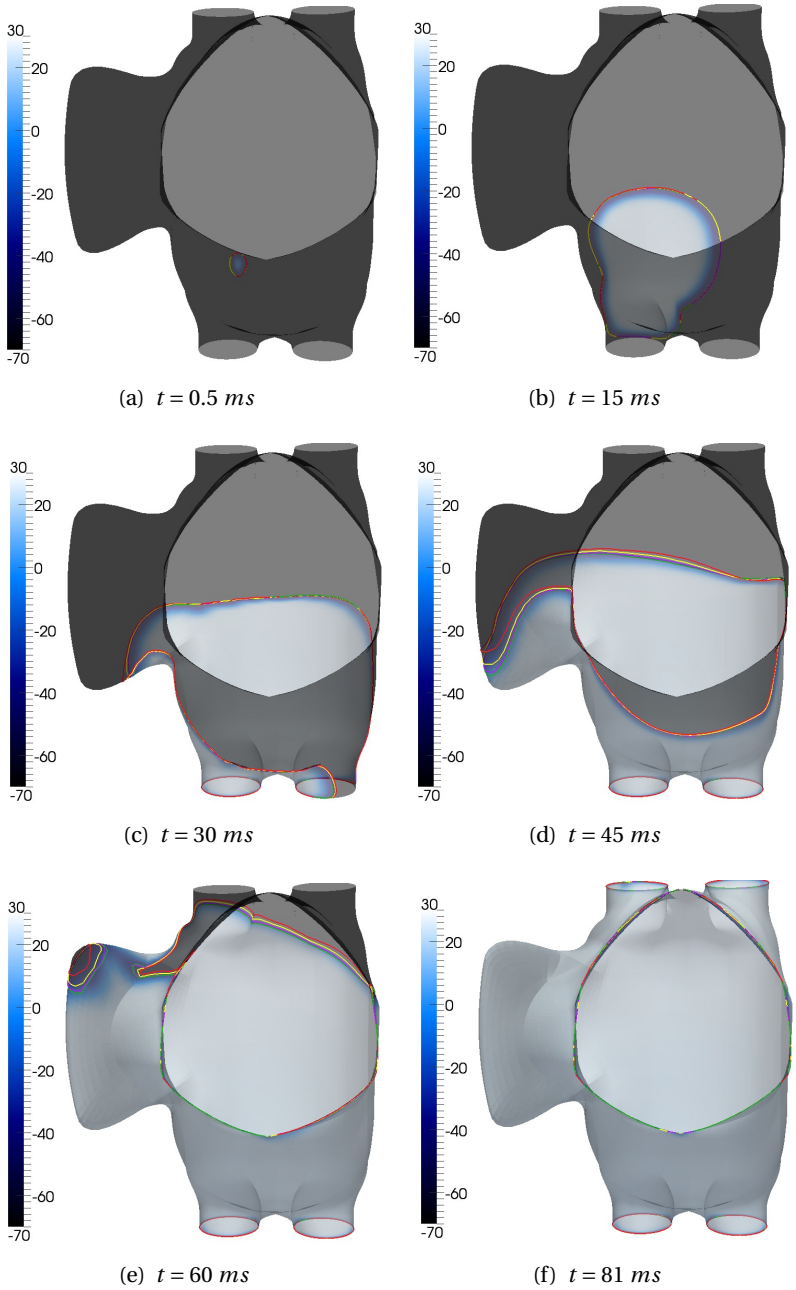
Figure 5.4: Solution obtained with quadratic NURBS basis functions at different time steps of the depolarization phase computed on $\mathcal{T}_{h,1}$

5.3 Numerical results

In order to numerically solve problem (5.1) we use the same methodology used in Chapter 4, i.e., a second order splitting method [28]. To solve the diffusion part we used the Galerkin method based on NURBS. We recall that the basis function used are of quadratic polynomial order and the

regularity is C^1 almost every where, but in some lines of the mesh where they are C^0. We considered four different meshes with different numbers of DOFs, which are reported in Table (5.1).

