



HAL
open science

The resolution of the inverse problem in electrocardiography

Alejandro Lopez-Rincon Lopez Rincon

► **To cite this version:**

Alejandro Lopez-Rincon Lopez Rincon. The resolution of the inverse problem in electrocardiography. General Mathematics [math.GM]. Université Sciences et Technologies - Bordeaux I, 2013. English. NNT : 2013BOR15261 . tel-01079063

HAL Id: tel-01079063

<https://theses.hal.science/tel-01079063>

Submitted on 25 Nov 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

THÈSE

PRÉSENTÉE A

L'UNIVERSITÉ BORDEAUX 1

ÉCOLE DOCTORALE DES SCIENCES Mathématiques

Par Alejandro, LOPEZ RINCON

POUR OBTENIR LE GRADE DE

DOCTEUR

SPÉCIALITÉ : MATHÉMATIQUES APPLIQUÉES

Le problème inverse en l'Electrocardiographie

Directeur de recherche : Bedreddine Ainseba

Soutenue le : 20/12/2013

Devant la commission d'examen formée de :

M. AINSEBA, Bedreddine	Professeur Université de Bordeaux
M. BENDAHMANE, Mostafa	Maître de conférences HDR
M. EL-BADDIA, Abdellatif	Professeur Université de Technologie Compiègne
M. LANGLAIS, Michel	Professeur Émérite Université de Bordeaux
M. MILNER, FABIO	Professeur Établissement étranger
M. OMRANE Abdennebi	Maître de conférences HDR

Chapter 1

Chapter Introduction

1.1 Introduction

Cardiovascular diseases (CVDs) are one of the leading causes of death all over the world: more people die annually from CVDs than from any other cause [1]. The number of people who die from CVDs, mainly from heart disease and stroke, will increase to reach 23.3 million by 2030 [1, 2]. CVDs are projected to remain the leading cause of death worldwide. Therefore any measure that may help curtail such devastating effects of CVDs is welcome by the medical profession and public health officials, and mathematical models to understand the mechanisms of the electrical activity of the heart could lead to a more accurate diagnosis and earlier, and in return reduce the quantity of heart-related deaths. The obvious difficulty of performing direct measurements on the heart's surface, due to such procedures being highly invasive, has motivated a wide interest in the numerical simulation of cardiac models, and the development of systems to measure and visualize the electrical activity of the heart. For example in order to measure the electrical activity directly in the heart's surface it will be necessary an intra-cardiac electrode. An intra-cardiac electrode is a small electronic device which is attached to one of the walls of the heart and measures the potential between two close points. Although this gives a very precise idea of the electrical activity locally, it cannot be used to give an approach of the overall electrical functioning. This is caused by the used reference to measure the potential difference between the points. In other words, the ground or base potential reference will not be the same if trying to couple a local and global system. To understand this; we are actually measuring the difference between points A, B from the base reference, but if we change the reference the voltage will be affected. The heart is a muscle (myocardium) built by fibers that are sensitive to electric potentials and electro-chemical reactions make it contract and expand. These electrochemical

reactions are given by the addition of different currents in the heart cells produced by the change in ionic concentrations. It consists of 4 chambers that are used to pump blood through the body. The superior chambers are called atria, and the lower ones - ventricles. The function of how the heart expands and contracts to pump blood is similar to a piezoelectric effect. This is known as heartbeat or systole. This effect produces mechanical displacement given an electrical stimulus. The heart tissue can be considered as a material with piezoelectric properties. The heart activity propagates throughout the body and, thus, can be measured. To measure the heartbeat and extract overall heart characteristics we use the electrocardiogram (ECG). The ECG is the representation of the measurements of action potentials originated in the heart by measurements taken in the thorax.

It is a physical test widely used to determine the condition of the heart by non-invasive means. The ECG gives a set of measures that summarize the heart's electrical activity. These measures, or waves, allow the diagnosis of certain diseases, and give information about the heart's condition. It is a test performed to evaluate the conductivity of the heart muscle and, in an indirect sense, the condition of the organ as a pump. In other words, the ECG is a graphic representation of the bioelectrical activity of the heart's muscle.

To measure the bioelectric potentials, or the ECG, an array of electrodes is placed on skin of the patient's chest. In comparison to the intra-cardiac electrodes, these do not present a high risk. These electrodes array will consist of three principal leads, each one using a bipolar configuration (two electrodes) and a reference electrode. The first one will be given by the left arm, left leg. The second one will be made using the right arm and left leg, and the third one using the left and right arm. For all three of them, the reference will be the same; the right leg. From this we create a global reference using the average of the three leads. Using this reference we can create monopolar leads. The monopolar lead measures the difference between one point (measured by an electrode) and the global reference. In the Figure 1.1 we can see the example of an ECG lead.

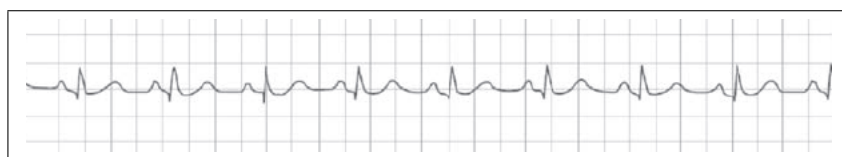


FIGURE 1.1: Example of ECG.(From [3])

The electrodes should have a galvanic contact with the body. The resulting electrical measurements of the heart are characterized by low amplitude and are highly sensitivity to noise. To amplify these signals it is necessary to use a complex electronic device, which at the same time, should reduce the noise in the signals.

The overall electrical activity in the heart's surface, which is measured using the electrodes, is generated by the cumulative effect of its cells and their action potential. The action potential is an electrical response in excitable cells. Excitable cells respond to a stimulus. When the stimulus surpasses threshold then voltage increases rapidly, and finally it propagates the electrical pulse [4] to the next cell though gap junctions which unite them. In Figure 1.2 there is a typical action potential of a ventricular cell. In these images we can see there are four phases in the action potential. The sections can be explained as

- Phase 0 is the fast change in the voltage or depolarization, created by the stimulus.
- Phase 1 is a first repolarization.
- Phase 2 is a brief state of equilibrium within the ionic currents.
- Phase 3 is the repolarization of the cell.
- Phase 4 is the repose condition.

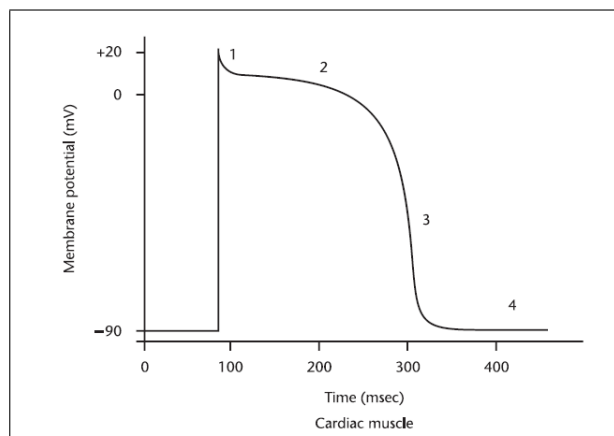


FIGURE 1.2: Action Potential from a ventricular myocardial cell.(From [3])

The heart tissue is made of many excitable fibers consisting of cardiac cells. The cells of muscles and nerves produce action potentials by the movement of ions through a semi-permeable membrane. The ionic movement produces a difference in potential. For example, if two electrodes are placed near an excitable cell, when it depolarizes a

difference of potential is generated between the two electrodes. Using bioelectric instrumentation we can measure this voltage created by the electrical activity of nerves and muscles.

The first model to describe the action potential is the Hodgkin and A.F. Huxley model of 1952 [5]. Measuring a giant nerve fiber of a squid, they developed a mathematical description of the flow of electric current through the surface membrane and their interaction to produce an electric potential. Their model describes the potential as a result of the interaction of the ionic currents created by Sodium and Potassium. This model was the basis for many models, including the Noble's model in 1962, which described the cardiac cell and considered the potential of the Purkinje fibers of the heart [6]. Another model highly use is the Fenton Karma which gives similar results, but using only 2 non-dimensional variables to describe the potential created from the Van Der Pol Oscillator. The electrical activity of the heart is fired by the nervous system. An impulse is received in the sinoatrial (SA) node. The SA node is located in the atria and it gives the pace of the heart. When the electrical current is fired by the SA node the atria contracts and the ventricles fill with blood depolarizing the cardiac cells in the atria. The electrical impulse propagates to the Atrioventricular (AV) node, then to the bundle of His and finally through the Purkinje fibers which produces the necessary stimulus for the cardiac cells in the ventricles to contract. The whole conduction system is shown in Figure 1.3.

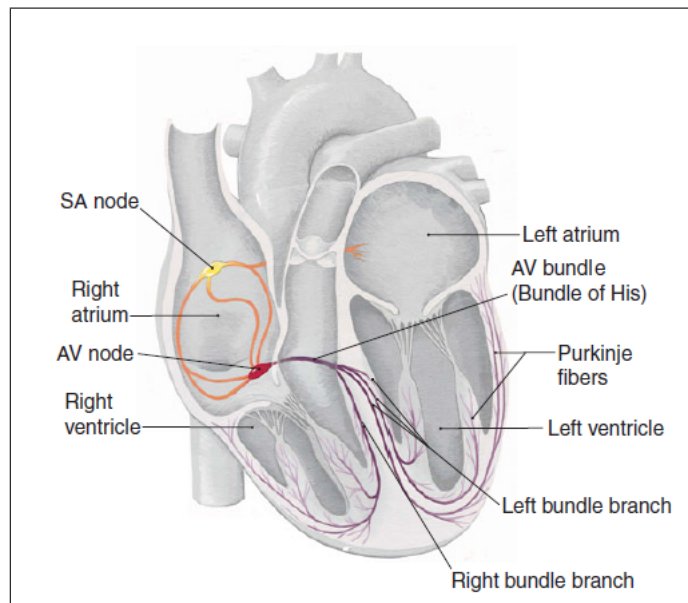


FIGURE 1.3: heart conduction system.(From [7])

The whole depolarization cycle is outlined in Figure 1.4.

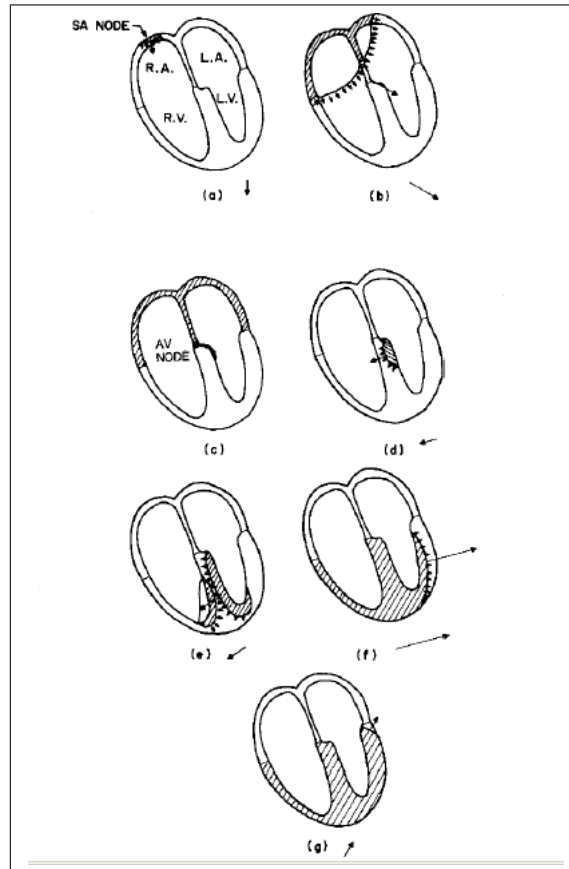


FIGURE 1.4: Depolarization cycle from the heart.(From [8])

The numerical simulations and solutions of ECG (direct and inverse problems) have received much attention for many years as a means to describe the electromechanical mechanisms in the heart and there have been many contributions on this subject. In this context we define direct as the forward simulation, that will be the propagation of the electrical activity from the heart to the thorax. The inverse problem summarized is the reconstruction of the original source from the measures in the thorax.

In electrocardiography, computational models of bioelectric phenomena from the heart have existed for over 40 years. In these models the body is modeled as a volume conductor, and the source is the heart. The first approach is to take into account that the change between two measurements is slow enough to consider a quasi-static condition, and the volume conductor condition described it with the Poisson's equation. Then the heart's activity is described with different models. One of the most accepted models to describe the electrical activity of the heart is the Bidomain model [9]. The Bidomain model was developed in 1978 and is used for numerical simulations of the activity of the heart. The model consists of a set of equations that describe the electrical properties and behavior of the heart considering it anisotropic in the extra and intra cardiac domains. In addition, the Bidomain model takes into account ionic properties of the

cardiac cell. A largely used simplification of the Bidomain model is the Monodomain model which simplifies the model equations and therefore its simulation by considering similar conductivities for both domains.

The inverse problem in electrocardiography is to determine, in a non-invasive way, information about the electrophysiological state of the heart that visual inspection of the electrocardiogram or the body surface mapping cannot provide. It consists in the reconstruction of the electrical cardiac sources from body surface potential measurements (BSPMs), considering the torso as a volume conductor.

In Figure 1.5 there is a triangulation (discretization) of the geometry of the body of a patient, and in green the positioning of the electrodes. This geometry is created from a sequence of images (x-ray) provided by the hospital Haut Leveque at Bordeaux. These electrodes are in a monopolar lead configuration. The quantity of electrodes exceeds the normal 12 derivation standard, therefore we call this a high density electrode array.

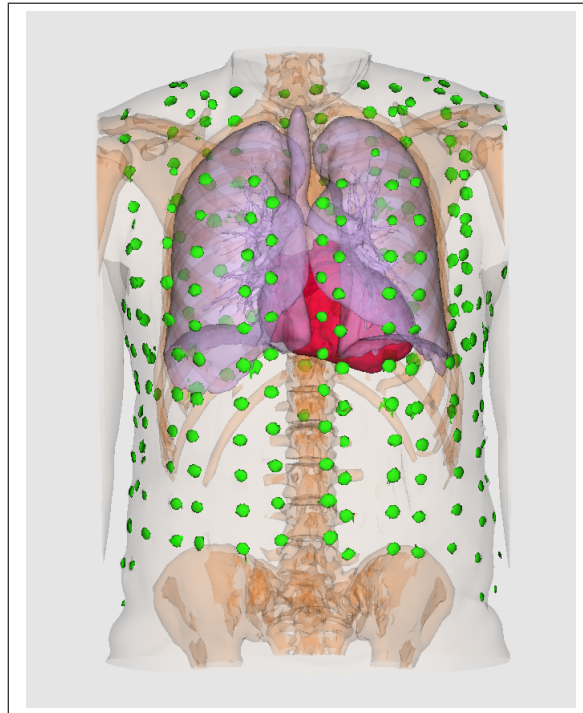


FIGURE 1.5: Example of triangulation of the torso and the positioning of an array of surface electrodes on a patient.

The static bidomain model is used in [10, 11] to compute the size and position of ischemic regions in the human heart. The problem there consists in looking for the transmembrane potential over the heart that gives the observed data over the thorax. In [12] the authors solved the inverse ECG problem by a regularization method of the Laplace ill-posed problem with two given boundary conditions on the torso. This approach is very similar to the one used by [10, 11], but the domain in this case corresponds to the

whole torso, instead of the thorax with a hole corresponding to the heart. In some sense the unknown in [12] corresponds to the potential on the surface of the heart (boundary) and in the case of [10, 11] corresponds to the potential on the heart (volume).

The approach used in [10, 11] is more realistic because of the constraints added on the heart, but still has the same difficulties as the one used by [12] because it still requires regularization on the potential. Moreover the inverse problem is still ill posed and noise and heterogeneities may disturb the computed inverse solution. The authors in [13] used another approach, the reconstruction of the activation time (time of arrival of the depolarization phase of action potential) over the heart surface. In this approach, the static bidomain model is used and the activation time can be obtained by using the inverse problem from body surface potential.

The purpose of this thesis is to solve the question of the inverse problem in electrocardiology by using a model consisting of a geometric torso model and a model of the electric activation in the myocardium (the heart) based on the bidomain model. We assume that the medium surrounding the body (the air) is nonconductive; thus, the normal derivative of the potential vanishes at the boundary of the insulating medium. Moreover, it is assumed that tissues of the thorax have a Laplace's equation to govern potential behaviors according to the theory of the Quasi-static Maxwell's equations due to low-frequency response of human tissue.

In this thesis we propose the use of a model of propagation from the heart stimulation to the torso to solve the inverse high density ECG problem. In order to make a comparison with the classical methods we visit once again, Tikhonov regularization to solve the Poisson's equation with two known boundary data on the same surface in the ECG context. In addition, we explore different techniques to solve the inverse problem in electrocardiography. We also propose new algorithms based on artificial intelligence to improve the static reconstruction and study the effects of homogeneous and non-homogeneous considerations in the thorax. For example in [14] we explain the use of artificial intelligence to solve the problem using a new two-step algorithm.

1.1.1 Outline of the thesis

In this thesis we consider the ill-posed boundary value inverse problem in electrocardiography, and solve it employing regularization techniques.

In chapter 2 we will describe the equations that summarize the forward problem for the electrical activity of the heart. We will explain the cell models like Hodgkin-Huxley and Fenton-Karma, Fitz-Hugh Nagumo that explain the ionic currents of the cardiac cells. Also we will show simulations of the models for the whole heart's electrical activity like

the Bidomain and the Monodomain that are used for the dynamic inverse in the third chapter. These continuum models consider the calculations of groups of cells, instead of each of them individually, considering the kinetics of the cell models.

The main focus of the thesis is the dynamic inverse problem in electrocardiography explained in chapter 3 using a summary of the Bidomain, and Monodomain models for the electrical activity of the heart. In this chapter we will explain how we create the operators using the Finite Element Method (FEM) technique for these two models, to show the results of our work we include several tests for the dynamic case using synthetic data. In chapter 4 we visit the classical methods to solve the inverse ECG problem. We start with the volume conductor model, that is the more simple model used for the inverse problem in electrocardiography. This is important to make a comparison between the different methods. In this chapter also we show how FEM and BEM can be used in the context of electrocardiography.

In chapter 5 we explain, and show how to create, the operators for the FEM and BEM methods numerically. We will introduce the regularization techniques and tests for Tikhonov regularization. We explain the necessary algorithms for the minimization, and the choice of parameter of regularization. To be more precise, we explain the L-Curve method to choose the parameter of regularization.

In Chapter 6 we explain the use of genetic algorithms to improve the static inverse ECG technique. In this chapter we include also a basic example for understanding genetic algorithms. This explanation is necessary to understand minimization with non linear methods. In this chapter we explain a two-step algorithm, which is the mixture of both gradient and genetic algorithms for the static inverse problem.

Chapter 2

Chapter 2. Forward Dynamic Heart Torso Equations

As mentioned in the first chapter, the heart is composed with excitable cells connected between each other, which react to an electrical stimulus. When an excitable cell receives a stimulus, the potential inside the cell (intracellular) and outside the cell (extracellular) changes, and propagates to the neighboring cell through the gap junctions. These domains are separated by the cell membrane. To consider the potential difference across the membrane we use the Bidomain model for electrical activity of the heart [15]. This model considers an average of the properties of many cells considering at the same time both domains, and taking into account the change in the voltage over time generated by the ionic currents.

2.1 The macroscopic bidomain model

The spatial domain of the heart for our models is a bounded open subset $\Omega_H \subset \mathbb{R}^3$ with a piecewise smooth boundary $\partial\Omega_H$. This represents a three-dimensional slice of the cardiac muscle regarded as inter-penetrating and superimposed (anisotropic) continuous media, namely the intracellular (i) and extracellular (e) tissues. These tissues are separated from each other (and connected at each point) by the cardiac cellular membrane.

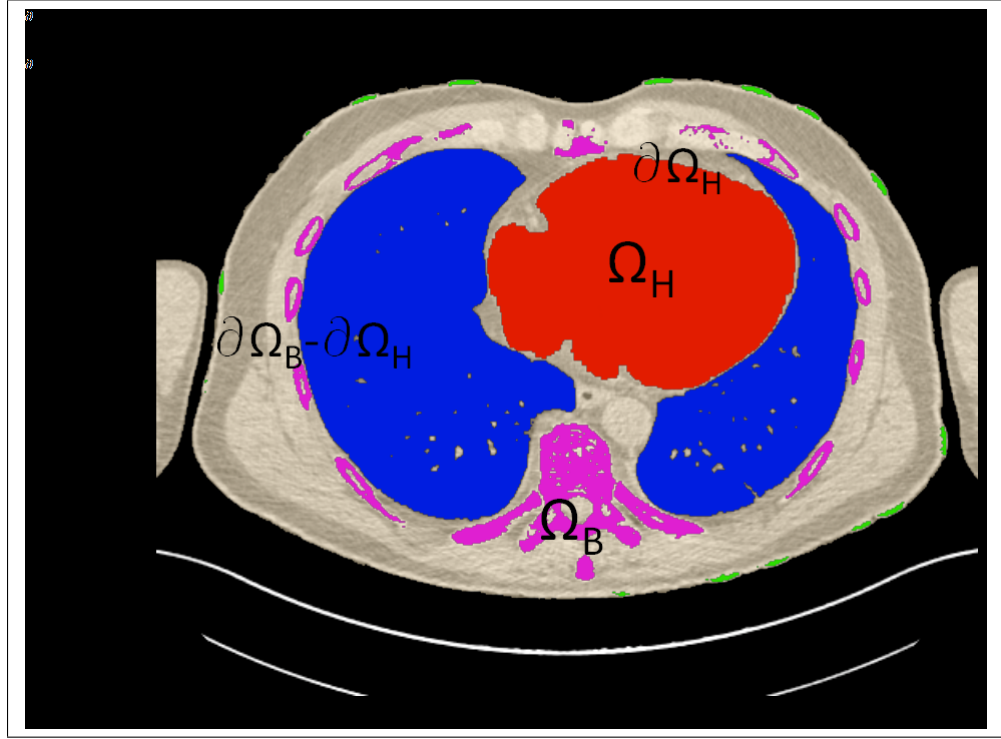


FIGURE 2.1: 2D Slice from segmentation to show the separation of domains.