The two atria are electrical connected in four points: Bachmann's bundle, anterior septum, posterior septum, coronary sinus musculature; details can be found in [38]. Thus, the action potential can pass from the right atrium to the left one, through these four connections. In [38] there is some evidence that the time when the four connection are activated can be the cause of some pathology of the heart, such as cardiac arrhythmias, and that the four times may depend on other anatomical aspect of the heart. For the sake of simplicity, we chose to initialize the activation of the LA only in one point: the Bachmann's bundle (BB). In Figure (5.4) we report the simulation obtained on $\mathcal{T}_{h,1}$ of the activation of the left atrium.



(a) $t = 0.5\ ms$ (b) $t = 15\ ms$

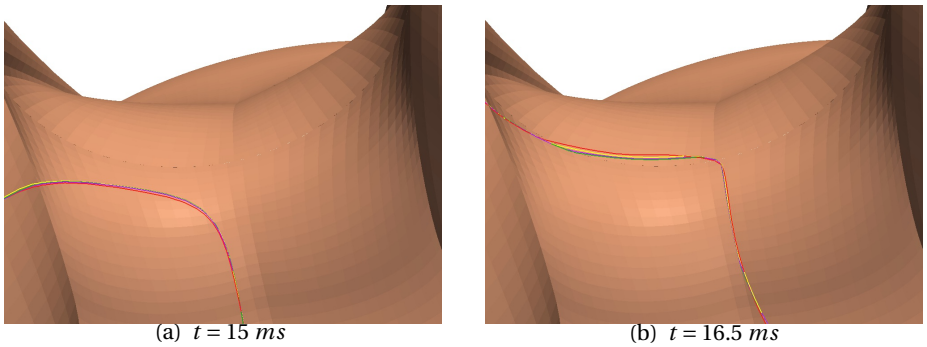
(c) $t = 30\ ms$ (d) $t = 45\ ms$

(e) $t = 60\ ms$ (f) $t = 81\ ms$

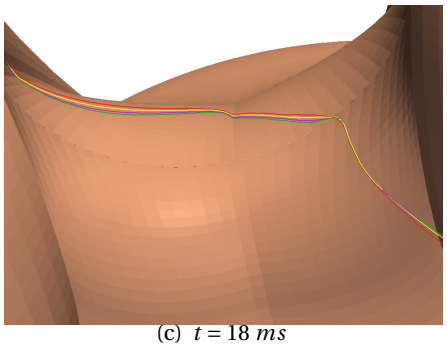
Figure 5.5: Comparison between solution obtained with Refinements $1-4$, see Table (5.1), at the value of $V_h = -60\ mV$. Lines colors: $\mathcal{T}_{h,1}$ (—), $\mathcal{T}_{h,2}$ (—), $\mathcal{T}_{h,3}$ (—), and $\mathcal{T}_{h,4}$ (—).

We analyze the conduction velocity of the different solutions obtained on the four considered

meshes $\mathcal{T}_{h,1},\ldots,\mathcal{T}_{h,4}$; details about DOFs are reported in Table (5.1). Thus, we compare the velocities of propagation of the front obtained on the considered meshes. In Figure (5.5) we report the contour lines of the approximated solution at the value of $-60\ mV$ obtained on the different meshes. We observe that the propagation of the action potential for the coarser mesh (—) is faster compared to the other simulations, in fact by following the front we observed that the velocity of conduction related to first refinement 1 (Table (5.1)) is faster than the other fronts. This is not in contradiction with the results obtained In Chapter 4, Figure (4.3), where we observed that with higher order continuous NURBS basis functions, the convergence of the conduction velocity is from below. The principal reason is that the NURBS basis functions used for the simulations are not globally C^1 but present some lines that are C^0. We show in Figure (5.6) the time steps of the simulation where the contour lines pass through a C^0 line between the lower pulmonary veins (with respect to Figure (5.7(a))). We observe that the velocities of propagation of the solution represented by the contour lines, change when passing through the C^0 line. In particular the solution obtain on the mesh $\mathcal{T}_{h,1}$ is faster while the slower is the one obtained on the finer mesh $\mathcal{T}_{h,4}$.



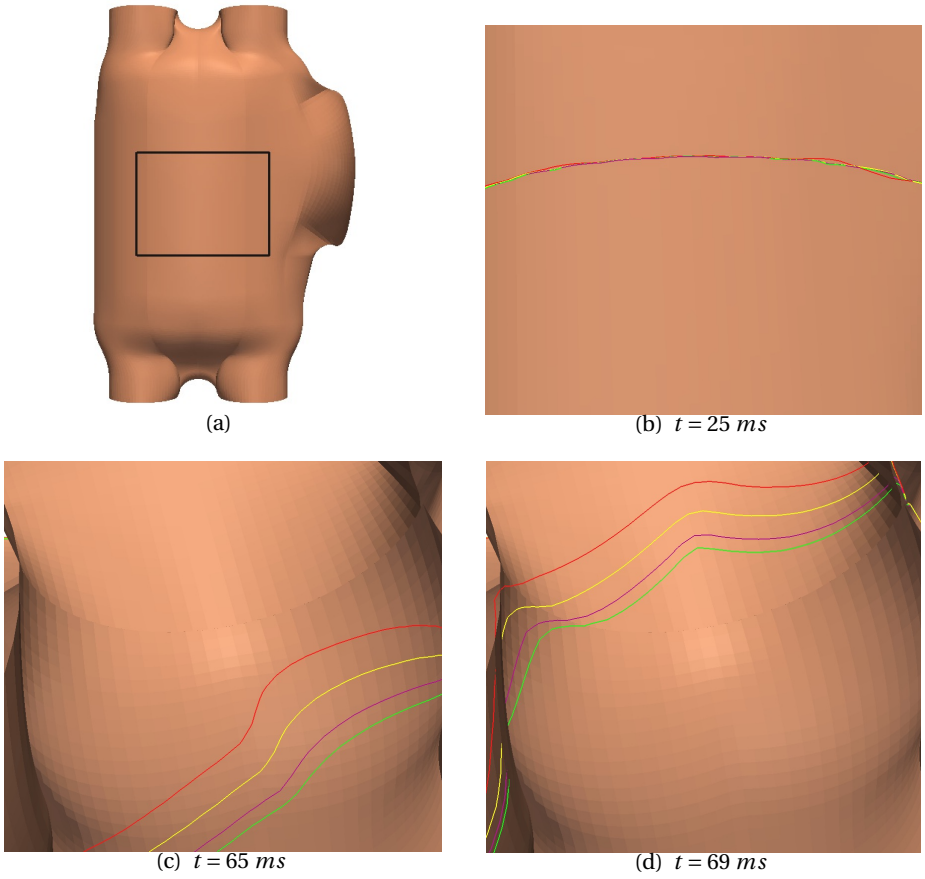
(a) $t = 15\ ms$ (b) $t = 16.5\ ms$



(c) $t = 18\ ms$

Figure 5.6: Contour lines of the approximated action potential ($V_h = -60\ mV$), corresponding to the refinements reported in Table (5.1), in the region between the two bottom pulmonary veins (with respect to (5.7(a)). Lines colors: $\mathcal{T}_{h,1}$ (—), $\mathcal{T}_{h,2}$ (—), $\mathcal{T}_{h,3}$ (—), and $\mathcal{T}_{h,4}$ (—).

In Figure (5.7(b)) we compare the different fronts of the solutions in the region highlighted by the black solid outline reported in (5.7(a)). We observe that the four contour lines are superposed, which implies that at time $t = 25\ ms$ the conduction velocities of the four solutions are the same in that region shown in Figure (5.7(a)). The last comparison of conduction velocity is made by looking at the fronts of the solutions passing between the upper pulmonary veins (with respect to Figure (5.7(a))). In Figures (5.7(c)-5.7(d)) we show the contour lines passing through the upper pulmonary veins, at the times $t = 65\ ms$ and $t = 69\ ms$. At the time $t = 65\ ms$ the contour lines are not yet passed through the C^0 line while at time $t = 69\ ms$ the four contour lines have passed the C^0 line. We remark that the conduction velocities are different for the four refinements at the two considered times of the simulation. The faster front is the one of the solution obtained on $\mathcal{T}_{h,1}$ while the slower one correspond to the solution obtained on $\mathcal{T}_{h,4}$.