The myocardium is surrounded by a volume conductor, Ω_B (the thorax) as seen in Fig. 2.1. The quantities of interest are *intracellular*, *extracellular* and the bathing medium electric potentials, $u_i = u_i(x, t)$, $u_e = u_e(x, t)$ at $(x, t) \in \Omega_{H,T} := \Omega_H \times (0, T)$, and $u_s = u_s(x, t)$, at $(x, t) \in \Omega_{B,T} := \Omega_B \times (0, T)$. The myocardium is surrounded by a volume conductor, Ω_B (the thorax). Note that ECG signals monitor the electrical activity of the heart from potential measurements at the torso skin surface $\partial\Omega_T$. The torso volume is commonly modeled as a passive conductor. The differences $v = v(x, t) := u_i - u_e$ and u_s are known as the *transmembrane potential* and the depth voltage between the tissue and the bath, respectively. The conductivity of the tissue is represented by scaled tensors $\mathbf{M}_i(x)$ and $\mathbf{M}_e(x)$ given by

$$\mathbf{M}_j(x) = \sigma_j^t \mathbf{I} + (\sigma_j^l - \sigma_j^t) \mathbf{a}_l(x) \mathbf{a}_l^T(x),$$

where $\sigma_j^l = \sigma_j^l(x) \in C^1(\mathbb{R}^2)$ and $\sigma_j^t = \sigma_j^t(x) \in C^1(\mathbb{R}^2)$, $j \in \{e, i\}$, are the intra- and extracellular conductivities along and transversal to the direction of the fiber (parallel to $\mathbf{a}_l(x)$), respectively. The conductivity tensor of the bathing medium \mathbf{M}_s is assumed to be a diagonal matrix.

For fibers aligned with the axis, $\mathbf{M}_i(x)$ and $\mathbf{M}_e(x)$ are diagonal matrices: $\mathbf{M}_i(x) = \text{diag}(\sigma_i^l, \sigma_i^t)$ and $\mathbf{M}_e(x) = \text{diag}(\sigma_e^l, \sigma_e^t)$. When the so-called *anisotropy ratios* σ_i^l/σ_i^t and σ_e^l/σ_e^t are

equal, we are in the case of *equal anisotropy*, but generally the conductivities in the longitudinal direction l are higher than those across the fiber (direction t); such a case is called *strong anisotropy* of electrical conductivity. When the fibers rotate from bottom to top, this type of anisotropy is often referred to as *rotational anisotropy*.

The bidomain model is given by the following coupled reaction-diffusion system [15]:

$$\begin{aligned} \beta c_m \partial_t v - \nabla \cdot (\mathbf{M}_i(x) \nabla u_i) + \beta I_{\text{ion}}(v, w) &= I_i, & (x, t) \in \Omega_{H,T}, \\ \beta c_m \partial_t v + \nabla \cdot (\mathbf{M}_e(x) \nabla u_e) + \beta I_{\text{ion}}(v, w) &= I_e, & (x, t) \in \Omega_{H,T}, \\ \partial_t w - H(v, w) &= 0, & (x, t) \in \Omega_{H,T}, \\ -\nabla \cdot (\mathbf{M}_s(x) \nabla u_s) &= 0, & (x, t) \in \Omega_{B,T}. \end{aligned} \quad (2.1.1)$$

Here, $c_m > 0$ is the *surface capacitance* of the membrane, β is the surface-to-volume ratio, $w(x, t)$ is the *gating* or *recovery variable*, which also takes into account the concentration variables, and I_e is an external current stimulus. The functions $H(v, w)$ and $I_{\text{ion}}(v, w)$ correspond to the fairly simple Mitchell-Schaeffer membrane model [16] for the membrane and ionic currents:

$$H(v, w) = \frac{w_\infty(v/v_p) - w}{R_m c_m \eta_\infty(v/v_p)}, \quad I_{\text{ion}}(v, w) = \frac{v_p}{R_m} \left(\frac{v}{v_p \eta_2} - \frac{v^2(1 - v/v_p)w}{v_p^2 \eta_1} \right), \quad (2.1.2)$$

where the dimensionless functions $\eta_\infty(s)$ and $w_\infty(s)$ are given by $\eta_\infty(s) = \eta_3 + (\eta_4 - \eta_3)\mathcal{H}(s - \eta_5)$ and $w_\infty(s) = \mathcal{H}(s - \eta_5)$, where \mathcal{H} denotes the Heaviside function, R_m is the surface resistivity of the membrane, and v_p and η_1, \dots, η_5 are given parameters. A simpler choice for the membrane kinetics is the widely known FitzHugh-Nagumo model [17], which is often used to avoid computational difficulties arising from a large number of coupling variables. This model is specified by

$$H(v, w) = av - bw, \quad I_{\text{ion}}(v, w) = -\lambda(w - v(1 - v)(v - \theta)), \quad (2.1.3)$$

where a, b, λ, θ are given parameters.

We utilize zero flux boundary conditions for the intracellular potential (the intracellular current does not propagate outside the heart) and we assume there is a perfect transmission between the heart and the torso:

$$\begin{aligned} (\mathbf{M}_i(x) \nabla u_i) \cdot \mathbf{n} &= 0 \text{ on } \Sigma_{H,T} := \partial\Omega_H \times (0, T), \\ u_e &= u_s \text{ and } (\mathbf{M}_e(x) \nabla u_e) \cdot \mathbf{n} = (\mathbf{M}_s(x) \nabla u_s) \cdot \mathbf{n} \text{ on } \Sigma_{H,T}, \\ (\mathbf{M}_s(x) \nabla u_s) \cdot \mathbf{n}_s &= 0 \text{ on } \Sigma_{B,T} := (\partial\Omega_B - \partial\Omega_H) \times (0, T), \end{aligned} \quad (2.1.4)$$

and impose initial conditions (which are degenerate for the transmembrane potential v):

$$v(0, x) = v_0(x), \quad w(0, x) = w_0(x), \quad x \in \Omega_H. \quad (2.1.5)$$

For the solution v of the bidomain model, we require the initial datum v_0 to be compatible with (2.1.4). Therefore, if we fix both $u_j(0, x)$, $j \in \{e, i\}$ as initial data, the problem may become unsolvable, since the time derivative involves only $v = u_i - u_e$ (this is also referred as *degeneracy in time*). Thus, we impose the compatibility condition

$$\int_{\Omega_H} u_e(x, t) dx = 0 \quad \text{for a.e. } t \in (0, T). \quad (2.1.6)$$

Note that in the case that $\mathbf{M}_i \equiv \lambda \mathbf{M}_e$ for some constant $\lambda \in \mathbb{R}$ and , the system (2.1.1) is equivalent to a scalar parabolic equation for the transmembrane potential v , coupled to an ODE for the gating variable w . This parabolic equation is obtained by multiplying the first equation in (2.1.1) by $1/(1+\lambda)$, the second by $\lambda/(1+\lambda)$ and adding the resulting equations. The final *monodomain model* can be stated as follows:

$$\begin{aligned} \beta c_m \partial_t v - \nabla \cdot \left(\frac{\mathbf{M}_i}{1+\lambda} \nabla v \right) + \beta I_{\text{ion}}(v, w) &= 0, \\ \partial_t w - H(v, w) &= 0, \quad (x, t) \in \Omega_T. \end{aligned} \quad (2.1.7)$$

This model is, of course, significantly less involved and requires substantially less computational effort than the bidomain model, and even though the assumption of equal anisotropy ratios is very strong and generally unrealistic, the monodomain model is adequate for a qualitative investigation of repolarization sequences and the distribution of patterns of durations of the action potential [18].

We assume that the functions \mathbf{M}_j , $j \in \{e, i, s\}$, I_{ion} , g and H are sufficiently smooth so that the following definitions of weak solutions make sense. Furthermore, we assume that $\mathbf{M}_j \in L^\infty(\Omega)$ and $\mathbf{M}_j \boldsymbol{\xi} \cdot \boldsymbol{\xi} \geq C_M |\boldsymbol{\xi}|^2$ for a.e. $x \in \Omega$, for all $\boldsymbol{\xi} \in \mathbb{R}^2$, $j \in \{e, i, s\}$, and a constant $C_M > 0$. Observe that in our model one can decompose I_{ion} as

$$I_{\text{ion}}(v, w) =: I_{1,\text{ion}}(v) + I_{2,\text{ion}}(w).$$

Then it is straightforwardly seen that there exists a constant $C_I > 0$ such that (see e.g. [19])

$$\frac{I_{1,\text{ion}}(v_1) - I_{1,\text{ion}}(v_2)}{v_1 - v_2} \geq -C_I, \quad \forall v_1 \neq v_2. \quad (2.1.8)$$

Additionally, there is a constant $C'_I > 0$ such that

$$0 < \liminf_{|v| \rightarrow \infty} \left| \frac{I_{1,\text{ion}}(v)}{v^3} \right| \leq \limsup_{|v| \rightarrow \infty} \left| \frac{I_{1,\text{ion}}(v)}{v^3} \right| \leq C'_I. \quad (2.1.9)$$

For later reference, we now state the definitions of a weak solution for the bidomain model.

Definition 2.1.1. A five-uple $\mathbf{u} = (v, u_i, u_e, u_s, w)$ of functions is a *weak solution of the bidomain model (2.1.1)–(2.1.5)* if $v \in L^2(0, T; H^1(\Omega_H)) \cap L^4(\Omega_{H,T})$, $\partial_t v \in L^2(0, T; (H^1(\Omega))^*) + L^{4/3}(\Omega_{H,T})$, $u_i, u_e \in L^2(0, T; H^1(\Omega))$, $u_s \in L^2(0, T; H^1(\Omega_B))$, $w \in C([0, T], L^2(\Omega))$, (2.1.6) is satisfied, and the following identities hold for all test functions $\varphi_i, \varphi_e \in L^2(0, T; H^1(\Omega_H)) \cap L^4(\Omega_{H,T})$, $\varphi_s \in L^2(0, T; H^1(\Omega_B))$ and $\phi \in C([0, T], L^2(\Omega))$:

$$\begin{aligned} \beta c_m \int_0^T \langle \partial_t v, \varphi_i \rangle dt + \iint_{\Omega_{H,T}} \left\{ \mathbf{M}_i(x) \nabla u_i \cdot \nabla \varphi_i + \beta I_{\text{ion}} \varphi_i \right\} dx dt &= \iint_{\Omega_{H,T}} I_i \varphi_i dx dt, \\ \beta c_m \int_0^T \langle \partial_t v, \varphi_e \rangle dt + \iint_{\Omega_{H,T}} \left\{ -\mathbf{M}_e(x) \nabla u_e \cdot \nabla \varphi_e + \beta I_{\text{ion}} \varphi \right\} dx dt \\ &\quad - \iint_{\Omega_{B,T}} \mathbf{M}_s(x) \nabla u_s \cdot \nabla \varphi_s dx dt = \iint_{\Omega_{H,T}} I_e \varphi_e dx dt, \\ \iint_{\Omega_H} \partial_t w \phi dx dt &= \iint_{\Omega_H} H \phi dx dt. \end{aligned} \quad (2.1.10)$$

We have the following result concerning the well-posedness of our model, where the proof can be found in [20] :

Theorem 2.1.1 (Bidomain torso model). If $v_0 \in L^2(\Omega_H)$, $w_0 \in L^2(\Omega_H)$ and $I_{i,e} \in L^2(\Omega_{H,T})$, then the bidomain problem (2.1.1)–(2.1.5) possesses a unique weak solution.

First, we introduce the following closed subset of the Banach space:

$$\mathcal{K} = L^2(\Omega_{H,T}).$$

With $\bar{v} \in \mathcal{K}$ fixed, let (v, u_s, w) be the unique solution of the system

$$\left\{ \begin{array}{ll} \beta c_m \partial_t v - \nabla \cdot (\mathbf{M}_i(x) \nabla u_i) + \beta I_{\text{ion}}^\varepsilon(\bar{v}, w) = I_i, & \text{in } \Omega_{H,T}, \\ \beta c_m \partial_t v + \nabla \cdot (\mathbf{M}_e(x) \nabla u_e) + \beta I_{\text{ion}}^\varepsilon(\bar{v}, w) = I_e, & \text{in } \Omega_{H,T}, \\ \partial_t w - H(v, w) = 0, & \text{in } \Omega_{H,T}, \\ -\nabla \cdot (\mathbf{M}_s(x) \nabla u_s) = 0, & \text{in } \Omega_{B,T}, \\ (\mathbf{M}_i(x) \nabla u_i) \cdot \mathbf{n} = 0, & \text{on } \Sigma_{H,T} := \partial\Omega_H \times (0, T), \\ u_e = u_s \text{ and } (\mathbf{M}_e(x) \nabla u_e) \cdot \mathbf{n} = (\mathbf{M}_s(x) \nabla u_s) \cdot \mathbf{n}, & \text{on } \Sigma_{H,T}, \\ (\mathbf{M}_s(x) \nabla u_s) \cdot \mathbf{n}_s = 0, & \text{on } (\partial\Omega_B - \partial\Omega_H) \times (0, T), \\ u_s = u_c, & \text{on } \Sigma_B := (\partial\Omega_B - \partial\Omega_H) \times (0, T), \\ v(0, \cdot) = v_0(\cdot), \quad w(0, \cdot) = w_0(\cdot), & \text{in } \Omega_H, \end{array} \right. \quad (2.1.11)$$

where $I_{\text{ion}}^\varepsilon = \frac{I_{\text{ion}}}{1 + \varepsilon |I_{\text{ion}}|}$. Regarding the quasilinear problem (2.1.11) we have immediately : If $v_0 \in L^2(\Omega)$ and $I_{i,e} \in L^2(\Omega_{H,T})$, then there exists a weak solution $v, u_i, u_e \in L^2(0, T; H^1(\Omega_H))$, $\partial_t v \in L^2(0, T; (H^1(\Omega_H))^*)$, $u_s \in L^2(0, T; H^1(\Omega_B))$ and $w \in C(0, T; L^2(\Omega_H))$ to problem (2.1.11).

In order to prove existence of weak solutions to (2.1.11), we introduce the map $\Theta : \mathcal{K} \rightarrow \mathcal{K}$ satisfying $\Theta(\bar{v}) = v$, where v solves (2.1.11). By using the Schauder fixed-point theorem, we prove that this map admits a fixed point. First, let us show that Θ is a continuous mapping. Let $(\bar{v}_n)_n$ be a sequence in \mathcal{K} and $\bar{v} \in \mathcal{K}$ be such that $\bar{v}_n \rightarrow \bar{v}$ in $L^2(\Omega_H)$ as $n \rightarrow \infty$. Define $v_n = \Theta(\bar{v}_n)$, that is, v_n is the solution of (2.1.11) associated with \bar{v}_n . The objective is to show that v_n converges to $\Theta(\bar{v})$ in $L^2(\Omega_H)$.

Multiplying the first, the second, the third and the fourth equations in (2.1.11) by $u_{i,n}$, $-u_{e,n}$, $u_{s,n}$ and w_n , respectively, and integrating over the corresponding domains for $u_{i,n}$, $u_{e,n}$, $u_{s,n}$ and w_n , we arrive at

$$\begin{aligned} & \frac{1}{2} \frac{d}{dt} \int_{\Omega_H} (|v_n|^2 + |w_n|^2) dx + \int_{\Omega_H} \mathbf{M}_i(x) \nabla u_{i,n} \cdot \nabla u_{i,n} dx + \int_{\Omega_H} \mathbf{M}_e(x) \nabla u_{e,n} \cdot \nabla u_{e,n} dx \\ & \quad + \int_{\Omega_B} \mathbf{M}_e(x) \nabla u_{e,n} \cdot \nabla u_{e,n} dx + \int_{\Omega_H} I_{\text{ion}}^\varepsilon(\bar{v}_n, w_n) v_n dx - \int_{\Omega_H} H(v_n, w_n) w_n dx \\ & = \int_{\Omega_H} I_i(x, t) u_{i,n} dx - \int_{\Omega_H} I_e(x, t) u_{e,n} dx \\ & = \int_{\Omega_H} I_i(x, t) v_n dx - \int_{\Omega_H} (I_e(x, t) - I_i(t, x)) u_{e,n} dx. \end{aligned} \quad (2.1.12)$$

Herein we have used the continuity of the flux and the potentials of the boundary conditions in (2.1.11). In view of the compatibility condition (2.1.6), the Poincaré inequality

and the Young inequalities, it follows from (2.1.12)

$$\begin{aligned} & \frac{1}{2} \frac{d}{dt} \int_{\Omega_H} (|v_n|^2 + |w_n|^2) dx + C_1 \int_{\Omega_H} |\nabla u_{i,n}|^2 dx + C_2 \int_{\Omega_H} |\nabla u_{e,n}|^2 dx + C_3 \int_{\Omega_B} |\nabla u_{s,n}|^2 dx \\ & \leq C(\varepsilon) + C_4 \int_{\Omega_H} (|v_n|^2 + |w_n|^2) dx, \end{aligned} \quad (2.1.13)$$

for some constants $C(\varepsilon), C_1, C_2, C_3, C_4 > 0$. Therefore an application of Gronwall inequality from (2.1.13), we get

$$\int_{\Omega_H} |v_n(x, t)|^2 dx + \int_{\Omega_H} |w_n(x, t)|^2 dx \leq C_5, \text{ for all } t \in (0, T), \quad (2.1.14)$$

for some constant $C_5 > 0$. This proves the $L^\infty(0, T; L^2(\Omega))$ bound of v_n and w_n . Moreover using this and (2.1.13), we arrive at

$$\iint_{\Omega_{H,T}} |\nabla u_{i,n}|^2 dx dt + \iint_{\Omega_{H,T}} |\nabla u_{e,n}|^2 dx dt + \iint_{\Omega_{B,T}} |\nabla u_{s,n}|^2 dx dt \leq C_6, \quad (2.1.15)$$

for some constant $C_6 > 0$. Observe that there exist $v, u_i, u_e \in L^2(0, T; H^1(\Omega_T))$, $u_s \in L^2(0, T; H^1(\Omega_B))$ and $w \in C(0, T; L^2(\Omega_H))$ such that, up to extracting subsequences if necessary,

$v_n, u_{i,n}, u_{e,n} \rightarrow v, u_i, u_e$ in $L^2(0, T; H^1(\Omega_H))$ weakly, $u_{s,n} \rightarrow u_s$ in $L^2(0, T; H^1(\Omega_B))$ weakly, and $w_n \rightarrow w$ in $L^2(\Omega_T)$ strongly,

and from this the continuity of Θ on \mathcal{K} follows.