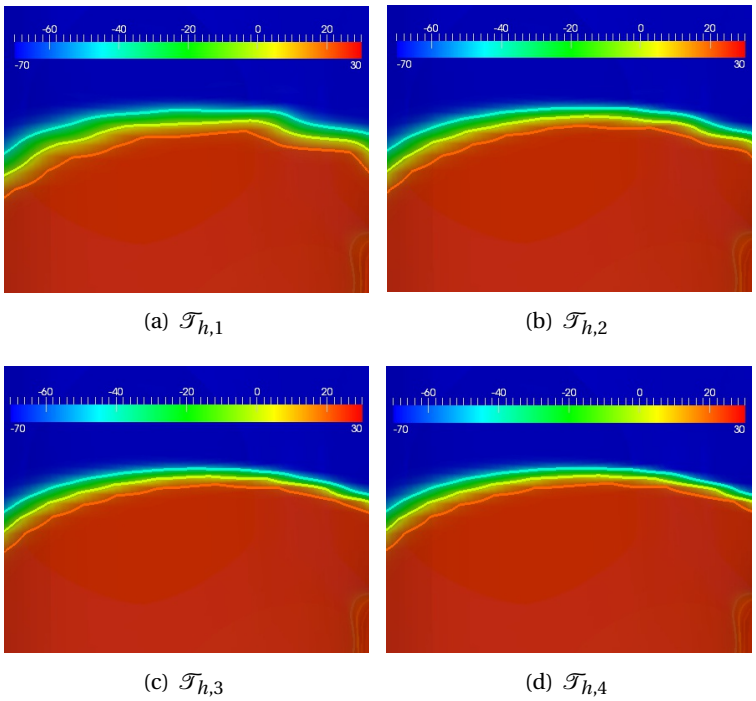
(a) (b) $t = 25\ ms$

(c) $t = 65\ ms$ (d) $t = 69\ ms$

Figure 5.7: Contour lines of the approximated action potential ($V_h = -60\ mV$), corresponding to the refinements reported in Table (5.1); (a) the solid black outline corresponds to the region of the mesh shown in (b); the region of the mesh reported in (c) and (d) corresponds to the region between the two upper pulmonary veins (with respect to (a)). Lines colors: $\mathcal{T}_{h,1}$ (—), $\mathcal{T}_{h,2}$ (—), $\mathcal{T}_{h,3}$ (—), and $\mathcal{T}_{h,4}$ (—).

In conclusion, by increasing the total number of degrees of freedom the conduction velocity decreases. We remark that the propagating velocities of the action potential obtained on the two finer meshes $\mathcal{T}_{h,3}$ and $\mathcal{T}_{h,4}$ are similar and slower with respect to the velocity obtained on the two coarser meshes. This indicates that with coarser meshes the actual propagation velocity is overestimated.

After showing the effect of the total numbers of degrees of freedom on the approximation of conduction velocity, we want to compare the fronts of the solutions obtained on the four considered meshes. We want to understand if the fronts of the solutions are qualitatively similar one to each other, in order to understand how much impact the different conduction velocity has on the quality of the approximated solution. Moreover, this analysis helps us to understand the relation between the quality of the solution at the front and the total number of degrees of freedom.