It is easy to see that $\Theta(\mathcal{K})$ is bounded in the set

$$\mathcal{W} = \{V \in L^2(0, T; H^1(\Omega_H)) : \partial_t V \in L^2(0, T; (H^1(\Omega_H))')\}. \quad (2.1.16)$$

Thus, $\mathcal{W} \hookrightarrow L^2(\Omega_T)$ is compact, thus Θ is compact. Now, by the Schauder fixed point theorem, the operator Θ has a fixed point v such that $\Theta(v) = v$. This implies that there

exists a solution $(v_\varepsilon, u_{i,\varepsilon}, u_{e,\varepsilon}, u_{s,\varepsilon}, w_\varepsilon)$ of

$$\begin{aligned}
& \iint_{\Omega_{H,T}} \left\{ \beta c_m \partial_t v_\varepsilon \varphi_i + \mathbf{M}_i(x) \nabla u_{i,\varepsilon} \cdot \nabla \varphi_i + \beta I_{\text{ion}}^\varepsilon \varphi_i \right\} dx dt = \iint_{\Omega_{H,T}} I_i \varphi_i dx dt, \\
& \iint_{\Omega_{H,T}} \left\{ \beta c_m \partial_t v_\varepsilon \varphi_e - \mathbf{M}_e(x) \nabla u_{e,\varepsilon} \cdot \nabla \varphi_e + \beta I_{\text{ion}}^\varepsilon \varphi \right\} dx dt \\
& \quad - \iint_{\Omega_{B,T}} \mathbf{M}_s(x) \nabla u_{s,\varepsilon} \cdot \nabla \varphi_s dx dt = \iint_{\Omega_{H,T}} I_e \varphi_e dx dt, \\
& \iint_{\Omega_H} w_\varepsilon \partial_t \phi dx dt = \iint_{\Omega_H} H(v_\varepsilon, w_\varepsilon) \phi dx dt,
\end{aligned} \tag{2.1.17}$$

for all test functions $\varphi_i, \varphi_e \in L^\infty(0, T, H^1(\Omega_H))$, $\varphi_s \in L^2(0, T; H^1(\Omega_B))$, and $\phi \in L^2(\Omega_H)$:

Now, substituting $\varphi_i = u_{i,\varepsilon}$, $\varphi_e = -u_{i,\varepsilon}$, $\varphi_s = u_{s,\varepsilon}$ and $\phi = w_\varepsilon$ in (2.1.17). The result is

$$\begin{aligned}
& \frac{1}{2} \frac{d}{dt} \int_{\Omega_H} (|v_\varepsilon|^2 + |w_\varepsilon|^2) dx + \int_{\Omega_H} \mathbf{M}_i(x) \nabla u_{i,\varepsilon} \cdot \nabla u_{i,\varepsilon} dx + \int_{\Omega_H} \mathbf{M}_e(x) \nabla u_{e,\varepsilon} \cdot \nabla u_{e,\varepsilon} dx \\
& \quad + \int_{\Omega_B} \mathbf{M}_e(x) \nabla u_{e,\varepsilon} \cdot \nabla u_{e,\varepsilon} dx + \int_{\Omega_H} \left(I_{\text{ion}}^\varepsilon v_\varepsilon + C_h \frac{|v_\varepsilon|^2}{1 + \varepsilon |I_{\text{ion}}|} \right) dx + \\
& \int_{\Omega_H} H(v_\varepsilon, w_\varepsilon) w_\varepsilon dx \\
& = \int_{\Omega_H} I_i(x, t) u_{i,\varepsilon} dx - \int_{\Omega_H} I_e(x, t) u_{e,\varepsilon} dx + C_h \int_{\Omega_H} \frac{|v_\varepsilon|^2}{1 + \varepsilon |I_{\text{ion}}|} dx \\
& = \int_{\Omega_H} I_i(x, t) v_\varepsilon dx - \int_{\Omega_H} (I_e(x, t) - I_i(t, x)) u_{e,\varepsilon} dx + C_h \int_{\Omega_H} \frac{|v_\varepsilon|^2}{1 + \varepsilon |I_{\text{ion}}|} dx.
\end{aligned} \tag{2.1.18}$$

Using the conditions (2.1.8) and (2.1.9) on I_{ion} , the compatibility condition (2.1.6), Poincare inequality and Young inequalities, we obtain

$$\begin{aligned}
& \frac{1}{2} \frac{d}{dt} \int_{\Omega_H} (|v_\varepsilon|^2 + |w_\varepsilon|^2) dx + C_7 \int_{\Omega_H} |\nabla u_{i,\varepsilon}|^2 dx + C_8 \int_{\Omega_H} |\nabla u_{e,\varepsilon}|^2 dx + C_9 \int_{\Omega_B} |\nabla u_{s,\varepsilon}|^2 dx \\
& \leq C_{10} + C_{11} \int_{\Omega_H} (|v_\varepsilon|^2 + |w_\varepsilon|^2) dx,
\end{aligned} \tag{2.1.19}$$

for some constants $C_7, C_8, C_9, C_{10}, C_{11} > 0$ that are independent of ε . Therefore an application of Gronwall inequality we get the $L^\infty(0, T; L^2(\Omega))$ bound of v_ε and w_ε . Using this, (2.1.19) and the condition on h , we get

$$\iint_{\Omega_{H,T}} |I_{\text{ion}}^\varepsilon v_\varepsilon| dx dt + \iint_{\Omega_{H,T}} |\nabla u_{i,\varepsilon}|^2 dx + \iint_{\Omega_{H,T}} |\nabla u_{e,\varepsilon}|^2 dx + \int_{\Omega_{B,T}} |\nabla u_{s,\varepsilon}|^2 dx dt \leq C_{12}, \tag{2.1.20}$$

for some constant $C_{12} > 0$ not depending on ε . Note that the consequence of (2.1.20) is

$$\iint_{\Omega_{H,T}} |v_\varepsilon|^4 dx dt \leq C_{13}, \quad (2.1.21)$$

for some constant $C_{13} > 0$. In view of (2.1.19), (2.1.21) and (2.1.21), and thanks of the assumption on I_{ion} , we can assume there exist limit functions v, u_i, u_e, u_s, w such that as $\varepsilon \rightarrow 0$ the following convergences hold (modulo extraction of subsequences, which we do not bother to relabel):

$$\begin{aligned} v_\varepsilon &\rightarrow v \text{ a.e. in } \Omega_{H,T}, \text{ strongly in } L^2(\Omega_{H,T}), \text{ and weakly in } L^2(0, T; H^1(\Omega_H)), \\ u_{s,\varepsilon} &\rightarrow u_s \text{ weakly in } L^2(0, T; H^1(\Omega_B)), w_\varepsilon \rightarrow w \text{ strongly in } L^2(\Omega_{H,T}), \\ I_{\text{ion}}^\varepsilon(v_\varepsilon, w_\varepsilon) &\rightarrow I_{\text{ion}}(v, w) \text{ a.e. in } \Omega_{H,T} \text{ and strongly in } L^1(\Omega_T). \end{aligned} \quad (2.1.22)$$

Thanks to all these convergences, it is easy to see that the limit triple (v, u_i, u_e, u_s, w) is a weak solution of (2.1.7).

2.2 Numerical Approximation of the Bidomain Torso Model

To simplify we will use the following notation for the bidomain model:

$$\mathbf{M}_i(x) = \frac{1}{C_m \beta} D \begin{bmatrix} \sigma_l^i & 0 & 0 \\ 0 & \sigma_t^i & 0 \\ 0 & 0 & \sigma_n^i \end{bmatrix} D^T, \quad (2.2.1)$$

$$\mathbf{M}_e(x) = \frac{1}{C_m \beta} D \begin{bmatrix} \sigma_l^e & 0 & 0 \\ 0 & \sigma_t^e & 0 \\ 0 & 0 & \sigma_n^e \end{bmatrix} D^T, \quad (2.2.2)$$

$$I'_{\text{Ion}} = \frac{I_{\text{Ion}}}{C_m}, \quad (2.2.3)$$

$$I'_{\text{App}} = \frac{I_{\text{app}}}{C_m}. \quad (2.2.4)$$

The conductive properties of heart are strongly Anisotropic, so $\mathbf{M}_i(x)$ and $\mathbf{M}_e(x)$, depend on the direction of the fibers of the heart muscle, $\sigma_l, \sigma_t, \sigma_n$ are the conductivities along the fibers, tangent to the fibers and normal respectively. For example in Figure there are the fiber directions generated with the software from [21].



FIGURE 2.2: Example of Generated Fiber Directions.

The matrix D has the unit vectors of the fiber directions as columns. With The ionic current per unit cell membrane is I_{ion} , and is given by a heart cell electrophysiology model, e.g. : FitzHugh-Nagumo Model, Hodgkin-Huxley, Fenton-Karma. The variables given by the ionic model are s (voltage, and gating channels). The external stimulus is I_{app} . The time is described by t . If we couple the equations of the Bidomain model from the heart with the thorax we get the following system:

$$\begin{aligned}
 \partial_t w &= H(v, w) & x \in \Omega_H, \\
 \nabla \cdot (\mathbf{M}_i(x) \nabla v) + \nabla \cdot (\mathbf{M}_i(x) \nabla u_e) &= \partial_t v + I'_{ion}(v, w) & x \in \Omega_H, \\
 \nabla \cdot (\mathbf{M}_i(x) \nabla v) + \nabla \cdot ((\mathbf{M}_i(x) + \mathbf{M}_e(x)) \nabla u_e) &= 0 & x \in \Omega_H, \\
 (\mathbf{M}_i(x) \nabla v + \mathbf{M}_e(x) \nabla u_e) \cdot \mathbf{n} &= 0 & x \in \partial\Omega_H, \\
 (\mathbf{M}_i(x) \nabla v) \cdot \mathbf{n} + ((\mathbf{M}_i(x) + \mathbf{M}_e(x)) \nabla u_e) \cdot \mathbf{n} &= (\mathbf{M}_T(x) \nabla u_t) \cdot \mathbf{n} & x \in \partial\Omega_H, \\
 u_e &= u_t & x \in \partial\Omega_H, \\
 \nabla(\mathbf{M}_T(x) \nabla u_t) &= 0 & x \in \Omega_T, \\
 (\mathbf{M}_T(x) \nabla u_t) \cdot \mathbf{n} &= 0 & x \in \partial\Omega_T.
 \end{aligned} \tag{2.2.5}$$

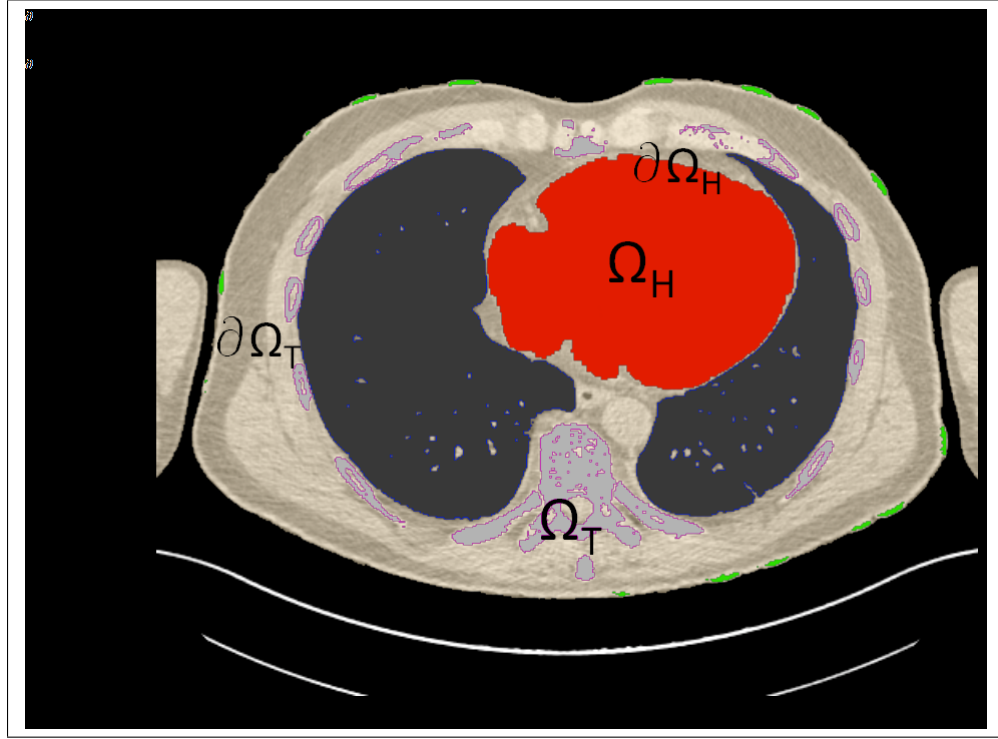


FIGURE 2.3: Heart Thorax domain.

If we use the operator splitting technique we can separate the following equation

$$\nabla \cdot (\mathbf{M}_i(x)\nabla v) + \nabla \cdot (\mathbf{M}_i(x)\nabla u_e) = \partial_t v + I'_{ion}(v, w) \quad x \in \Omega_H, \quad (2.2.6)$$

into

$$\nabla \cdot (\mathbf{M}_i(x)\nabla v) + \nabla \cdot (\mathbf{M}_i(x)\nabla u_e) = \partial_t v, \quad (2.2.7)$$

$$-I'_{ion}(v, w) - I'_{app} = \partial_t v. \quad (2.2.8)$$

The operator splitting technique works in the following way. For an equation

$$\frac{dq}{dt} = A(q) + B(q), \quad (2.2.9)$$

we can decompose the solution in two steps, this is known as Godunov splitting

$$q^{n*} = q^n + \Delta t A(q^n), \quad (2.2.10)$$

$$q^{n+1} = q^{n*} + \Delta t B(q^{n*}). \quad (2.2.11)$$

$$(2.2.12)$$

Another Operator splitting technique is Strang;

$$q^{n*} = q^n + \frac{\Delta t}{2} A(q^n), \quad (2.2.13)$$

$$q^{n**} = q^{n*} + \Delta t B(q^{n*}). \quad (2.2.14)$$

$$q^{n+1} = q^{n**} + \frac{\Delta t}{2} A(q^{n**}). \quad (2.2.15)$$

$$(2.2.16)$$

Equation 2.2.7 can be solved applying the θ rule.

$$\frac{v^{n+1} - v^n}{\Delta t} = \theta [\nabla(\mathbf{M}_i(x)\nabla v^{n+1})] + (1 - \theta) [\nabla(\mathbf{M}_i(x)\nabla v^n)] + \nabla(\mathbf{M}_i(x)\nabla u_e), \quad (2.2.17)$$

$$v^{n+1} - v^n = \theta \Delta t [\nabla(\mathbf{M}_i(x)\nabla v^{n+1})] + (1 - \theta) \Delta t [\nabla(\mathbf{M}_i(x)\nabla v^n)] + \Delta t \nabla(\mathbf{M}_i(x)\nabla u_e). \quad (2.2.18)$$

If we multiply by a test function ψ and integrating we have:

$$\begin{aligned} \int_{\Omega_H} v^{n+1} \psi - \int_{\Omega_H} v^n \psi &= \theta \Delta t \left[- \int_{\Omega_H} \mathbf{M}_i(x) \nabla v^{n+1} \nabla \psi + \int_{\partial\Omega_H} \psi \mathbf{M}_i(x) \nabla v^{n+1} \cdot \mathbf{n} \right] + \\ &\quad (1 - \theta) \Delta t \left[- \int_{\Omega_H} \mathbf{M}_i(x) \nabla v^n \nabla \psi + \int_{\partial\Omega_H} \psi \mathbf{M}_i(x) \nabla v^n \cdot \mathbf{n} \right] + \\ &\quad \Delta t \left[- \int_{\Omega_H} \mathbf{M}_i(x) \nabla u_e \nabla \psi + \int_{\partial\Omega_H} \psi \mathbf{M}_i(x) \nabla u_e \cdot \mathbf{n} \right]. \end{aligned} \quad (2.2.19)$$

From the following equation we get the boundary conditions:

$$(\mathbf{M}_i(x) \nabla v) \cdot \mathbf{n} + (\mathbf{M}_i(x) \nabla u_e) \cdot \mathbf{n} = 0, \quad (2.2.20)$$

$$\theta \int_{\partial\Omega_H} \psi \mathbf{M}_i(x) \nabla v^{n+1} \cdot \mathbf{n} + (1 - \theta) \int_{\partial\Omega_H} \psi \mathbf{M}_i(x) \nabla v^n \cdot \mathbf{n} + \int_{\partial\Omega_H} \psi \mathbf{M}_i(x) \nabla u_e \cdot \mathbf{n} = 0. \quad (2.2.21)$$

Applying the boundary conditions in Equation 2.2.19 we get:

$$\begin{aligned} \int_{\Omega_H} v^{n+1} \psi - \int_{\Omega_H} v^n \psi &= -\theta \Delta t \int_{\Omega_H} \mathbf{M}_i(x) \nabla v^{n+1} \nabla \psi \\ &\quad - (1 - \theta) \Delta t \int_{\Omega_H} \mathbf{M}_i(x) \nabla v^n \nabla \psi + -\Delta t \int_{\Omega_H} \mathbf{M}_i(x) \nabla u_e \nabla \psi. \end{aligned} \quad (2.2.22)$$

We make the same procedure for the following equation, and multiply it by Δt

$$\nabla \cdot (\mathbf{M}_i(x) \nabla v) + \nabla \cdot ((\mathbf{M}_i(x) + \mathbf{M}_e(x)) \nabla u_e) = 0, \quad (2.2.23)$$

$$\begin{aligned}
& \theta \Delta t \left[- \int_{\Omega_H} \mathbf{M}_i(x) \nabla v^{n+1} \nabla \psi + \int_{\partial \Omega_H} \psi \mathbf{M}_i(x) \nabla v^{n+1} \cdot \mathbf{n} \right] + \\
& (1 - \theta) \Delta t \left[- \int_{\Omega_H} \mathbf{M}_i(x) \nabla v^n \nabla \psi + \int_{\partial \Omega_H} \psi \mathbf{M}_i(x) \nabla v^n \cdot \mathbf{n} \right] + \\
& \Delta t \left[- \int_{\Omega_H} (\mathbf{M}_i(x) + \mathbf{M}_e(x)) \nabla u_e \nabla \psi + \int_{\partial \Omega_H} (\mathbf{M}_i(x) + \mathbf{M}_e(x)) \nabla u_e \psi \cdot \mathbf{n} \right] = 0.
\end{aligned} \tag{2.2.24}$$

or

$$\begin{aligned}
& \theta \Delta t \left[- \int_{\Omega_H} \mathbf{M}_i(x) \nabla v^{n+1} \nabla \psi + \int_{\partial \Omega_H} \psi \mathbf{M}_i(x) \nabla v^{n+1} \cdot \mathbf{n} \right] + \\
& (1 - \theta) \Delta t \left[- \int_{\Omega_H} \mathbf{M}_i(x) \nabla v^n \nabla \psi + \int_{\partial \Omega_H} \psi \mathbf{M}_i(x) \nabla v^n \cdot \mathbf{n} \right] + \\
& \Delta t \left[- \int_{\Omega_H} (\mathbf{M}_i(x) + \mathbf{M}_e(x)) \nabla u_e \nabla \psi + \int_{\partial \Omega_H} (\mathbf{M}_i(x)) \nabla u_e \psi \cdot \mathbf{n} + \int_{\partial \Omega_H} (\mathbf{M}_e(x)) \nabla u_e \psi \cdot \mathbf{n} \right] = 0.
\end{aligned} \tag{2.2.25}$$

Applying boundary conditions to Equation 2.2.25 we get

$$\begin{aligned}
& -\theta \Delta t \int_{\Omega_H} \mathbf{M}_i(x) \nabla v^{n+1} \nabla \psi \\
& -(1 - \theta) \Delta t \int_{\Omega_H} \mathbf{M}_i(x) \nabla v^n \nabla \psi \\
& -\Delta t \int_{\Omega_H} (\mathbf{M}_i(x) + \mathbf{M}_e(x)) \nabla u_e \nabla \psi + \int_{\partial \Omega_H} (\mathbf{M}_e(x)) \nabla u_e \psi \cdot \mathbf{n} = 0.
\end{aligned} \tag{2.2.26}$$

To calculate the remaining normal component from Equation 2.2.26, we use the thorax' equations. We use the following equation

$$\nabla(\mathbf{M}_T(x) \nabla u_T) = 0, \tag{2.2.27}$$

$$\int_{\Omega_T} \nabla(\mathbf{M}_T(x) \nabla u_T) \psi = - \int_{\Omega_T} \mathbf{M}_T(x) \nabla u_T \nabla \psi + \int_{\partial \Omega_T} \mathbf{M}_T(x) \nabla u_T \psi \cdot \mathbf{n}. \tag{2.2.28}$$

Applying the conditions of continuation of flux we can write the equation in the following way

$$\int_{\partial \Omega_H} \mathbf{M}_e(x) \nabla u_e \cdot \mathbf{n} \psi = \int_{\Omega_T} \mathbf{M}_T(x) \nabla u_T \nabla \psi. \tag{2.2.29}$$

The whole system of the heart's equations can be written in the following system for the operator splitting technique

$$\begin{aligned}
& \int_{\Omega_H} v^{n+1} \psi - \int_{\Omega_H} v^n \psi = -\theta \Delta t \int_{\Omega_H} \mathbf{M}_i(x) \nabla v^{n+1} \nabla \psi \\
& -(1 - \theta) \Delta t \int_{\Omega_H} \mathbf{M}_i(x) \nabla v^n \nabla \psi + -\Delta t \int_{\Omega_H} \mathbf{M}_i(x) \nabla u_e \nabla \psi, \\
& \theta \Delta t \int_{\Omega_H} \mathbf{M}_i(x) \nabla v^{n+1} \nabla \psi + (1 - \theta) \Delta t \int_{\Omega_H} \mathbf{M}_i(x) \nabla v^n \nabla \psi \\
& + \Delta t \int_{\Omega_H} (\mathbf{M}_i(x) + \mathbf{M}_e(x)) \nabla u_e \nabla \psi + \Delta t \int_{\Omega_T} \mathbf{M}_T(x) \nabla u_T \nabla \psi = 0.
\end{aligned} \tag{2.2.30}$$

If we consider

$$v^{n+1} = \sum_{j=1}^N v_j^{n+1} \phi_j \quad (2.2.31)$$

$$v^n = \sum_{j=1}^N v_j^n \phi_j \quad (2.2.32)$$

$$u = \begin{cases} u_e, & \text{if } x \in \Omega_H \\ u_t, & \text{if } x \in \Omega_T \end{cases} = \sum_{j=1}^M u_j \phi_j \quad (2.2.33)$$

$$(2.2.34)$$

then

$$\begin{aligned} \sum_{j=1}^N v_j^{n+1} \left(\int_{\Omega_H} \phi_j \phi_i + \theta \Delta t \int_{\Omega_H} \mathbf{M}_i(x) \nabla \phi_j \nabla \phi_i \right) + \sum_{j=1}^M u_j (\Delta t \int_{\Omega_H} \mathbf{M}_i(x) \nabla \phi_j \nabla \phi_i) = \\ \sum_{j=1}^N v_j^n \left(\int_{\Omega_H} \phi_j \phi_i - (1 - \theta) \Delta t \int_{\Omega_H} \mathbf{M}_i(x) \nabla \phi_j \nabla \phi_i \right), \\ \sum_{j=1}^N v_j^{n+1} \theta \Delta t \int_{\Omega_H} \mathbf{M}_i(x) \nabla \phi_j \nabla \phi_i + \sum_{j=1}^N v_j^n (1 - \theta) \Delta t \int_{\Omega_H} \mathbf{M}_i(x) \nabla \phi_j \nabla \phi_i \\ + \sum_{j=1}^M u_j (\Delta t \int_{\Omega_H} (\mathbf{M}_i(x) + \mathbf{M}_e(x)) \nabla \phi_j \nabla \phi_i + \Delta t \int_{\Omega_T} \mathbf{M}_T(x) \nabla \phi_j \nabla \phi_i) = 0. \end{aligned} \quad (2.2.35)$$

2.3 Monodomain

We can make a simplification if we consider that $\mathbf{M}_i(x) = \lambda \mathbf{M}_e(x)$. Considering the following equation

$$\nabla \cdot (\mathbf{M}_i(x) \nabla v) + \nabla \cdot ((\mathbf{M}_i(x) + \mathbf{M}_e(x)) \nabla u_e) = 0, \quad (2.3.1)$$

we can modify it to

$$\nabla \cdot (\mathbf{M}_i(x) \nabla v) + \nabla \cdot ((1 + \lambda) \mathbf{M}_i(x) \nabla u_e) = 0, \quad (2.3.2)$$

or

$$\begin{aligned} \nabla \cdot (\mathbf{M}_i(x) \nabla v) + (1 + \lambda) \nabla \cdot (\mathbf{M}_i(x) \nabla u_e) = 0, \\ \nabla (\mathbf{M}_i(x) \nabla u_e) = -\frac{1}{1 + \lambda} \nabla \cdot (\mathbf{M}_i(x) \nabla v). \end{aligned} \quad (2.3.3)$$

Using this substitution we modify the Equation 2.2.7

$$\nabla \cdot (\mathbf{M}_i(x) \nabla v) - \frac{1}{1+\lambda} \nabla \cdot (\mathbf{M}_i(x) \nabla v) = \partial_t v, \quad (2.3.4)$$

giving

$$\frac{\lambda}{1+\lambda} \nabla \cdot (\mathbf{M}_i(x) \nabla v) = \partial_t v. \quad (2.3.5)$$

Using the θ rule and multiplying by a test function ψ we can express it in the following form:

$$\begin{aligned} & \theta \frac{\lambda}{1+\lambda} \Delta t \left[- \int_{\Omega_H} \mathbf{M}_i(x) \nabla v^{n+1} \nabla \psi + \int_{\Omega_H} \mathbf{M}_i(x) \nabla v^{n+1} n \psi \right] + \\ (1-\theta) & \frac{\lambda}{1+\lambda} \Delta t \left[- \int_{\Omega_H} \mathbf{M}_i(x) \nabla v^n \nabla \psi + \int_{\Omega_H} \mathbf{M}_i(x) \nabla v^n n \psi \right] = \int_{\Omega_H} v^{n+1} \psi - \int_{\Omega_H} v^n \psi. \end{aligned} \quad (2.3.6)$$

We will consider

$$\int_{\Omega_H} \mathbf{M}_i(x) \nabla v^{n+1} n \psi + \int_{\Omega_H} \mathbf{M}_i(x) \nabla v^n n \psi = 0, \quad (2.3.7)$$

and to solve for u_e we will use Equation 2.2.35, as mentioned in [22]. The monodomain equation will be the following

$$\begin{aligned} & \int_{\Omega_H} v^{n+1} \psi + \theta \frac{\lambda}{1+\lambda} \Delta t \left[\int_{\Omega_H} \mathbf{M}_i(x) \nabla v^{n+1} \nabla \psi \right] \\ = & \int_{\Omega_H} v^n \psi - (1-\theta) \frac{\lambda}{1+\lambda} \Delta t \left[\int_{\Omega_H} \mathbf{M}_i(x) \nabla v^n \nabla \psi \right] \end{aligned} \quad (2.3.8)$$

or

$$\begin{aligned} & \sum_{j=1}^N v_j^{n+1} \left(\int_{\Omega_H} \phi_j \phi_i + \theta \frac{\lambda}{1+\lambda} \Delta t \left[\int_{\Omega_H} \mathbf{M}_i(x) \nabla \phi_j \nabla \phi_i \right] \right) \\ = & \sum_{j=1}^N v_j^n \left(\int_{\Omega_H} \phi_j \phi_i - (1-\theta) \frac{\lambda}{1+\lambda} \Delta t \left[\int_{\Omega_H} \mathbf{M}_i(x) \nabla \phi_j \nabla \phi_i \right] \right). \end{aligned} \quad (2.3.9)$$

2.4 Considering an isolated heart

We consider there is no flux leaving the heart, thus, we can simplify the system. From

the heart equations coupled with the thorax we make the following modification

$$\begin{aligned}
\partial_t w &= H(v, w) & x \in \Omega_H, \\
\nabla \cdot (\mathbf{M}_i(x) \nabla v) + \nabla \cdot (\mathbf{M}_i(x) \nabla u_e) &= \partial_t v + I'_{ion}(v, w) & x \in \Omega_H, \\
\nabla \cdot (\mathbf{M}_i(x) \nabla v) + \nabla \cdot ((\mathbf{M}_i(x) + \mathbf{M}_e(x)) \nabla u_e) &= 0 & x \in \Omega_H, \\
(\mathbf{M}_i(x) \nabla v + \mathbf{M}_i(x) \nabla u_e) \cdot \mathbf{n} &= 0 & x \in \partial\Omega_H, \\
(\mathbf{M}_i(x) \nabla v) \mathbf{n} + ((\mathbf{M}_i(x) + \mathbf{M}_e(x)) \nabla u_e) \cdot \mathbf{n} &= 0 & x \in \partial\Omega_H, \\
u_e &= u_t & x \in \partial\Omega_H, \\
\nabla(\mathbf{M}_T(x) \nabla u_t) &= 0 & x \in \Omega_T, \\
(\mathbf{M}_T(x) \nabla u_t) \cdot \mathbf{n} &= 0 & x \in \partial\Omega_T.
\end{aligned} \tag{2.4.1}$$

This modification will modify the equation 2.2.26 in

$$\begin{aligned}
-\theta \Delta t \int_{\Omega_H} \mathbf{M}_i(x) \nabla v^{n+1} \nabla \psi - (1 - \theta) \Delta t \int_{\Omega_H} \mathbf{M}_i(x) \nabla v^n \nabla \psi \\
- \Delta t \int_{\Omega_H} (\mathbf{M}_i(x) + \mathbf{M}_e(x)) \nabla u_e \nabla \psi = 0,
\end{aligned} \tag{2.4.2}$$

or using the decomposition

$$u_e = \sum_{j=1}^N u_j \phi_j, \tag{2.4.3}$$

$$\begin{aligned}
\sum_{j=1}^N v^{n+1} \theta \Delta t \int_H \mathbf{M}_i(x) \nabla \phi_j \nabla \phi_i + \sum_{j=1}^N v^n (1 - \theta) \Delta t \int_H \mathbf{M}_i(x) \nabla \phi_j \nabla \phi_i \\
+ \sum_{j=1}^N u_j \Delta t \int_H (\mathbf{M}_i(x) + \mathbf{M}_e(x)) \nabla \phi_j \nabla \phi_i = 0.
\end{aligned} \tag{2.4.4}$$

2.5 Mathematical ionic models in Cardiac Electrophysiology

2.5.0.1 Hodgkin Huxley (1952)

The equation

$$\partial_t w = H(v, w) \tag{2.5.1}$$

is a system of ODEs that describes electrochemical reactions in the cells. In a cell the membrane potential v is given by

$$c_m \frac{dv}{dt} = -I_{ion}(v, w) + I_{app}. \tag{2.5.2}$$

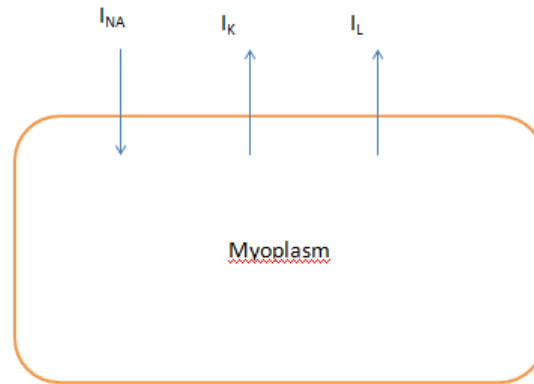


FIGURE 2.4: The ionic currents in the Hodgkin-Huxley model.

where I_{ion} is the ionic current, I_{app} is an applied stimulus current, which triggers the action potential of the cell. The current I_{ion} depends of v , and w which are the gating variables. In the following section we explain different models, and show their simulation in one cell.

The membrane potential in a cell is given by the difference between the extracellular and intracellular potential. This potential is given by the addition of the gated currents, leakage current and the impedance given by the capacitance of the cell membrane. When the addition of the currents in the cell exceeds a certain threshold, then an action potential is created. To model this behavior of the cell, we use a series of gated currents that are related to the concentration of ions in the membrane. For example in the Hodgkin-Huxley model, there is a current related to the concentration of sodium (I_{NA}), and potassium (I_K). The leakage current (I_L) is the voltage (only) dependent current, while the other two depend on the ionic concentrations.

This model was generated by considering the cable equation. From the cable equation the authors derived the model using a squid axon making a current clamp and a space clamp. They introduced a highly conductive cable into the axon, and were able to measure and separate the currents of the cell membrane. This model is the basis for many electrophysiological models.

The currents are controlled by a series of gates. The probability of a channel to be open is given by m , h and n . The combined action of m and h controls the sodium channels. The potassium gates are controlled by n .

Component Membrane

$$I_{app} = \begin{cases} 20, & \text{if } 10 \geq t \leq 10.5 \\ 0, & \text{otherwise} \end{cases}, \quad (2.5.3)$$

$$\frac{dV}{dt} = \frac{-((-I_{app}) + I_{Na} + I_K + I_L)}{C_m} \quad (2.5.4)$$

$$C_m = 1. \quad (2.5.5)$$

Sodium channel m gate

$$\alpha_m = \frac{-0.10 * (V + 50)}{e^{\frac{-(V+50)}{10}} - 1} \quad (2.5.6)$$

$$\beta_m = 4e^{\frac{-(V+75)}{18}} \quad (2.5.7)$$

$$\frac{dm}{dt} = \alpha_m(1 - m) - (\beta_m m). \quad (2.5.8)$$

Sodium channel h gate

$$\alpha_h = 0.07e^{\frac{-(V+75)}{20}} \quad (2.5.9)$$

$$\beta_h = \frac{1}{e^{\frac{-(V+45)}{10}} + 1} \quad (2.5.10)$$

$$\frac{dh}{dt} = \alpha_h(1 - h) - (\beta_h h). \quad (2.5.11)$$

Potassium channel n gate

$$\alpha_n = \frac{-0.01 * (V + 65)}{e^{\frac{-(V+65)}{10}} - 1} \quad (2.5.12)$$

$$\beta_n = 0.125e^{\frac{V+75}{80}} \quad (2.5.13)$$

$$\frac{dn}{dt} = \alpha_n(1 - n) - (\beta_n n). \quad (2.5.14)$$

Currents

$$I_L = 0.3(V - (-75 + 10.613)) \quad (2.5.15)$$

$$I_K = 36n^4(V - (-75 - 12)) \quad (2.5.16)$$

$$I_{Na} = 120m^3h(V - (-75 + 115)) \quad (2.5.17)$$

Initialization

$$V_0 = -75 \quad (2.5.18)$$

$$m_0 = 0.05 \quad (2.5.19)$$

$$h_0 = 0.6 \quad (2.5.20)$$

$$n_0 = 0.325 \quad (2.5.21)$$

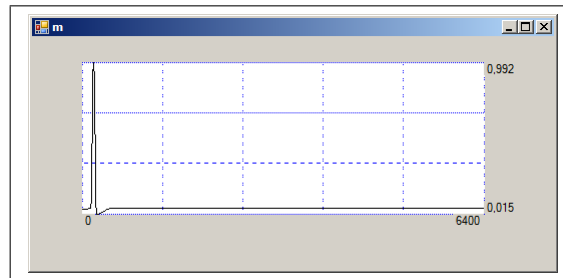


FIGURE 2.5: m gate.

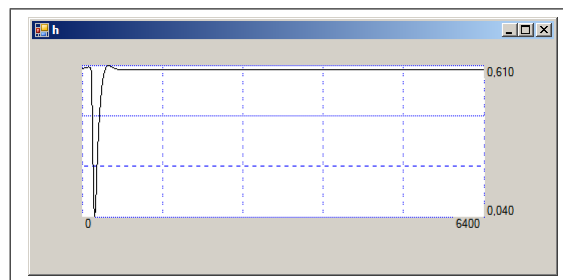


FIGURE 2.6: h gate .

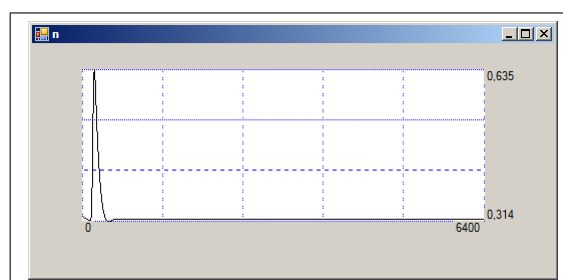
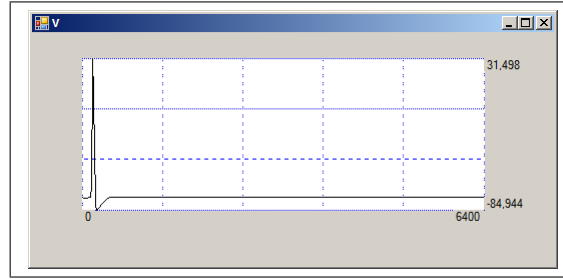


FIGURE 2.7: n gate.

FIGURE 2.8: V_m (mV).

2.5.0.2 FitzHugh-Nagumo (1961)

The FitzHugh-Nagumo is a simplified version of the Hodgkin-Huxley model explained in the above section. This model is a modification of the Van der Pol oscillator to obtain a model that generates a similar shape of the action potential, but using only two variables. The membrane potential V , and the recovery variable r .

The membrane potential V is given by

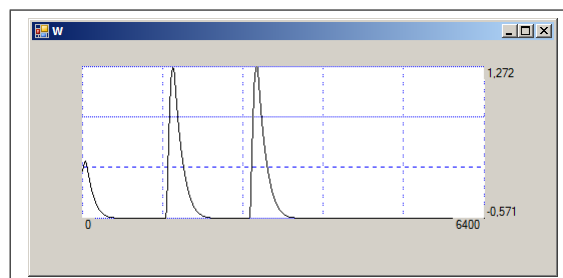
$$\frac{dV}{dt} = V - V^3/3 - r + I_{app}. \quad (2.5.22)$$

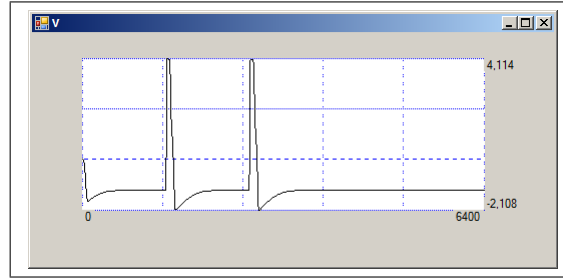
The recovery variable r

$$\frac{dr}{dt} = 0.08(V + 0.7 - 0.8 * r). \quad (2.5.23)$$

Applied Stimulus I

$$I_{app} = \begin{cases} 20, & \text{if } 100 \geq t \leq 105 \\ 20, & \text{if } 200 \geq t \leq 205 \\ 0, & \text{otherwise} \end{cases} \quad (2.5.24)$$

FIGURE 2.9: r recovery variable.

FIGURE 2.10: V_m (mV).

2.5.0.3 Fenton Karma 1998

The Fenton Karma model (1998) is a simplified model to explain the physioelectrical activity of the heart's cell. The model consists of three membrane currents I_{fi} , I_{so} , I_{si} that approximate well the restitution properties and spiral wave behavior of more complex ionic models of cardiac action potentials [23]. The suffix f means fast, s slow and i, o input, output. I_{fi} corresponds to the sodium current. I_{so} corresponds to the potassium current. I_{si} corresponds to the calcium current.

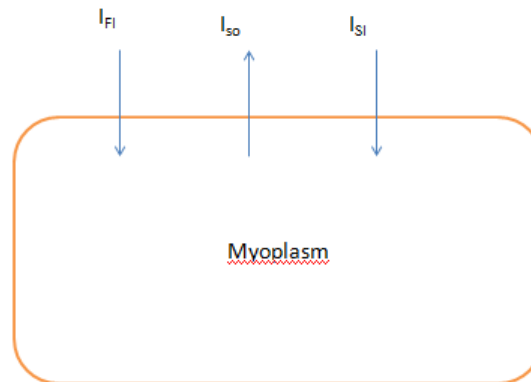


FIGURE 2.11: The ionic currents in the Fenton Karma model.

The potential is normalized in the variable u , then algebraically scaled to the potential values for the membrane potential v .

Component Membrane

$$\frac{d}{dt}(u) = -(I_{fi} + I_{so} + I_{si} + I_{app}) \quad (2.5.25)$$

$$v = V_0 + u(V_{fi} - V_0) \quad (2.5.26)$$

Component p

$$p = \begin{cases} 1, & \text{if } u < u_c \\ 0, & \text{otherwise} \end{cases} \quad (2.5.27)$$

Component q

$$q = \begin{cases} 1, & \text{if } u < u_v \\ 0, & \text{otherwise} \end{cases} \quad (2.5.28)$$

Component I_{fi}

$$\tau_d = \frac{C_m}{g_{fi}^{Max}} \quad (2.5.29)$$

$$I_{fi} = \frac{(-u_2)p(1-u)(u-u_c)}{\tau_d} \quad (2.5.30)$$

Component u_2 gate

$$\tau_v^- = q\tau_{v1}^- + (1-q)\tau_{v2}^- \quad (2.5.31)$$

$$\frac{du_2}{dt} = \frac{(1-p)(1-u_2)}{\tau_v^-} - \frac{pu_2}{\tau_v^+} \quad (2.5.32)$$

Component: I_{so}

$$I_{so} = \frac{u(1-p)}{\tau_0} - \frac{p}{\tau_r} \quad (2.5.33)$$

Component: I_{si}

$$I_{si} = \frac{(-u_3)(1 + \tanh(k(u - u_{csi})))}{2\tau_{si}} \quad (2.5.34)$$

Component u_3 gate

$$\frac{du_3}{dt} = \frac{(1-p)(1-u_3)}{\tau_w^-} - \frac{pu_3}{\tau_w^+} \quad (2.5.35)$$

In the following Figures we can see a plot of the variables I_{app} , u , v , u_3 , and u_2 .

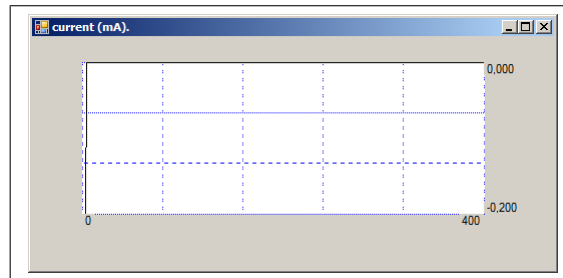
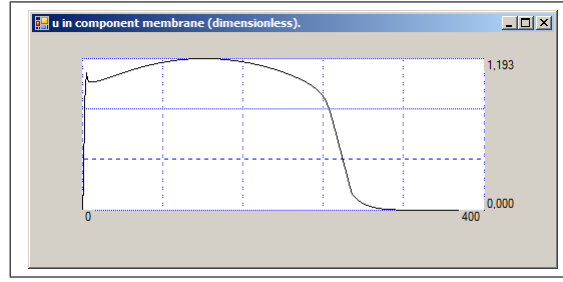
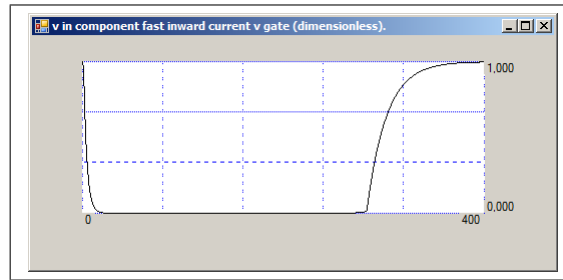
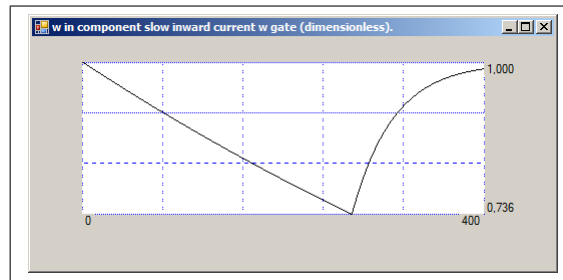
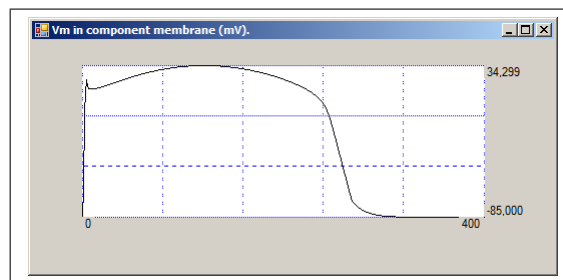


FIGURE 2.12: I_{app} current in mA.