(a) $\mathcal{T}_{h,1}$ (b) $\mathcal{T}_{h,2}$

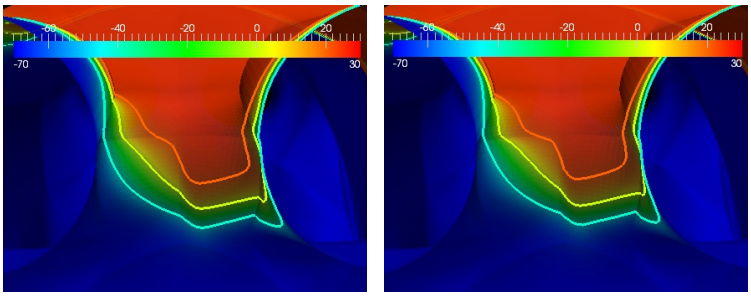
(c) $\mathcal{T}_{h,3}$ (d) $\mathcal{T}_{h,4}$

Figure 5.8: Front of the approximated solution at $t = 25\ ms$ of the four different meshes, in the region shown in Figure (5.7(a))

In Figure (5.8) we compare the solution at time $t = 25\ ms$, in the region in the solid black outline shown in Figure (5.7(a)). The fronts at the values of -40, 0, and 20 mV are highlighted. We observe that the solutions obtained on the two finer meshes, (5.8(c)) and (5.8(c)), are qualitatively similar, while the solutions obtained on the two coarser meshes, (5.8(a)) and (5.8(b)), are qualitatively different . The interesting fact is that by increasing the total number of DOFs, the three highlighted fronts are smoothed out and the distance between the fronts decreases.

We want now to analyze the quality of the fronts of the solutions between the lower and upper pulmonary veins (with respect to Figure (5.7(a))) at the times $t = 33\ ms$ and $t = 67.5\ ms$ respectively. From the analysis made on the propagating velocity of the fronts, we learn that in the two latter region, the conduction velocity of the four solutions are different. Thus we expect to remark differences also on the quality of the whole solution, in particular for the region between the two upper pulmonary veins (with respect to Figure (5.7(a))).

In Figure (5.9) the solutions for the lower pulmonary veins is reported. Again the fronts of the solutions at the values of -40, 0, and 20 mV are highlighted. In this case the influence of the total number of degrees of freedom is less emphatic with respect to the fronts reported in Figure (5.8), but again the two solutions computed on the two finer meshes, (5.9(c)) and (5.9(d)), are qualitatively similar, while the two others fronts of the solutions computed on $\mathcal{T}_{h,1}$ and $\mathcal{T}_{h,2}$, (5.9(a)) and (5.9(b)),

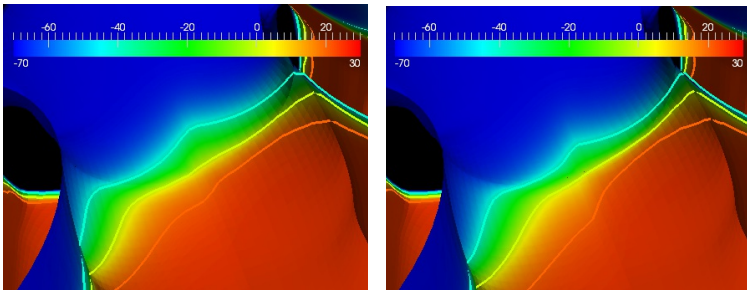


(a) $\mathcal{T}_{h,1}$ (b) $\mathcal{T}_{h,2}$

(c) $\mathcal{T}_{h,3}$ (d) $\mathcal{T}_{h,4}$

Figure 5.9: Front of the approximated solution at $t = 33\ ms$ of the four different meshes, in the region between the bottom pulmonary veins (with respect to Figure (5.7(a)))

present some regions along the fronts that are qualitatively different to the finer ones.



(a) $\mathcal{T}_{h,1}$ (b) $\mathcal{T}_{h,2}$

(c) $\mathcal{T}_{h,3}$ (d) $\mathcal{T}_{h,4}$

Figure 5.10: Front of the approximated solution at $t = 67.5\ ms$ of the four different meshes, in the region upper the bottom pulmonary veins (with respect to Figure (5.7(a)))

Finally, the solutions obtained at time $t = 67.5\ ms$ between the lower pulmonary veins are shown in Figure (5.10). In this case we can clearly observe the effect that the total number of degrees of

freedom have on the front of the solution. Specifically, when comparing the fronts obtained on the two coarser meshes with the ones obtained on the finer meshes. The differences on the approximation of the solutions at the fronts are consequences of the fact that the conduction velocities of the fronts are different for the four solutions, Figure (5.10). But we observe that the two solutions that are qualitatively most similar, are the ones obtained on the finer meshes, Figures (5.10(c)) and (5.10(d)).

Conclusions

In this project, we first studied the advantages of using Isogeometric Analysis for the spatial discretization of a benchmark electrophysiology problems arising in cardiac applications. We verified that high continuous NURBS (Non-Uniform Rational B-Splines) basis functions provide an efficient way of approximating the conduction velocity of the front of the action potential in terms of total number of degrees of freedom (DOFs). In fact, by using linear polynomial basis functions the actual value of the conduction velocity is achieved only for fine meshes and thus for a large number of DOFs. Indeed, by considering high order polynomial basis functions globally C^0-continuous, the true propagating velocity of the front is achieved with a quite large number of DOFs, which made the computation time high. Thus, we conclude that high order continuous basis functions represent an alternative, which permits to optimize the DOFs of the system to be solved, by performing a $k-$refinement on the NURBS basis functions.

The second problem we considered was the Aliev-Panfilov model [1] describing the propagation of the action potential in the cardiac tissue. Four aspects of the simulation of the Aliev-Panfilov have been emphasized: the approximation of the conduction velocity, the approximation of the action potential duration, the simulation of spiral wave, and the simulation of complex dynamics. The more important outcome is that high order continuous basis functions efficiently approximate the conduction velocity of the traveling for a small numbers of DOFs. For the other three numerical aspects we confirm that the $k-$refinement, peculiar for NURBS basis functions, provides a way to optimize the quality of the simulations in terms of DOFs.

The last part of the project is dedicated to the realistic simulation of the Mitchell-Schaeffer model [2] on the human left atrium represented by means of a NURBS surface. The fiber orientation of the cardiac tissue were taken into account by solving the Laplace-Beltrami problem on the atrium; the normalized gradient of the obtained approximated solution was then used to define locally the components of the anisotropic conduction tensor for the action potential model. The NURBS basis functions used for the geometrical representation, and thus for the approximation of the model, were globally C^1-continuous, but in some lines only C^0-continuous. We compared the conduction velocities of the solutions obtained for different meshes. We verified that the computed velocity of the front converges to the correct value from "above", for which we deduce that coarser meshes overestimate the velocity values.

We conclude that, Isogeometrical Analysis may play an important role in increasing the efficiency and accuracy of numerical simulations of electrophysiological models, in virtue of the $k-$refinement procedure. Moreover, the possibility of representing exactly complex geometries, allowed the numerical solutions off action potential on smooth surfaces, thus minimizing the effect of the geometrical approximation on the representation of fronts and velocities.

One of the future developments of this work, consists in the study of more complex ionic models of the activation of the human atria. Another extension may be the study of the coupling between Partial Differential Equation and Ordinary Differential Equations arising in the numerical approximation of electrophysiological models by means of IGA.

Acknowledgements

First of all, I want to thank Prof. Alfio Quarteroni for gave me the opportunity to carry out my Master thesis within the Chair of Modelling and Scientific Computing (CMCS) at the EPFL.

A very special thanks goes out to my supervisors Dr. Luca Dede' and Dr. Toni Lassila for helping me during the whole duration of this project, for their kindness, and availability. I sincerely appreciated.

I would also like to acknowledge Andrea Bartezzaghi for his help and for having taught me really advanced underground fashion terminal's tricks (alias ls = "eject").

Je remercie tous les amis que j'ai rencontré à l'EPFL pour tous les moments inoubliables passés ensemble. *(Petit)**Lemme***: Vous êtes les meilleurs, d'où C^∞ sur le tore. La preuve triviale est laissée au lecteur.