FIGURE 2.13: u in component membrane (dimensionless).FIGURE 2.14: u_2 in component fast inward current v gate (dimensionless).FIGURE 2.15: u_3 in component slow inward current w gate (dimensionless).FIGURE 2.16: v in component membrane (mV).

Initialization The initialization values will be the following: $c_m = 1$, $V_0 = -85$, $V_{fi} = 15$, $u_c = 0.13$, $u_v = 0.04$, $g_{fi}^{Max} = 4$, $\tau_{v1}^- = 1250$, $\tau_{v2}^- = 19.6$, $\tau_v^+ = 3.33$, $\tau_0 = 12.5$, $\tau_r = 33.33$, $\tau_{si} = 29$, $u_{csi} = 0.85$, $k = 10$, $\tau_w^- = 41$, $\tau_w^+ = 870$.

2.6 Example of Forward Simulation

We apply a stimulus in the heart's surface and we calculate the thorax potential (ECG) from the simulation. In the following image we can appreciate the transmembrane potential in the heart's surface.

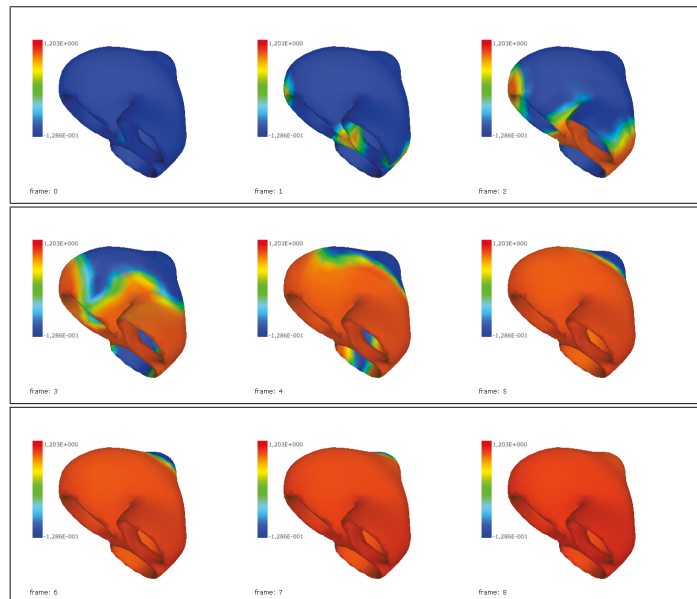


FIGURE 2.17: Voltage Distribution for the transmembrane potential.

This transmembrane potential produces the following extracellular potential.

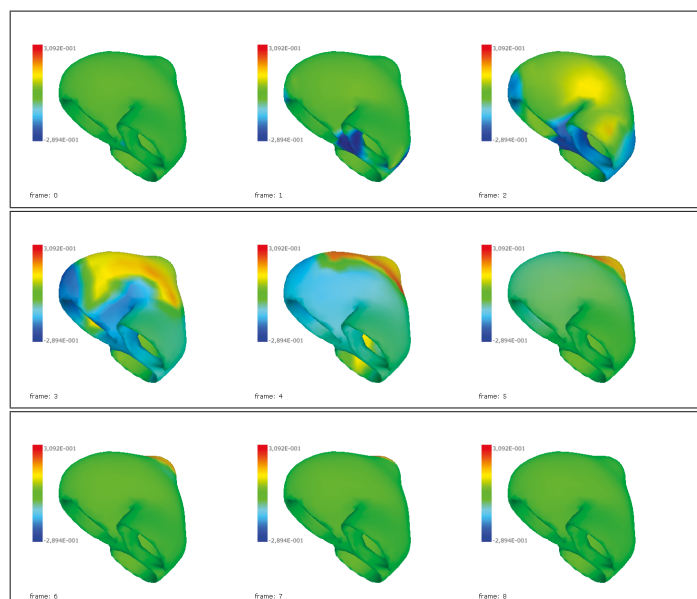


FIGURE 2.18: Voltage Distribution for the extracellular potential.

The extracellular potential creates the next distribution in the thorax.

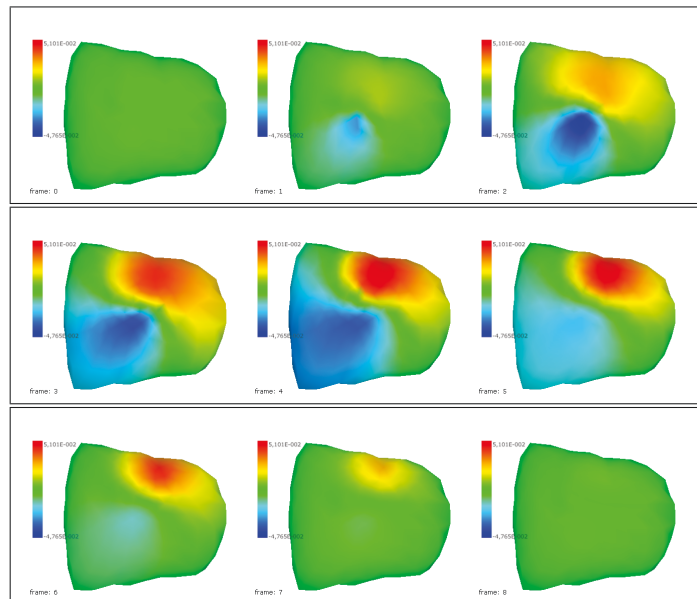


FIGURE 2.19: Voltage Distribution calculated in the thorax.

From the thorax potentials we calculate the VL,VR,VF,V1,V2,V3,V4,V5 and V6 (the standard 12 ECG derivations).

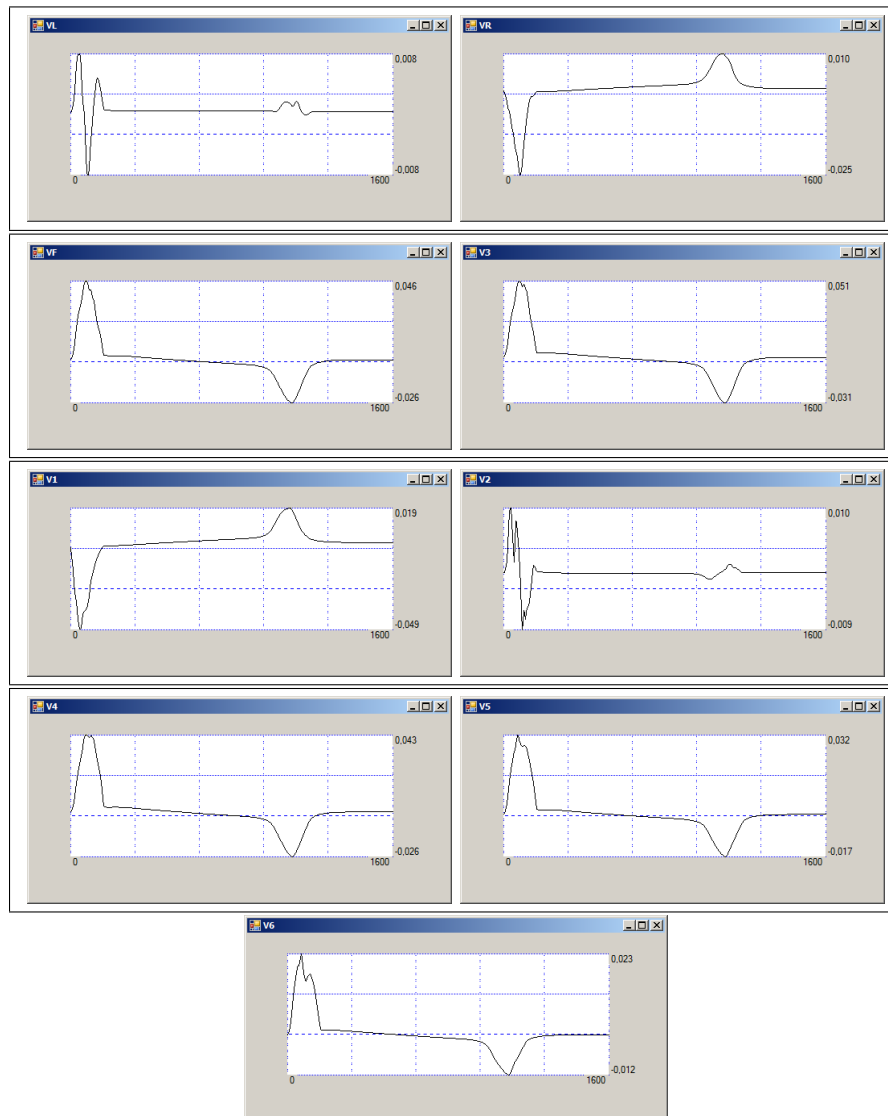


FIGURE 2.20: Standard 12 lead derivation measured in the thorax.

Chapter 3

Chapter 3. Inverse Dynamic Heart Torso Equations

In the last chapter we explained the forward problem in computational electrocardiography. We explained the Bidomain model and the necessary considerations to simulate the propagation of potential from the heart to the thorax. In this chapter we will explain the inverse problem using the constraints from the Monodomain model, as a simplification of the Bidomain. Using these constraints we are able to create solutions that depend not only of the potential on the thorax surface, but also the potential from the last calculation.

3.1 The inverse problem

By an inverse problem we mean the problem of parameter identification, that means we try to determine some of the unknown values of the model parameters according to measurements in a real site and results obtained by simulations. In our inverse problem we wish to minimize the applied current in the ionic part of the bidomain model to reconstruct the transmembrane potential, and the potential sources. In comparison the quasistatic Poisson's equation,

$$\begin{aligned} -\nabla \cdot (\mathbf{M}_s \nabla u_s) &= 0 \in \Omega_B, \\ (\mathbf{M}_s \nabla u_s) \cdot \mathbf{n} &= 0 \text{ on } \partial\Omega_B - \partial\Omega_H, \\ u_s &= u_e \text{ on } \partial\Omega_H. \end{aligned} \tag{3.1.1}$$

whereas we will minimize the extracellular potential in the surface of the heart in our study we will minimize the applied current (stimulus) in the bidomain model. Essentially,

we want to minimize the cost functional:

$$\begin{aligned} J(I_i, I_e) &= \iint_{\Sigma_{B,T}} |u_s(t, y, I_s) - u_c(t, y)|^2 dy dt \\ &+ \frac{\varepsilon}{2} \iint_{\Omega_{H,T}} |I_s(t, x)|^2 dx dt, \text{ where } I_s = I_i - I_e. \end{aligned} \quad (3.1.2)$$

Here we have the existence result of optimal solution of (3.1.2):

Lemma 3.1.1. Given $v_0 \in L^2(\Omega)$ and $I_{i,e} \in L^2(\Omega_{H,T})$, there exists a solution I_s^* of the optimal control problem (3.1.2).

The Lagrangian related to the optimal control problem is given by

$$\begin{aligned} \mathcal{L}(\theta) &= \frac{\varepsilon}{2} \iint_{\Omega_{H,T}} |I_i - I_e|^2 dx dt + \int_{\Sigma_{B,T}} |u_s(t, y, I_s) - u_c(t, y)|^2 dy dt \\ &+ \iint_{\Omega_{H,T}} \left(\beta c_m \partial_t v - \nabla \cdot (\mathbf{M}_i(x) \nabla u_i) + \beta I_{\text{ion}}(v, w) - I_i \right) p_i dx dt \\ &- \iint_{\Omega_{H,T}} \left(\beta c_m \partial_t v + \nabla \cdot (\mathbf{M}_e(x) \nabla u_e) + \beta I_{\text{ion}}(v, w) - I_e \right) p_e dx dt \\ &- \int_{\Omega_{B,T}} \nabla \cdot (\mathbf{M}_s(x) \nabla u_s) p_s dx dt + \iint_{\Omega_{H,T}} \left(\partial_t w - H(v, w) \right) q dx dt + \\ &\int_{\Omega_H} \left(v(x, 0) - v_0(x) \right) z_1 dx \\ &+ \iint_{\Sigma_{H,T}} \left(u_e - u_s \right) z_2 dy dt + \iint_{\Sigma_{H,T}} \left(\mathbf{M}_e(y) \nabla u_e - \mathbf{M}_s(y) \nabla u_s \right) \cdot \eta z_3 dy dt, \end{aligned} \quad (3.1.3)$$

where $\theta = (u_i, u_e, u_s, w, I_i, I_e, p_i, p_e, p_s, q, z_1, z_2, z_3)$. Observe that from (3.1.3) we get

$$\left(\frac{\partial \mathcal{L}(u_i, u_e, u_s, w, I_i, I_e, p_i, p_e, p_s, q, z_1, z_2, z_3)}{\partial I_i}, \delta I_i \right) = ((I_i - I_e) - p_i, \delta I_i),$$

and

$$\left(\frac{\partial \mathcal{L}(u_i, u_e, u_s, w, I_i, I_e, p_i, p_e, p_s, q, z_1, z_2, z_3)}{\partial I_e}, \delta I_e \right) = (-(I_i - I_e) - p_e, \delta I_e).$$

The first order optimality system is given by the Karush-Kuhn-Tucker (KKT) conditions which result from equating the partial derivatives of \mathcal{L} with respect to u_i, u_e, u_s

and w to zero

$$\begin{aligned}
& -\beta c_m \partial_t (p_i - p_e) - \nabla \cdot (\mathbf{M}_i(x) \nabla p_i) + \\
& \quad \beta I_{\text{ion}v}(v, w)(p_i - p_e) - H_v(v, w)q = 0, \quad (x, t) \in \Omega_{B,T}, \\
& -\beta c_m \partial_t (p_i - p_e) + \nabla \cdot (\mathbf{M}_e(x) \nabla p_e) + \\
& \quad \beta I_{\text{ion}v}(v, w)(p_i - p_e) + H_v(v, w)q = 0, \quad (x, t) \in \Omega_{B,T}, \\
& -\nabla \cdot (\mathbf{M}_s(x) \nabla p_s) = 0, \quad (x, t) \in \Omega_{B,T}, \\
& -\partial_t q + \beta I_{\text{ion}w}(v, w)(p_i - p_e) - H_w(v, w)q = 0, \quad (x, t) \in \Omega_{H,T}.
\end{aligned} \tag{3.1.4}$$

Herein $I_{\text{ion}v}$, $I_{\text{ion}w}$, H_v and H_w are the derivative of I_{ion} and H with respect to v , w . We complete the system (3.1.4) with terminal conditions and boundary conditions:

$$\begin{aligned}
& p_i(\cdot, T) - p_e(\cdot, T) = 0, q(\cdot, T) = 0 \text{ in } \Omega_H \text{ and } p_s(\cdot, T) = 0 \text{ in } \Omega_B, \\
& p_e = p_s \text{ and } \mathbf{M}_e(\cdot) \nabla p_e \cdot \eta = \mathbf{M}_s(\cdot) \nabla p_s \cdot \eta \text{ on } \Sigma_{H,T}, \\
& \mathbf{M}_s(\cdot) \nabla p_s \cdot \eta = 2(u_s - u_e) \text{ on } \Sigma_{B,T}.
\end{aligned} \tag{3.1.5}$$

Theorem 3.1.2. Given an optimal control I_i, I_e and a corresponding solution $(u_{i,e}, v, w, u_s)$, there exists a weak solution (p_i, p_e, p_s, q) to the adjoint system (3.1.4)-(3.1.4) satisfying $p_i, p_e \in L^2(0, T; H^1(\Omega_H))$, $q \in C([0, T], L^2(\Omega_H))$, $p_s \in L^2(0, T; H^1(\Omega_B))$, and the following identities hold for all test functions $\varphi_j, \phi \in \mathcal{D}((0, T) \times \bar{\Omega}_H)$, $j = i, e$ and $\varphi_s, \in \mathcal{D}((0, T) \times \bar{\Omega}_B)$:

$$\begin{aligned}
& \iint_{\Omega_{H,T}} \left\{ \beta c_m (p_i - p_e) \partial_t \varphi_i + \mathbf{M}_i(x) \nabla p_i \cdot \nabla \varphi_i + \beta I_{\text{ion}v}(v, w)(p_i - p_e) \varphi_i - H_v(v, w)q \varphi_i \right\} dx dt = 0, \\
& \iint_{\Omega_{H,T}} \left\{ \beta c_m (p_i - p_e) \partial_t \varphi_e - \mathbf{M}_e(x) \nabla p_e \cdot \nabla \varphi_e - \mathbf{M}_s(x) \nabla p_s \cdot \nabla \varphi_s + \beta I_{\text{ion}v}(v, w)(p_i - p_e) \varphi_e + H_v(v, w)q \right. \\
& \quad \left. - \iint_{\Omega_{B,T}} \mathbf{M}_s(x) \nabla u_s \cdot \nabla \varphi_s dx dt = 0, \right. \\
& \left. \iint_{\Omega_H} q \partial_t \phi dx dt + \iint_{\Omega_H} (\beta I_{\text{ion}w}(v, w)(p_i - p_e) - H_w(v, w)q) \phi dx dt = 0. \right.
\end{aligned} \tag{3.1.6}$$

3.2 Numerical approximation for solving the inverse bidomain model

In this section, we present the finite element method for approximation of the inverse bidomain model and we give the minimization procedures to our inverse problem.

3.2.1 A finite element method

In our discretization for simplicity instead to use the strong coupling boundary conditions (3.2.1), we utilize the following weak coupling boundary conditions: (we assume there is a weak transmission between the heart and the torso):

$$\begin{aligned} (\mathbf{M}_i(x)\nabla u_i) \cdot \mathbf{n} &= 0 \text{ on } \Sigma_{H,T} := \partial\Omega_H \times (0, T), \\ u_e = u_s \text{ and } (\mathbf{M}_e(x)\nabla u_e) \cdot \mathbf{n} &= (\mathbf{M}_s(x)\nabla u_s) \cdot \mathbf{n} = 0 \text{ on } \Sigma_{H,T}, \\ (\mathbf{M}_s(x)\nabla u_s) \cdot \mathbf{n}_s &= 0 \text{ on } (\partial\Omega_B - \partial\Omega_H) \times (0, T). \end{aligned} \quad (3.2.1)$$

For numerical simulations we rewrite (2.1.1) in terms of v , u_e and u_s as the strongly coupled parabolic-elliptic PDE-ODE system (see for e.g. [15])

$$\begin{aligned} \beta c_m \partial_t v + \nabla \cdot (\mathbf{M}_e(x)\nabla u_e) + \beta I_{\text{ion}}(v, w) &= I_e, & (x, t) \in \Omega_{H,T}, \\ \nabla \cdot ((\mathbf{M}_i(x) + \mathbf{M}_e(x))\nabla u_e) + \nabla \cdot (\mathbf{M}_i(x)\nabla v) &= I_e - I_i, & (x, t) \in \Omega_{H,T}, \\ \partial_t w - H(v, w) &= 0, & (x, t) \in \Omega_{H,T}, \\ -\nabla \cdot (\mathbf{M}_s(x)\nabla u_s) &= 0, & (x, t) \in \Omega_{B,T}. \end{aligned} \quad (3.2.2)$$

Now we let \mathcal{T}_H and \mathcal{T}_B (with $\mathcal{T} = \mathcal{T}_H \cap \mathcal{T}_B$) regular partitions of Ω_H and Ω_B , respectively, into tetrahedra K with boundary ∂K and diameter h_K . We define the mesh parameter $h = \max_{K \in \mathcal{T}} \{h_K\}$ and the associated finite element spaces \mathbf{V}_h for the approximation of electrical potentials. For the electrical potentials and ionic variables, we use piecewise linear elements. That is, the involved space is defined as

$$\mathbf{V}_h = \{v \in H^1(\Omega) \cap C^0(\bar{\Omega}) : v|_K \in \mathbb{P}_1(K) \text{ for all } K \in \mathcal{T}\},$$

where $\mathbb{P}_1(K)$ is the set of continuous piecewise linear functions on K . A semidiscrete Galerkin finite element formulation then reads: For $t > 0$, find $\mathbf{u}_h \in \mathbf{V}_h$, $u_e(t), v(t), w(t) \in M_h$, $p \in Q_h$ such that

$$\begin{aligned} \beta c_m \left(\frac{v_h^{n+1} - v_h^n}{\Delta t}, \varphi_{1,h} \right)_{\mathcal{T}_H} - \left(\mathbf{M}_e(x)\nabla u_{e,h}^{n+1}(t), \nabla \varphi_h \right)_{\mathcal{T}_H} &= \\ \left(I_{e,h}^{n+1} - \beta I_{\text{ion}}(v_h^{n+1}(t), w_h^{n+1}(t)), \varphi_{1,h} \right)_{\mathcal{T}_H}, & \\ \left((\mathbf{M}_i(x) + \mathbf{M}_e(x))\nabla u_{e,h}^{n+1}(t), \nabla \varphi_{2,h} \right)_{\mathcal{T}_H} + \left(\mathbf{M}_i(x)\nabla v_h^{n+1}(t), \nabla \varphi_{2,h} \right)_{\mathcal{T}_H} &= \\ \left(I_{e,h}^{n+1} - I_{i,h}^{n+1}, \varphi_{2,h} \right)_{\mathcal{T}_H}, & \\ \left(\frac{v_h^{n+1} - v_h^n}{\Delta t}, \phi_h \right)_{\mathcal{T}_H} = \left(H(v_h^{n+1}(t), w_h^{n+1}(t)), \phi_h \right)_{\mathcal{T}_H}, & \end{aligned} \quad (3.2.3)$$

with $\int_{\mathcal{T}_H} u_{e,h}^{n+1} = 0$, $u_{e,h}^{n+1} = u_{s,h}^{n+1}$ on $\partial\mathcal{T}_H$ and

$$\left(\mathbf{M}_s(x) \nabla u_{s,h}(t) \nabla \varphi_{s,h} \right)_{\mathcal{T}_B} = 0, \quad (3.2.4)$$

for all $\varphi_{j,h}, \phi_h \in \mathbf{V}_h$ for $j = 1, 2, s$. Herein, Δt is a fixed time step, the variables with the superscript n are computed at time $t^n = n\Delta t$.