Ringrazio la mia piccola Stella che con il suo sorriso ha illuminato le mie giornate. Ringrazio di cuore mia sorella Barbara, Marco e la piccola Elisa per tutto l'affetto dimostratomi e per essere, qui a Losanna, il mio punto di riferimento. Un grazie speciale va anche a mio fratello Gabriele per tutto il suo sostegno. Grande brate.

Infine, il ringraziamento più importante va ai miei genitori per il lror sostegno, il loro affetto e per aver sempre creduto in quel ragazzino che al gheva sempro da tofignà quaicos, che al stava mai fermo e che da cres al gheva propi mia vöia.

Lausanne, June 20, 2014

Bibliography

[1] R.R. Aliev and A.V. Panfilov. “A simple two-variable model of cardiac Excitation”. In: *Cha., Sol. & Fract.* 7 (1996), pp. 293–301.

[2] C.C. Mitchell and D.G. Schaeffer. “A two current model for the dynamics of cardiac membrane”. In: *Bull, of Math. Bio.* 65 (2003), pp. 767–793.

[3] J. Keener and J. Sneyd. *Mathematical Physiology: Cellular Physiology*. Springer, 2009.

[4] J. Keener and J. Sneyd. *Mathematical Physiology: System Physiology*. Springer, 2009.

[5] A. Quarteroni and A. Valli. *Numerical Approximation of Partial Differential Equation*. Spinger-Verlag, Berlin, 2008.

[6] A. Quarteroni. *Numerical Models for Differential Problems*. Spinger Milan, Milano, 2009.

[7] D. Ambrosi, A. Quarteroni, and G. Rozza. *Modeling of Physiological Flows*. Springer-Verlag, Italy, 2012.

[8] S. Kirshnamoorthi, M. Sarkar, and W.S. Klug. “Numerical quadrature and operator splitting in finite element methods for cardiac electrophysiology”. In: *Int. J. Numer. Meth. Biomed. Engng.* (2013).

[9] J.A. Cottrell, T.J.R Hughes, and Y. Bazilevs. *Isogeometric Analysis: Toward integration of CAD and FEA*. John Wiley and Sons, 2009.

[10] T.J.R. Huges, J.A. Cottrell, and Y. Bazilevs. “Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement”. In: *Comput. Methods Appl. Mech. and Engrg.* 194 (2005), pp. 4135–4185.

[11] I.J. LeGrice et al. “Laminar Structure of the Heart: Ventricular myocyte arrangement and connective tissue architecture in the dog”. In: *Am. J. Physiol.* 269(38) (1995), H571–H582.

[12] I.J. LeGrice, B.H. Smaill, and P.J Hunter. “Laminar structure of the heart: a mathematical model”. In: *Am. J. Physiol.* 272(41) (1997), H2466–H2476.

[13] A. L. Hodgkin and A.F Huxley. “A quantitative description of membrane current and its application to conduction and excitation in nerve”. In: *J. Physiol.* 117 (1952), pp. 500–544.

[14] R. FitzHugh. “Impulses and physiological states in theoretical models of nerve membrane”. In: *Biophys. J.* 1 (1961), pp. 445–465.

[15] J.M. Greenberg and S.P. Hastings. “Spatial patterns for discrete models of diffusion in excitable media”. In: *SIAM J. Appl. Math.* 34 (1978), pp. 515–523.

[16] A. Bueno-Orovio, E.M. Cherry, and F.H. Fenton. “Minimal model for human ventricular action potentials in tissue”. In: *J. of Theor. Biol.* (2008).

[17] K.H.W.J. ten Tusscher, D. Noble, and A.V. Panfilov. “A model for human ventricular tissue”. In: *Am. J. Physiol. Heart. Circ. Physiol.* 286 (2004), H1573–H1589.

[18] K.H.W.J ten Tusscher and A.V. Panfilov. “Alternans and spiral breakup in a human ventricular tissue model”. In: *Am. J. Physiol. Heart. Circ. Physiol.* 291 (2006), H1088–H1100.

[19] J.S. Nagumo, S. Arimoto, and S. Yoshizawa. “An active pulse transmission line simulating nerve axon”. In: *Proc. IRE* 50 (1962), pp. 2061–2071.

[20] M. Countermanche, W. Skaggs, and A.T. Winfree. “Stable three-dimensioanl action potential circulation in the FitzHugh-Nagumo model”. In: *Physica D* 41 (1990), pp. 173–183.

[21] V. Elharrar and B. Surawicz. “Cycle length effect on resitution of action potential duration in dog cardiac fibers”. In: *Am. J. Physiol.* 244 (1983), H782–H792.

[22] L. Piegl and W. Tiller. *The NURBS Book*. Spinger-Verlag, New York, 1997.

[23] J. Chung and G.M. Hulbert. “A time integration algorithm for structural dynamics with improved numerical dissipation: the generalized-a method”. In: *J. of App. Mech.* 60 (1993), 371–375.

[24] J. Liu et al. “Isogeometric analysis of the advective Cahn-Hilliard equation: Spinodal decomposition under shear flow”. In: *J. of Comp. Phys.* 242 (2013), pp. 321–350.

[25] S. Göktepe and K. Kuhl. “Electromechanics of the htwo-variableseart: a unified approach to the strongly coupled excitation-contraction problem”. In: *Comp. Mech.* 79 (2010), pp. 227–243.

[26] S. Rossi. “Anisotropic Modeling of Cardiac Mechanical Activation”. PhD thesis. EPFL, 2014.

[27] Z. Qu and A. Garfinkel. “An advanced algorithm for solving partial differential equation in cardiac conduction”. In: *IEEE Trans Biomed. Eng.* 46 (1999), pp. 1166–1168.

[28] A. Hundsdorfer and J.G. Verwer. *Numerical Solution of Time-Dependent Advection-Diffusion-Reaction Equations*. Spinger, Amsterdam, 2003.

[29] K.E. Jansen, C.H. Whiting, and G.M. Hulbert. “A generalized$-\alpha$ method for integrating the filtered Navier–Stokes equations with a stabilized finite element method”. In: *Comput. Methods Appl. Mech. and Engrg.* 190 (2000), 305–319.

[30] Y. Bazilevs et al. “Variational multiscale residual-based turbulence modeling for large eddy simulation of incompressible flows”. In: *Comput. Methods Appl. Mech. and Engrg.* 197 (2007), 173–201.

[31] R. Gray and J. Jalife. “Spiral waves and the heart”. In: *Int. J. Bifurcation Chaos App. Sci. Eng.* 6 (1996), p. 415.

[32] F. Fenton and A. Karma. “Vortex dynamic in three-dimensional continuous myocardium with fiber rotation: Filament instability and fibrillation”. In: *Chaos* 8 (1998), pp. 20–47.

[33] E.M. CHerry and F. Fenton. “Visualization of spiral and scroll waves in simulated and experimental cardiac tissue”. In: *New J. Phys* 10 (2008), pp. 1–43.

[34] D.M. Harrild and C.S. Henriquez. “A computer model of normal conduction in the human atria”. In: *Cir. Res.* 87 (2000), e25–e36.

[35] C. Tobón et al. “A three-dmensional human atrial model with fiber orientation. Electrograms and Arrhythmic activation patterns relationship”. In: *PLoS ONE* 8 (2013), pp. 1–13.

[36] S.Y. Ho, R.H. Anderson, and D. Sánchez-Quintana. “Atrial structure and fibers: morphologic bases of atrial conduction”. In: *Card. R.* 54 (2002), pp. 325–336.

[37] A. Collin et al. “Surface-based electrophysiology modeling and assessment of physiological simulations in atria”. In: *FIMH - 7th Int. Conf. on Funct. Imag. and Mod. of the Heart* 7945 (2013), pp. 352–359.

[38] S. Sakamoto et al. “Interatrial electrical connections: the precise location and preferental conduction”. In: *J. Card. Electro.* 16 (2005), pp. 1077–1086.