Note that when solving the Bidomain system, the unknowns of the discrete problem are represented by the vector $(v_h, u_{e,h}, u_{s,h}, w_h)$. Moreover the system (3.2.3) is equivalent to the ODE's:

$$\mathcal{A} \frac{\mathbf{u}_h^{n+1} - \mathbf{u}_h^n}{\Delta t} + \mathcal{B} \mathbf{u}_h^{n+1} = \mathbf{f}_h^n,$$

where \mathcal{A} and \mathcal{B} are the mass and the stiffness matrices, \mathbf{f}_h^n is the source term and $\mathbf{u}_h^n = (v_h, u_{e,h}, u_{s,h}, w_h)$. In the next subsection we give the control and the minimization procedures to our inverse problem.

3.2.2 The minimization procedure

The optimization stage at the discrete level is carried out using the well known non-linear conjugate gradient method (see e.g. [24]). Here we consider the “discretize-then-optimize” approach, and at each iteration of the minimization procedure, the method requires the solution of the discrete state and adjoint equations. The discrete state equations can be solved by marching forward in time starting from the initial conditions (2.1.5), while the discrete adjoint equations can be solved by marching backward in time starting from the terminal conditions (3.1.5).

To compute the optimal control, we improve the initial guess $I_s^0 = I_i^0 - I_e^0$ by using the Jacobian of the reduced objective \hat{J}^k in the conjugate direction $d^k = -\nabla \hat{J}^k$, the latter being also updated at each iteration step, according to the rule $d^{k+1} = -\nabla \hat{J}^k + \varrho^k d^k$, where the sequence $\{\varrho^k\}_k$, is computed using the Hestenes-Stiefel formula [25]

$$\varrho^k = \frac{(\nabla \hat{J}^{k+1}, \nabla \hat{J}^{k+1} - \nabla \hat{J}^k)_{L^2}}{(d^{k-1}, \nabla \hat{J}^{k+1} - \nabla \hat{J}^k)_{L^2}}. \quad (3.2.5)$$

The scaling for the updating of the control at step k is given by δ^k , which is updated following Armijo's rule, i.e., it is reduced by the half until the first Wolfe condition

$$\hat{J}(I_s^k + \delta^k d^k) \leq \hat{J}^k + \alpha d^k \nabla \hat{J}^k$$

is satisfied.

Before presenting our numerical examples, we provide a formal description of the overall solution algorithm.

Algorithm 1 (Overall solution algorithm).

1. Initialization of parameters.
 - (a) Choose tolerance α_{abs} , α_{rel} , set $k = 0$, δ^0 and ϱ^0 .
 - (b) Provide an initial guess I_s^0 for the control variable I_s .
2. **do** $k = 1, \dots, max_outer_iterations$
 - (a) **do** $t = t^1, \dots, t^{total_time_steps}$

Solve the state equations (3.2.2) for (v, u_e, u_i, u_s, w) .

enddo
 - (b) Evaluate the reduced cost functional \hat{J}^k .
 - (c) **do** $t = t^{total_time_steps}, \dots, t^1$

Being known the state variables (v, u_e, u_i, u_s, w) , compute the solution (p, p_i, p_e, p_s, q) of the adjoint problem (3.1.4).

enddo
 - (d) Compute the Jacobian $\nabla \hat{J}^k$.

if the relative and absolute stopping criteria ($\|\nabla \hat{J}^k\|_{L^2} \leq \alpha_{rel} \|\nabla \hat{J}^0\|_{L^2}$ and $\|\nabla \hat{J}^k\|_{L^2} \leq \alpha_{abs}$) are fulfilled,

then exit.

else

 - (i) Compute step length $\delta^k > 0$.
 - (ii) Update the value of the control variable $I_s^{k+1} = I_s^k + \delta^k d^k$.
 - (iii) Compute the step ϱ^k from (3.2.5).
 - (iv) Update the conjugate direction $d^{k+1} = -\nabla \hat{J}^k + \varrho^k d^k$.

endif

enddo

3.2.3 Create the Dynamic Operator

3.2.3.1 Create the Monodomain Inverse Operator

The following system is the monodomain Model. We will use a notation similar to the

one used in [15]

$$\begin{aligned} \frac{\lambda}{1+\lambda} \nabla \cdot (\mathbf{M}_i(x) \nabla v) &= \beta C_m \partial_t v + \beta I_{ion}(v, w) + \beta I_{app}, & (x, t) \in \Omega_{H,T}, \\ \partial_t w &= H(v, w), & (x, t) \in \Omega_{H,T}, \\ (\mathbf{M}_i(x)v) \cdot \mathbf{n} &= 0, & (x, t) \in \partial\Omega_{H,T}, \end{aligned} \quad (3.2.6)$$

and the following equation to get the extracellular potential

$$\nabla \cdot (\mathbf{M}_i(x) \nabla v) = -\nabla \cdot ((1+\lambda)\mathbf{M}_i(x) \nabla u_e), \quad (x, t) \in \Omega_{H,T}. \quad (3.2.7)$$

We will scale the equations with to simplify

$$\mathbf{M}_i(x) = \frac{\mathbf{M}_i(x)}{C_m \beta}. \quad (3.2.8)$$

To solve the Monodomain numerically we use the Godunov operator splitting technique [26], to divide the system into the following:

$$\partial_t v = -I_{ion}(v, w) - I_{app}, \quad (3.2.9)$$

$$\partial_t v = \frac{\lambda}{1+\lambda} \nabla \cdot (\mathbf{M}_i(x) \nabla v). \quad (3.2.10)$$

With Godunov splitting, we first solve

$$\frac{\partial s_1}{\partial t} = -I_{ion}(v, w) - I_{app}, \quad (3.2.11)$$

$$s_1(0) = v(0) \quad (3.2.12)$$

for $t \in [0, \Delta t]$. This gives us $s_1(\Delta t)$. Next we solve

$$\frac{\partial s_2}{\partial t} = \frac{\lambda}{1+\lambda} \nabla \cdot (\mathbf{M}_i(x) \nabla s_2), \quad (3.2.13)$$

$$s_2(0) = s_1(\Delta t) \quad (3.2.14)$$

for $t \in [0, \Delta t]$ to get $s_2(\Delta t)$ which we set equal to

$$v(\Delta t) = s_2(\Delta t). \quad (3.2.15)$$

This means we can solve the system by steps. If we discretize over time 3.2.9, we obtain

$$\frac{v_{n+1} - v_n}{\Delta t} = -I_{ion}(v_n, w_n) - I_{app}, \quad (3.2.16)$$

or

$$v_{n+1} = -\Delta t I_{ion}(v_n, w_n) - \Delta t I_{app} + v_n. \quad (3.2.17)$$

The equation 3.2.10 will be discretized by the following using the θ -rule

$$\frac{v_{n+2} - v_{n+1}}{\Delta t} = \theta \left(\frac{\lambda}{1 + \lambda} \nabla \cdot (\mathbf{M}_i(x) \nabla v_{n+2}) \right) + ((1 - \theta) \frac{\lambda}{1 + \lambda} \nabla \cdot (\mathbf{M}_i(x) \nabla v_{n+1})). \quad (3.2.18)$$

If we multiply by a test function ψ , and rearrange the terms, we have

$$v_{n+2} \psi - \theta \left(\frac{\Delta t \lambda}{1 + \lambda} \nabla \cdot (\mathbf{M}_i(x) \nabla v_{n+2}) \right) \psi = v_{n+1} \psi + ((1 - \theta) \frac{\Delta t \lambda}{1 + \lambda} \nabla \cdot (\mathbf{M}_i(x) \nabla v_{n+1})) \psi, \quad (3.2.19)$$

and applying Green's identity to 3.2.19

$$\begin{aligned} \int_{\Omega_H} v_{n+2} \psi + \theta \left(\frac{\Delta t \lambda}{1 + \lambda} \right) \int_{\Omega_H} \mathbf{M}_i(x) \nabla v_{n+2} \nabla \psi - \int_{\partial \Omega_H} \mathbf{M}_i(x) \psi \nabla v_{n+2} \cdot \mathbf{n} = \\ \int_{\Omega_H} v_{n+1} \psi - (1 - \theta) \frac{\Delta t \lambda}{1 + \lambda} \int_{\Omega_H} \mathbf{M}_i(x) \nabla v_{n+1} \nabla \psi + \int_{\partial \Omega_H} \mathbf{M}_i(x) \psi \nabla v_{n+1} \cdot \mathbf{n}. \end{aligned} \quad (3.2.20)$$

We will consider v_{n+1}, v_{n+2} as a linear combination of basis functions

$$v_{n+1} = \sum_{j=1}^N v_j^{n+1} \phi_j, \quad (3.2.21)$$

$$v_{n+2} = \sum_{j=1}^N v_j^{n+2} \phi_j \quad (3.2.22)$$

and an isolated heart, then equation 3.2.20 becomes, applying FEM approximation:

$$\begin{aligned} \sum_{j=1}^N v_j^{n+2} \left(\int_{\Omega} \phi_j \phi_i + \theta \left(\frac{\Delta t \lambda}{1 + \lambda} \right) \int_{\Omega} \mathbf{M}_i(x) \nabla \phi_j \nabla \phi_i \right) = \\ \sum_{j=1}^N v_j^{n+1} \left(\int_{\Omega} \phi_j \phi_i - (1 - \theta) \frac{\Delta t \lambda}{1 + \lambda} \int_{\Omega} \mathbf{M}_i(x) \nabla \phi_j \nabla \phi_i \right), i, j = 1, \dots, N \end{aligned} \quad (3.2.23)$$

and we get the following matrix

$$A_{ij} = \int_{\Omega} \phi_j \phi_i + \theta \left(\frac{\Delta t \lambda}{1 + \lambda} \right) \int_{\Omega} \mathbf{M}_i(x) \nabla \phi_j \nabla \phi_i, \quad (3.2.24)$$

$$B_{ij} = \int_{\Omega} \phi_j \phi_i - (1 - \theta) \frac{\Delta t \lambda}{1 + \lambda} \int_{\Omega} \mathbf{M}_i(x) \nabla \phi_j \nabla \phi_i. \quad (3.2.25)$$

From equations [3.2.37-3.2.38], we can construct the matrix equation

$$AV_{n+2} = BV_{n+1}, \quad (3.2.26)$$

where V_{n+2}, V_{n+1} are the vectors with the nodal values for the iteration $n + 1, n + 2$ in the numerical resolution. From the equation 3.2.7 we make a relationship from the

transmembrane potential, and the extracellular potential, given by

$$RV_{n+2} = QU_e. \quad (3.2.27)$$

From the equations [3.2.39-3.2.27] we can declare the vector with the nodal vaules for the extracellular potential U_e explicitly by

$$U_e = Q^{-1}RA^{-1}BV_{n+1}. \quad (3.2.28)$$

A relationship with the voltage distribution over the thorax can be constructed using the transfer matrix, considering an isolated heart

$$U_e = U_h, \quad (3.2.29)$$

$$TQ^{-1}RA^{-1}BV_{n+1} = U_t, \quad (3.2.30)$$

or

$$TQ^{-1}RA^{-1}B(-\Delta t\underline{I}_{ion} - \Delta t\underline{I}_{app} + V_n) = U_t, \quad (3.2.31)$$

where $\underline{I}_{app}, \underline{I}_{ion}$ are the vectors with the nodal values for the applied current and ionic flux. From this we will create the operator P , which will be

$$P = TQ^{-1}RA^{-1}B, \quad (3.2.32)$$

and

$$-P\Delta t\underline{I}_{ion} - P\Delta t\underline{I}_{app} + PV_n = U_t. \quad (3.2.33)$$

The inverse problem in electrocardiography is an ill-posed problem, therefore a regularization technique is necessary. We used the following Tikhonov functional:

$$\min_{I_{app}} (|| -P\Delta t\underline{I}_{app} - U_t - P\Delta t\underline{I}_{ion} ||^2 + \mu ||C(\underline{I}_{app} - \underline{I}_{app}') ||^2), \quad \mu > 0, \quad (3.2.34)$$

with the L-Curve method [27] to find the regularization parameter. Here C is a constrained matrix (the identity matrix), and \underline{I}_{app}' is the priori information ($\underline{I}_{app}' = 0$). For our tests we use the value $\mu = 0.00001$

3.2.4 Tests

3.2.4.1 Test 1

This is the original potential distribution generated for the membrane potential (Figure 3.1) and extracellular potential (Figure 3.2) using the monodomain as a model for

electrical activity of the heart. In the images, the basal plane is at the bottom, and the apex is at the top, the main axis of the heart is inverted for visualization purposes. In this experiment an applied current was inserted in a node over the left ventricle at the basal plane during $t < 1$. The order of the images is from left to right, top to bottom.

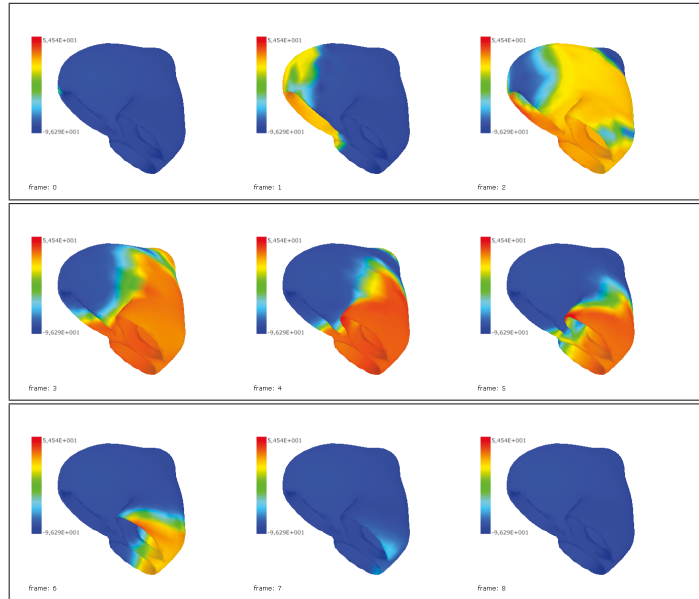


FIGURE 3.1: Membrane Potential for the heart at $t=0$ ms, 50 ms, 100 ms, 150 ms, 200 ms, 250 ms, 300 ms, 350 ms, and 400 ms for one applied impulse at the left ventricle.

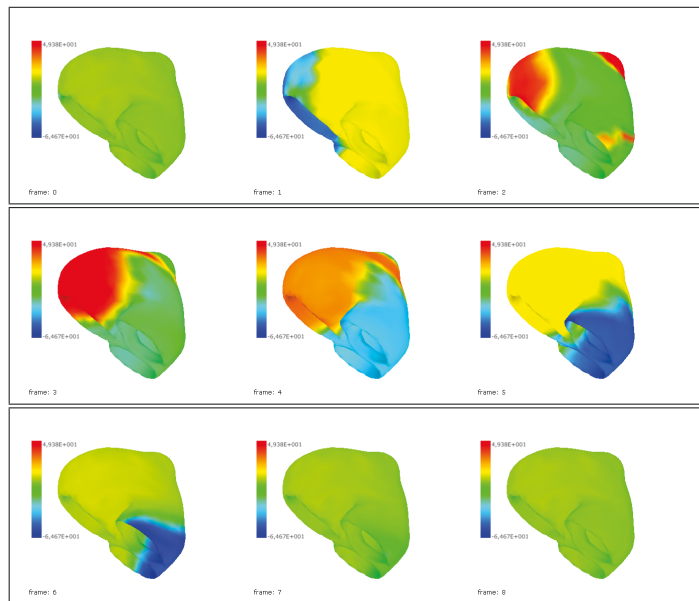


FIGURE 3.2: Extracellular Potential for the heart at $t=0$ ms, 50 ms, 100 ms, 150 ms, 200 ms, 250 ms, 300 ms, 350 ms, and 400 ms for one applied impulse at the left ventricle.

From the extracellular potential and the transfer matrix created from the volume conductor model, and the relationship between the extracellular and thorax potential, we

create the voltage distribution over the thorax in a forward solution. Using the the minimization from equation 3.2.34, for a value $\mu = 0.00001$, and the voltage distribution over the thorax we make the reconstruction of the membrane potential over the heart (Figure 3.3), and the extracellular potential (Figure 3.4).

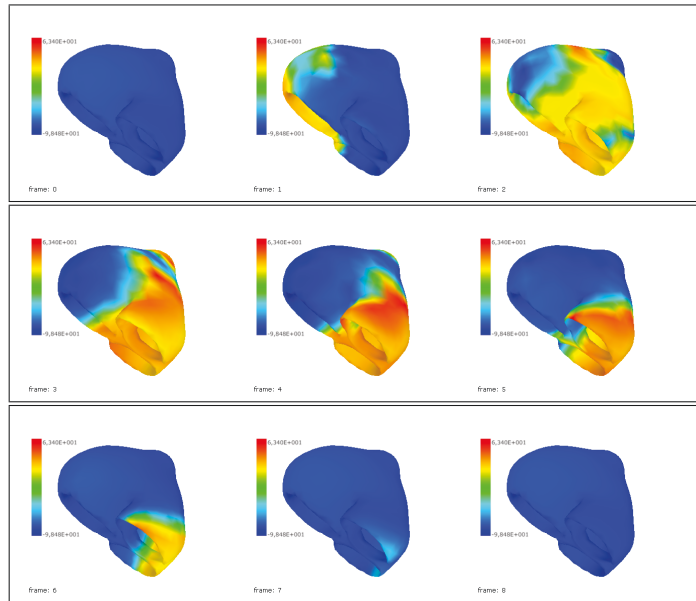


FIGURE 3.3: Membrane Potential calculated using the Monodomain operator for the heart at $t=0$ ms, 50 ms, 100 ms, 150 ms, 200 ms, 250 ms, 300 ms, 350 ms, and 400 ms.

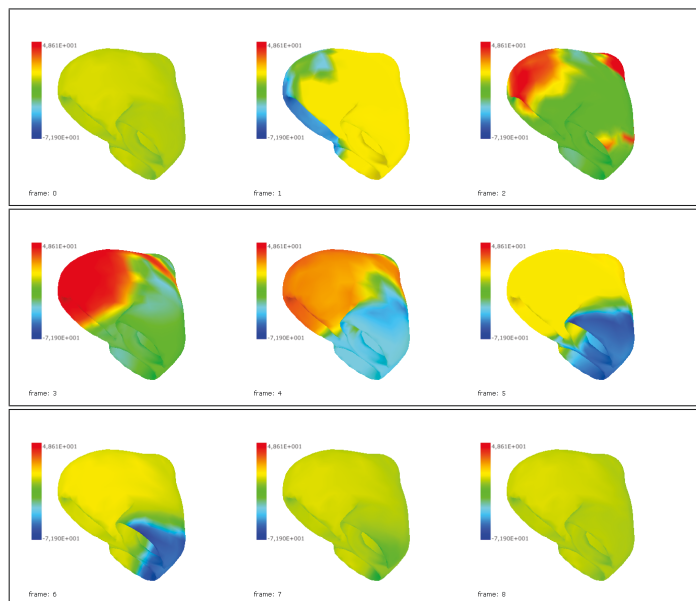


FIGURE 3.4: Extracellular Potential reconstructed with the Monodomain operator for the heart at $t=0$ ms, 50 ms, 100 ms, 150 ms, 200 ms, 250 ms, 300 ms, 350 ms, and 400 ms.

3.2.4.2 Test 2

We make the same process as in experiment 1, but we apply a stimulus at three points instead of one on the basal plane of the heart. The generated electrical activity for the membrane potential and extracellular potential can be seen at Figures 3.5, and 3.6 respectively.

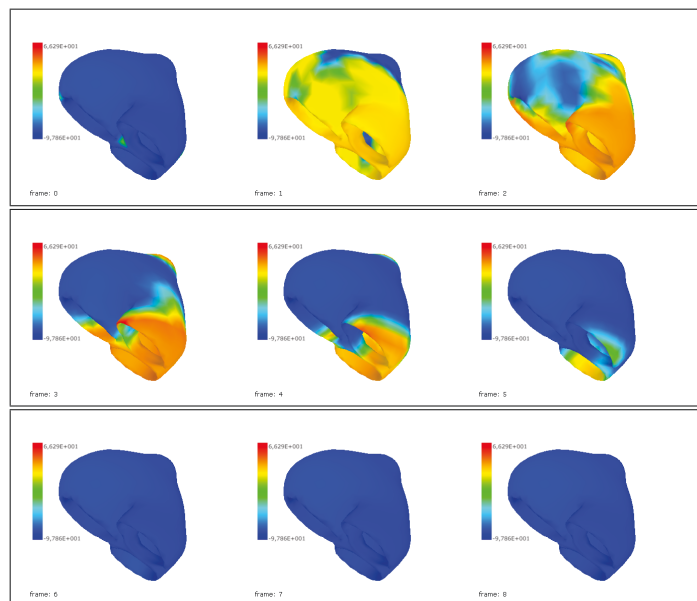


FIGURE 3.5: Membrane Potential for the heart at $t=0$ ms, 50 ms, 100 ms, 150 ms, 200 ms, 250 ms, 300 ms, 350 ms, and 400 ms for three applied stimulus at the basal plane on the heart.

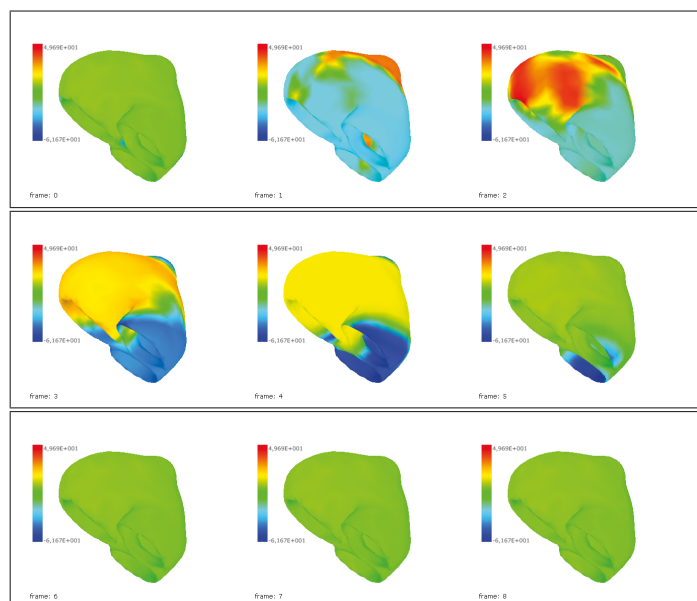


FIGURE 3.6: Extracellular Potential for the heart at $t=0$ ms, 50 ms, 100 ms, 150 ms, 200 ms, 250 ms, 300 ms, 350 ms, and 400 ms for three applied stimulus at the basal plane on the heart.

Using the minimization from equation 3.2.34 we make the reconstruction of both extra-cellular and membrane potential for the thorax distribution generated using a three-point stimulus, Figures 3.7, 3.8 and respectively.

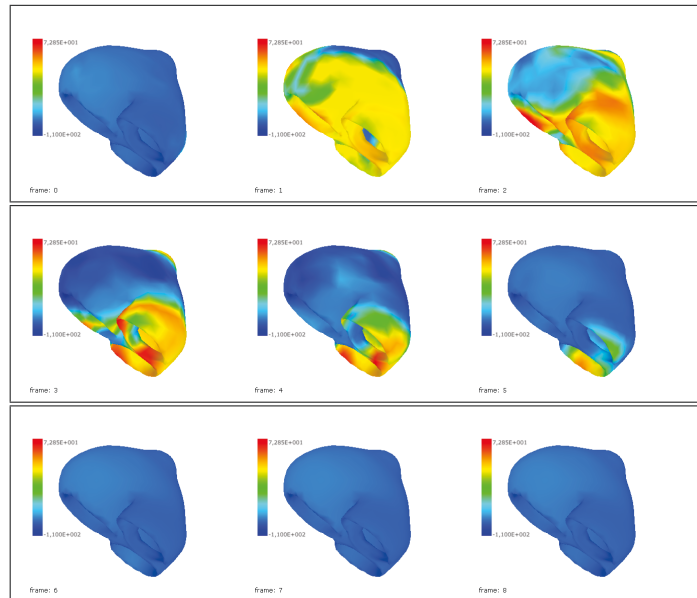


FIGURE 3.7: Membrane Potential calculated using the Monodomain operator for the heart at $t=0$ ms, 50 ms, 100 ms, 150 ms, 200 ms, 250 ms, 300 ms, 350 ms, and 400 ms for the three point stimulus.

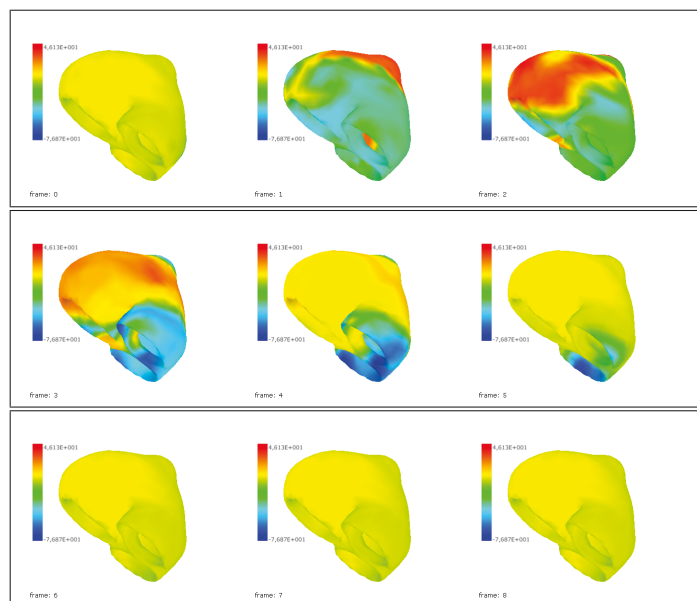


FIGURE 3.8: Extracellular Potential reconstructed with the Monodomain operator for the heart at $t=0$ ms, 50 ms, 100 ms, 150 ms, 200 ms, 250 ms, 300 ms, 350 ms, and 400 ms for the three point stimulus.

3.2.4.3 Test 3

Using the data sets generated by the forward problem in Figure 3.1, we add 1% noise to the potential distribution at the thorax. Then, we make the reconstruction of membrane and extracellular potential. To create the noise we did the following; first we calculate the range from the dataset values over the thorax $Range = Max - Min$. Then for each value

$$Data_{i,j} = Data_{i,j} + 0.01 * Range * Random, Random = -1, \dots, 1. \quad (3.2.35)$$

As in Experiment 1, we make the reconstruction of the membrane potential and extracellular potential Figures 3.9 and 3.10 respectively from the potential distribution over the thorax with added noise.

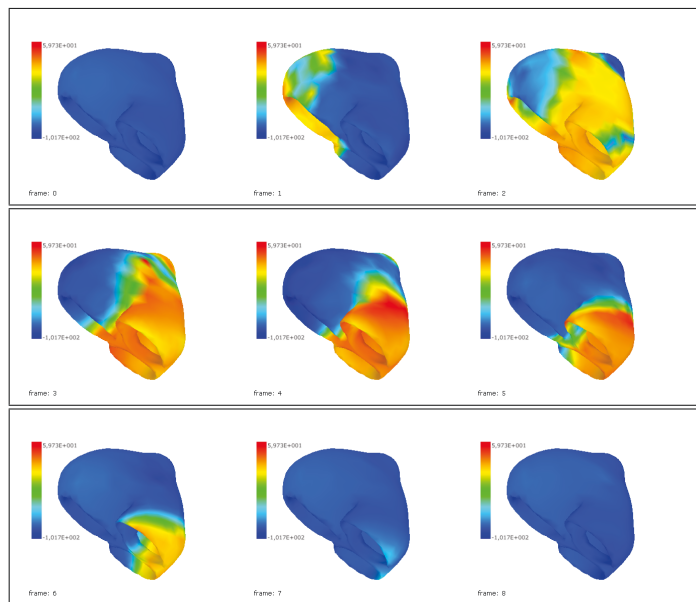


FIGURE 3.9: Membrane Potential calculated using the Monodomain operator for the heart at $t=0$ ms, 50 ms, 100 ms, 150 ms, 200 ms, 250 ms, 300 ms, 350 ms, and 400 ms for the 1 point stimulus, with a 1% noise over the thorax' measures.

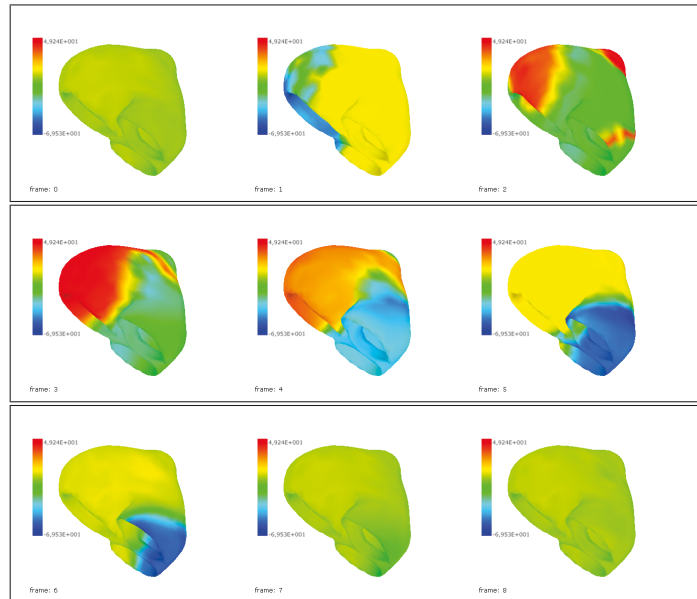


FIGURE 3.10: Extracellular Potential reconstructed using the Monodomain operator for the heart at $t=0$ ms, 50 ms, 100 ms, 150 ms, 200 ms, 250 ms, 300 ms, 350 ms, and 400 ms for the 1 point stimulus, with a 1% noise over the thorax' measures.

3.2.4.4 Test 4

For the membrane potential distribution Figure 3.1, we calculate the forward solution in the thorax using a non-homogeneous operator transfer matrix (including the lungs). Then, for the inverse solution we use the generated distribution at the thorax, and solve the inverse problem with an homogeneous operator. The procedure is detailed in Figure 3.11. The reconstructed membrane, and extracellular potential distribution are in Figures 3.12,3.13 using the monodomain operator.

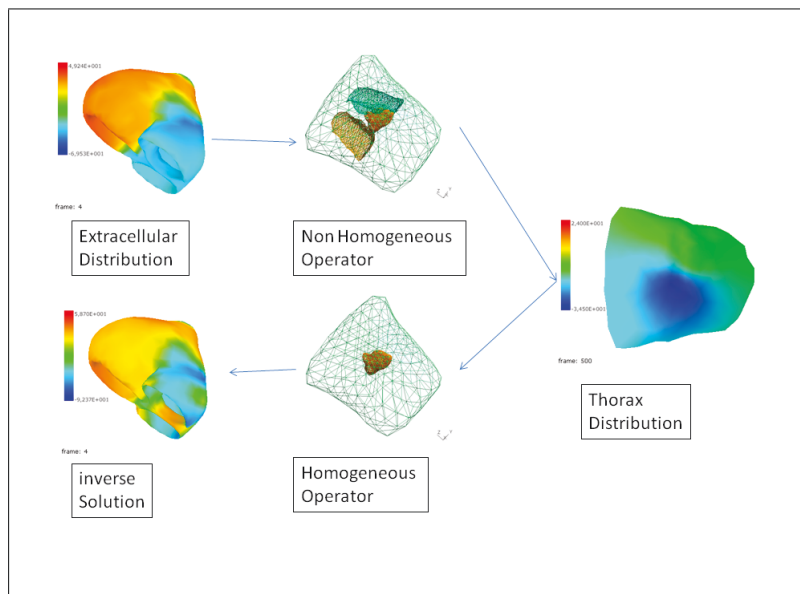
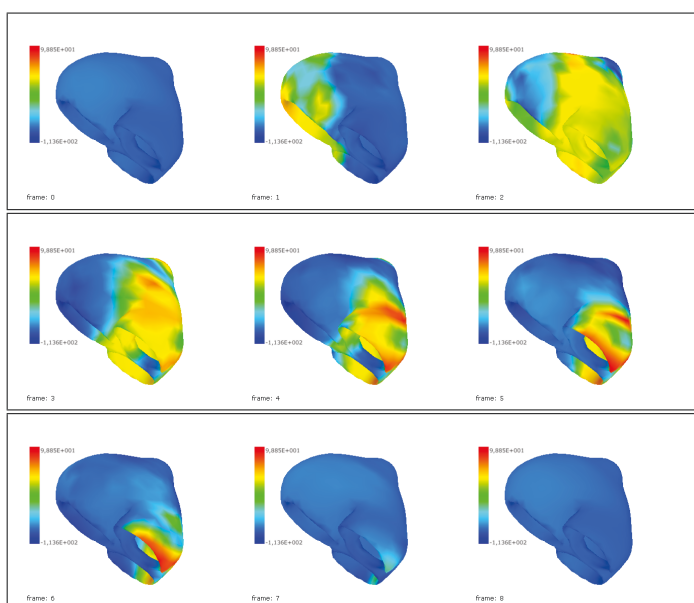


FIGURE 3.11: Procedure of the experiment.

FIGURE 3.12: Membrane Potential calculated using the Monodomain operator for the heart at $t=0$ ms, 50 ms, 100 ms, 150 ms, 200 ms, 250 ms, 300 ms, 350 ms, and 400 ms for the 1 point stimulus, with a homogeneous operator for a non-homogeneous created body surface potentials.

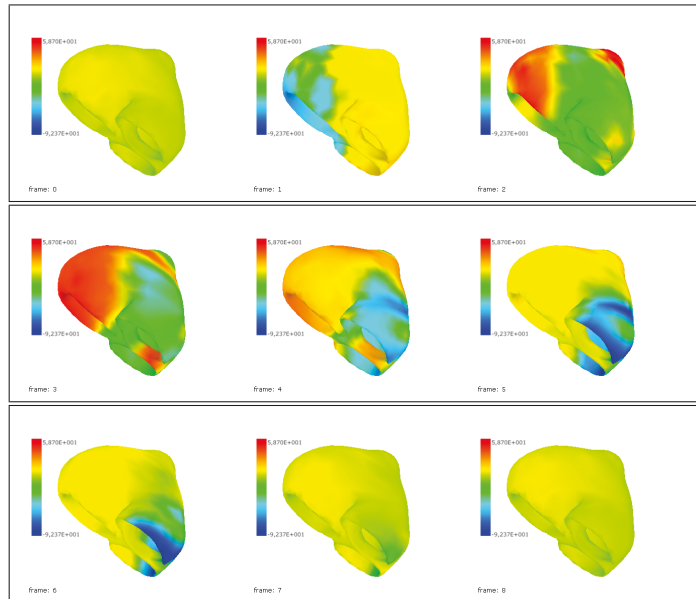


FIGURE 3.13: Extracellular Potential calculated using the Monodomain operator for the heart at $t=0$ ms, 50 ms, 100 ms, 150 ms, 200 ms, 250 ms, 300 ms, 350 ms, and 400 ms for the 1 point stimulus, with a homogeneous operator for a non-homogeneous created body surface potentials.

3.2.4.5 Summary of Results

For each of the tests the error was calculated with,

$$error = \frac{\sum (u_{h_i} - u_{c_i})^2}{\sum (u_{h_i})^2} \quad (3.2.36)$$

for the difference between the original distribution and the calculated: using the monodomain inverse operator. The time simulated is 1 second.

	Test1	Test2	1% Noise	Homogeneous-Non Homogeneous
<i>MonodomainExtracellular</i>	0.0790	0.2318	0.0841	0.3375
<i>MonodomainMembrane</i>	0.0091	0.0219	0.0097	0.0386

3.2.5 Create the Monodomain Inverse Operator on the case of non Isolated Heart

In this section we create an operator considering the continuity of flux between heart and thorax. Until now, we considered the heart decoupled, and therefore no flux. From

the last section we consider the following discrete equations:

$$A_{ij} = \int_{\Omega} \phi_j \phi_i + \theta \left(\frac{\Delta t \lambda}{1 + \lambda} \right) \int_{\Omega} \mathbf{M}_i \nabla \phi_j \nabla \phi_i, \quad (3.2.37)$$

$$B_{ij} = \int_{\Omega} \phi_j \phi_i - (1 - \theta) \frac{\Delta t \lambda}{1 + \lambda} \int_{\Omega} \mathbf{M}_i \nabla \phi_j \nabla \phi_i, \quad (3.2.38)$$

and the matrix equation

$$AV_{n+2} = BV_{n+1}. \quad (3.2.39)$$

From the equation 3.2.7 we will use again the relationship from the transmembrane potential, and the extracellular potential, given by

$$RV_{n+2} = QU. \quad (3.2.40)$$

The difference will be that to consider the continuity of flux this equation will consider the continuity of the flux throughout the thorax giving the following relationship (expressed as a matrix-vector equation):

$$\begin{bmatrix} Q_{hh} & Q_{hv} & 0 \\ Q_{vh} & Q_{vv} & Q_{vt} \\ 0 & Q_{tv} & Q_{tt} \end{bmatrix} U = \begin{bmatrix} R \\ 0 \\ 0 \end{bmatrix} V_{n+2}, \quad (3.2.41)$$

where U is a vector with the nodal values for all the points in the mesh in the discretization for Ω_H, Ω_B . From this equation we will create the following operators:

$$S_1 = Q_{vv} - Q_{vt} Q_{tt}^{-1} Q_{tv}, \quad (3.2.42)$$

$$S_2 = Q_{hh} - Q_{hv} S_1^{-1} Q_{vh}, \quad (3.2.43)$$

$$(3.2.44)$$

and we can express the electrical distribution over the thorax U_t as

$$U_t = Q_{tt}^{-1} Q_{tv} S_1^{-1} Q_{vh} S_2^{-1} R V_{n+1}. \quad (3.2.45)$$

From this we create the operator P_2 , which will be

$$P_2 = Q_{tt}^{-1} Q_{tv} S_1^{-1} Q_{vh} S_2^{-1} R A^{-1} B, \quad (3.2.46)$$

and

$$- P_2 \Delta t \underline{I_{ion}} - P_2 \Delta t \underline{I_{app}} + P_2 V_n = U_t. \quad (3.2.47)$$

The Tikhonov functional will be:

$$\min_{\underline{I}_{app}} (|| - P_2 \Delta t \underline{I}_{app} - U_t - P_2 \Delta t \underline{I}_{ion} ||^2 + \mu || C(\underline{I}_{app} - \underline{I}_{app}') ||^2), \quad \mu > 0, \quad (3.2.48)$$

3.2.6 Tests

3.2.6.1 Test 1

This is the original potential distribution generated for the membrane potential (Figure 3.15) and extracellular potential (Figures 3.14,3.16) using the bidomain as a model for electrical activity of the heart considering continuity of the flux.

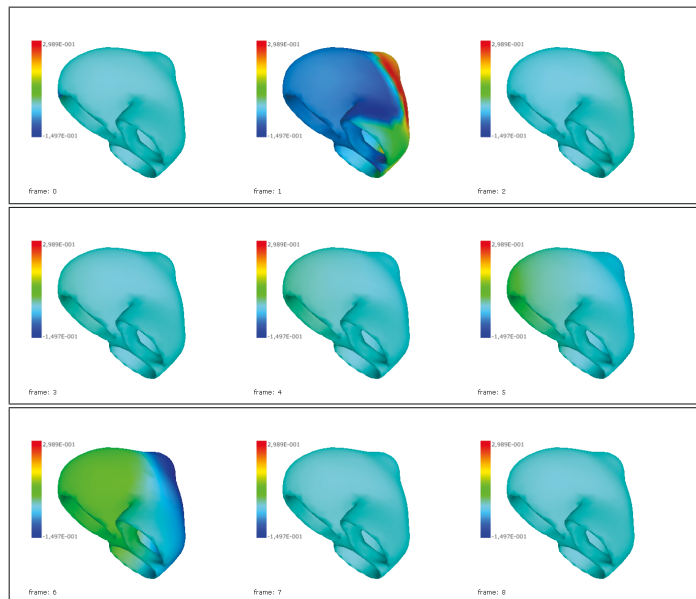


FIGURE 3.14: Extracellular Potential for the heart at $t=0$ ms, 100 ms, 200 ms, 300 ms, 400 ms, 500 ms, 600 ms, 700 ms, and 800 ms for one applied impulse at the left ventricle.

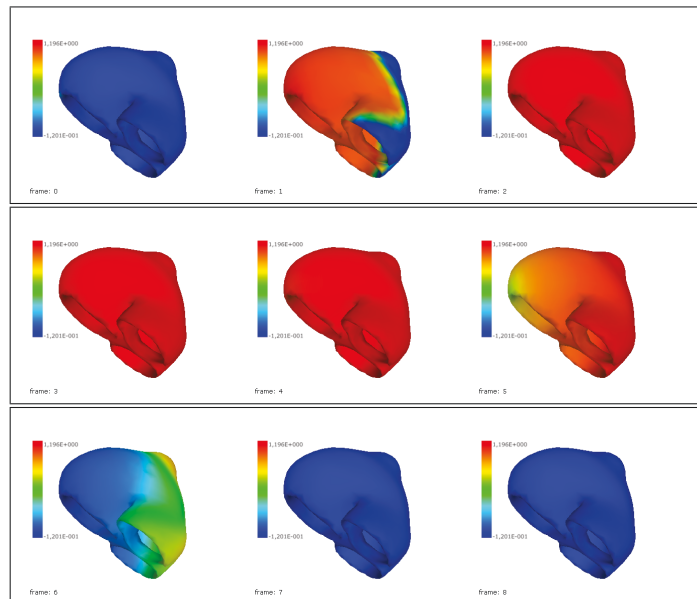


FIGURE 3.15: Transmembrane Potential for the heart at $t=0$ ms, 100 ms, 200 ms, 300 ms, 400 ms, 500 ms, 600 ms, 700 ms, and 800 ms for one applied impulse at the left ventricle.

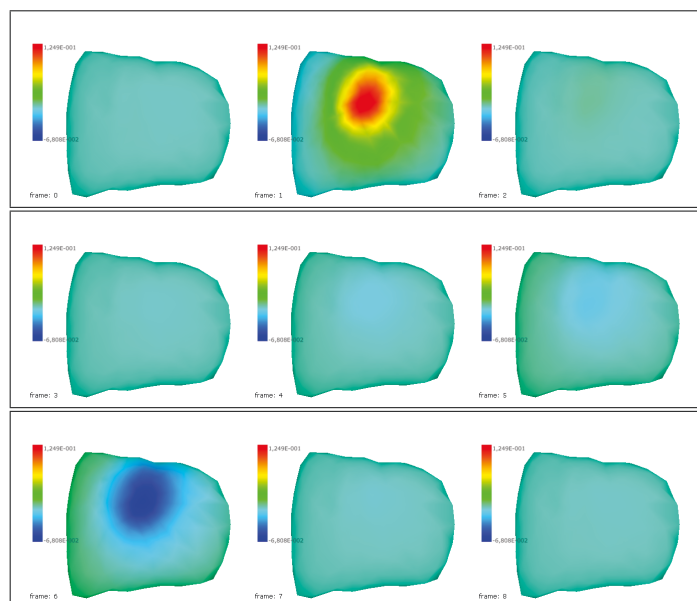


FIGURE 3.16: Extracellular Potential for the thorax at $t=0$ ms, 100 ms, 200 ms, 300 ms, 400 ms, 500 ms, 600 ms, 700 ms, and 800 ms for one applied impulse at the left ventricle.

The reconstruction using the operator with flux for the extracellular, and transmembrane potential are shown in Figures 3.17,3.18 respectively.

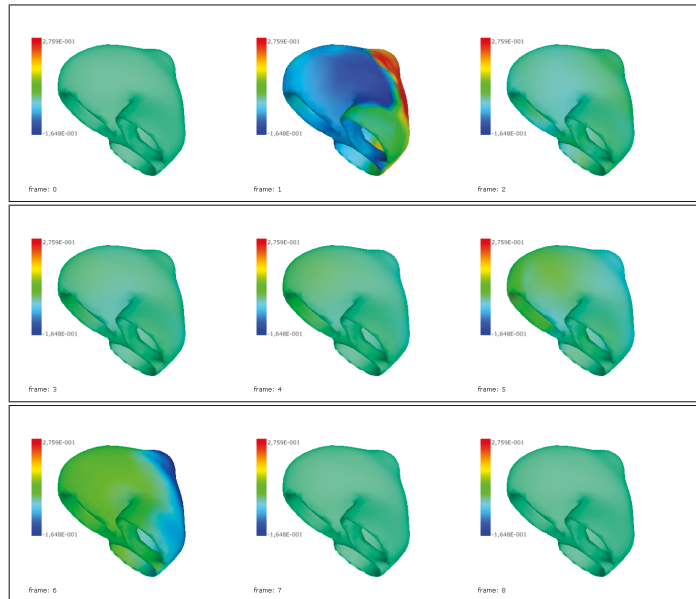


FIGURE 3.17: Reconstruction of the Extracellular Potential for the heart at $t=0$ ms, 100 ms, 200 ms, 300 ms, 400 ms, 500 ms, 600 ms, 700 ms, and 800 ms for one applied impulse at the left ventricle.

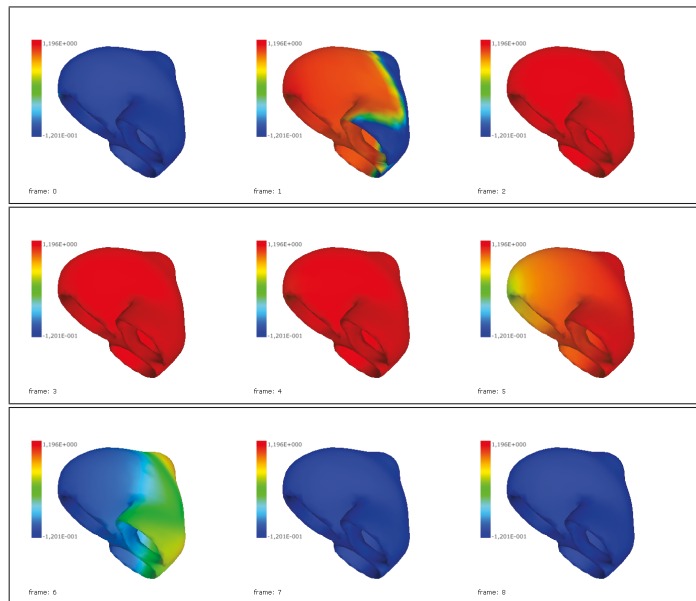


FIGURE 3.18: Reconstruction of the Transmembrane Potential for the heart at $t=0$ ms, 100 ms, 200 ms, 300 ms, 400 ms, 500 ms, 600 ms, 700 ms, and 800 ms for one applied impulse at the left ventricle.

The results are in the following table:

	Test1
<i>MonodomainwithFluxTransmembrane</i>	0.0166
<i>MonodomainwithFluxExtracellular</i>	0.2184

Chapter 4

Chapter 4. Numerical Methods for the Static Volume Conductor Model

4.1 Volume Conductor

In the last chapter we explained the dynamic inverse problem in electrocardiography using the Bidomain model to constraint the solution. To understand better the creation of the operators and the behaviour of the bidomain, monodomain models and their interpretation it is necessary to explain the volume conductor model. We explain in detail the numerical methods for the volume conductor model for the inverse problem in electrocardiography, as it appears in [12, 28]. In chapter 6 we use these techniques in addition to artificial intelligence to improve the solution of the volume conductor model for the inverse problem in electrocardiography.

4.2 The Thorax as a Volume Conductor

The first approach is to consider the body as a volume conductor. In this approach we consider the domain Ω as the torso whose boundary is the thorax surface. We consider the condition that the body is surrounded by an isolating medium, meaning there will

be no current leaving the body. The model will be described by

$$\begin{aligned} -\nabla \cdot (\mathbf{M}(x)\nabla u) &= 0, & x \in \Omega, \\ (\mathbf{M}(x)\nabla u) \cdot \mathbf{n} &= 0, & \text{on } \Sigma, \\ u &= u_e, & \text{on } \partial\Omega_h. \end{aligned} \quad (4.2.1)$$

Here we are dividing the boundary in thorax's surface Σ , and heart's surface $\partial\Omega_h$. Where u_e is the potential in the epicardium or extracellular potential in the heart. If we consider the non-homogeneities in the thorax we can divide the domain Ω in several domains with constant conductivities. For example if we consider the lungs, then we will have two extra domains Ω_L, Ω_R . To ensure continuity, we will declare that the values in the boundary of Ω_T coincide with the values in Ω_L and Ω_R . There is an example of the domains in Figure 4.1.

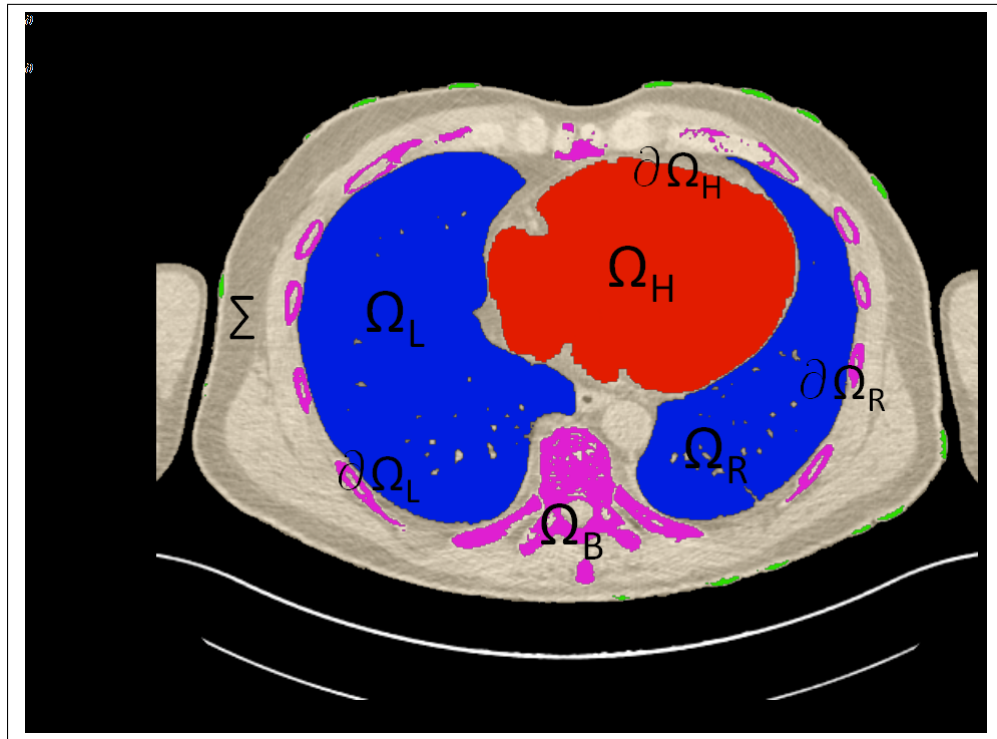


FIGURE 4.1: Separation of domains.

4.2.1 The mathematical model

The spatial domain of our model (thorax, heart, left and right lung) is a bounded open subset $\Omega := \Omega_T \cup \bar{\Omega}_h \cup \bar{\Omega}_L \cup \bar{\Omega}_R$ in \mathbb{R}^3 with a piecewise smooth boundary, $\partial\Omega$. Herein, Ω_T , Ω_h , Ω_L and Ω_R are the spatial domains of non-organic thorax, heart, left and right lung, respectively, with piecewise smooth boundaries $\partial\Omega_T$, $\partial\Omega_h$, $\partial\Omega_L$ and $\partial\Omega_R$. The quantities

of interest are the electric potentials of *left lung*, *right lung* and the bathing medium, $u_L = u_L(x)$, $u_R = u_R(x)$ and $u_T = u_T(x)$ for $x \in \Omega_L$, $x \in \Omega_R$ and $x \in \Omega_T$, respectively. Observe that $\partial\Omega_T = \Sigma \cup \partial\Omega_h \cup \partial\Omega_L \cup \partial\Omega_R$, where Σ is a thoracic surface. Note that ECG signals monitor the electrical activity of the heart from potential measurements at the torso skin surface Σ . The conductivity of the tissue is represented by scaled tensors $\mathbf{M}_L(x)$ (for the left lung), $\mathbf{M}_R(x)$ (for the right lung) and $\mathbf{M}_T(x)$ (for the bathing medium). The forward problem consists of solving the following system: for $k = L, R, T$

$$\begin{aligned}
-\nabla \cdot (\mathbf{M}_k(x) \nabla u_k) &= 0, & x \in \Omega_k, \\
& \text{[continuity of electric potential and flux]} \\
u_L &= u_T \text{ and } \mathbf{M}_L(x) \nabla u_L \cdot \mathbf{n}_L = \mathbf{M}_T(x) \nabla u_T \cdot \mathbf{n}_L \text{ on } \partial\Omega_L, \\
u_R &= u_T \text{ and } \mathbf{M}_R(x) \nabla u_R \cdot \mathbf{n}_R = \mathbf{M}_T(x) \nabla u_T \cdot \mathbf{n}_R \text{ on } \partial\Omega_R, & (4.2.2) \\
& \text{[skin is an insulator]} \\
(\mathbf{M}_T(x) \nabla u_T) \cdot \mathbf{n}_T &= 0 \text{ on } \Sigma, \\
u_T &= u_e \text{ on } \partial\Omega_h,
\end{aligned}$$

for u_e given on the epicard $\partial\Omega_h$. Moreover, the inverse problem consists on solving for u_t on $\partial\Omega_h$ the following boundary value problem: for $k = L, R, T$

$$\begin{aligned}
-\nabla \cdot (\mathbf{M}_k(x) \nabla u_k) &= 0, & x \in \Omega_k, \\
u_L &= u_T \text{ and } \mathbf{M}_L(x) \nabla u_L \cdot \mathbf{n}_L = \mathbf{M}_T(x) \nabla u_T \cdot \mathbf{n}_L \text{ on } \partial\Omega_L, \\
u_R &= u_T \text{ and } \mathbf{M}_R(x) \nabla u_R \cdot \mathbf{n}_R = \mathbf{M}_T(x) \nabla u_T \cdot \mathbf{n}_R \text{ on } \partial\Omega_R, & (4.2.3) \\
(\mathbf{M}_T(x) \nabla u_T) \cdot \mathbf{n}_T &= 0 \text{ and } u_T = \bar{u} \text{ on } \Sigma,
\end{aligned}$$

with a given \bar{u} on Σ . Herein, \mathbf{n}_k is the outward unit normal to $\partial\Omega_k$ for $k = L, R, T$.

4.2.2 Solution of the Forward Problem

We assume that the matrix function \mathbf{M}_j , $j \in \{L, R, T\}$, is sufficiently smooth so that the following definition of weak solution makes sense. Furthermore, we assume $\mathbf{M}_j \in L^\infty(\Omega, \mathbb{R}^{3 \times 3})$ and $\mathbf{M}_j \boldsymbol{\xi} \cdot \boldsymbol{\xi} \geq C_M |\boldsymbol{\xi}|^2$ for a.e. $x \in \Omega$, for all $\boldsymbol{\xi} \in \mathbb{R}^3$, $j \in \{L, R, T\}$, and a constant $C_M > 0$. Before we define a weak solution, we let

$$\mathbf{M}(x) = \begin{cases} \mathbf{M}_L(x) & \text{for } x \in \Omega_L, \\ \mathbf{M}_R(x) & \text{for } x \in \Omega_R, \\ \mathbf{M}_T(x) & \text{for } x \in \Omega_T, \end{cases} \text{ and } \mathbf{u}(x) = \begin{cases} u_R(x) & \text{for } x \in \Omega_R, \\ u_L(x) & \text{for } x \in \Omega_L, \\ u_T(x) & \text{for } x \in \Omega_T. \end{cases}$$

For later reference, we now state the definition of a weak solution for our direct problem (4.2.2):

Definition: A function \mathbf{u} is a *weak solution* of the system (4.2.2) if $\mathbf{u} \in H^1(\Omega)$ and $\mathbf{u} = u_e$ on $\partial\Omega_h$, and the following identity holds:

$$\int_{\Omega} \mathbf{M}(x) \nabla \mathbf{u} \cdot \nabla \varphi \, dx = 0, \quad (4.2.4)$$

for all test functions $\varphi \in H^1(\Omega)$ such that

$$\varphi = \begin{cases} \varphi_L & \text{in } \Omega_L, \\ \varphi_R & \text{for } \Omega_R, \\ \varphi_T & \text{for } \Omega_T, \end{cases} \begin{cases} \varphi_R = \varphi_T & \text{on } \partial\Omega_R, \\ \varphi_L = \varphi_T & \text{on } \partial\Omega_L. \end{cases}$$

Theorem 4.1. *If $u_e \in H^{1/2}(\partial\Omega_h)$, then the system (4.2.2) possesses a unique weak solution.*

In Theorem 4.1, $H^{1/2}(\partial\Omega_h)$ is defined as follows

$$H^{1/2}(\partial\Omega_h) = \{u \in L^2(\partial\Omega_h) : \|u\|_{H^{1/2}(\Sigma)} = \left(\sum_{i=1}^{\infty} \alpha_i^{1/2} u_i^2 \right)^{1/2} < \infty\}.$$

Herein, for $i = 1, \dots, \infty$, u_i are the Fourier coefficients of u relative to the eigenfunctions of the Laplace operator $-\Delta_{\partial\Omega_h}$ on $\partial\Omega_h$, and the corresponding eigenvalues α_i .

Existence: First, we define the following space

$$\mathcal{H} = \{\varphi \in H^1(\Omega) : \varphi = 0 \text{ on } \partial\Omega_h\}.$$

Next, we let $\mathbf{v} \in H^1(\Omega)$ be such that $\mathbf{v} = u_e$ on $\partial\Omega_h$. Using the trace theorem, we get

$$\|\mathbf{v}\|_{H^1(\Omega)} \leq \|u_e\|_{H^{1/2}(\partial\Omega_h)}.$$

Moreover, according to the classical operator theory (see for e.g. [29]), there exists $w \in \mathcal{H}$ such that

$$\int_{\Omega} \mathbf{M}(x) \nabla \mathbf{v} \cdot \nabla \varphi \, dx = - \int_{\Omega} \mathbf{M}(x) \nabla w \cdot \nabla \varphi \, dx,$$

for all $\varphi \in \mathcal{H}$. Now we let $\mathbf{u} = \mathbf{v} + w$ and we see that

$$\int_{\Omega} \mathbf{M}(x) \nabla \mathbf{u} \cdot \nabla \varphi \, dx = \int_{\Omega} \mathbf{M}(x) \nabla (\mathbf{v} + w) \cdot \nabla \varphi \, dx = 0 \text{ and } \mathbf{u} = \mathbf{v} + w = u_e \text{ on } \partial\Omega_h,$$

for all $\varphi \in \mathcal{H}$.

Uniqueness: According to the definition of weak solution, the following equation holds for all test functions $\varphi \in \mathcal{H}$:

$$\int_{\Omega} \mathbf{M}(x) \nabla(\mathbf{u}_1 - \mathbf{u}_2) \cdot \nabla \varphi \, dx = 0, \quad (4.2.5)$$

where \mathbf{u}_1 and \mathbf{u}_2 are two weak solutions. We let $\varphi = \mathbf{u}_1 - \mathbf{u}_2$ in (4.2.5) to obtain (note that $\mathbf{u}_1 - \mathbf{u}_2 = 0$ on $\partial\Omega_h$)

$$\int_{\Omega} \mathbf{M}(x) \nabla(\mathbf{u}_1 - \mathbf{u}_2) \cdot \nabla(\mathbf{u}_1 - \mathbf{u}_2) \, dx = 0.$$

This implies, using coercivity that $\mathbf{u}_1 - \mathbf{u}_2$ is constant in Ω and since $\mathbf{u}_1 - \mathbf{u}_2 = 0$ on $\partial\Omega_h$, we deduce that

$$\mathbf{u}_1 - \mathbf{u}_2 = 0 \text{ in } \Omega.$$

4.2.3 Finite Element Method for the Inverse Problem

The relationship between heart and thorax is given by the model of Laplace in 3D with inhomogeneous boundary conditions. We discretize our equation by finite element method (FEM) approximation to create a transfer matrix.

$$\begin{aligned} -\nabla \cdot (\mathbf{M}(x) \nabla u) &= 0 & x \in \Omega, \\ u &= u_e & \text{on } \partial\Omega_h, \\ (\mathbf{M}(x) \nabla u) \cdot n &= 0 & \text{on } \Sigma, \\ u &= u_T & \text{on } \Sigma. \end{aligned} \quad (4.2.6)$$

In this system we have as given information the potential on Σ that represents the thorax, and $\partial\Omega_h$ will represent the heart's surface. We consider as given the conditions on the thorax's surface, and the unknown is the potential distribution on the heart's surface. These conditions are; the measures that will be considered as Dirichlet conditions, and the null flux condition. In summary, we have *two* boundary conditions on the thorax and no condition on the heart's surface. We consider that the body is surrounded by a non-conductive medium and the outward normal flux is too small to be considered in the model; thus a null-flux condition.

In the Figure 4.2 we can see a simplified triangulation of the heart, thorax and lungs, which will be used for the later calculations. This geometry was modified from the physionet library [30] 2007 datasets. A nice feature of the FEM is that these types of systems produce a matrix where most of the entries are zero and, therefore, are easy to solve. The problem using FEM is that it is necessary to create the elements for the

whole domain. This implies that the number of equations will be quite large, and it is also necessary to treat the geometry of the domain.

Several articles use this formulation to develop an operator for the volume conductor

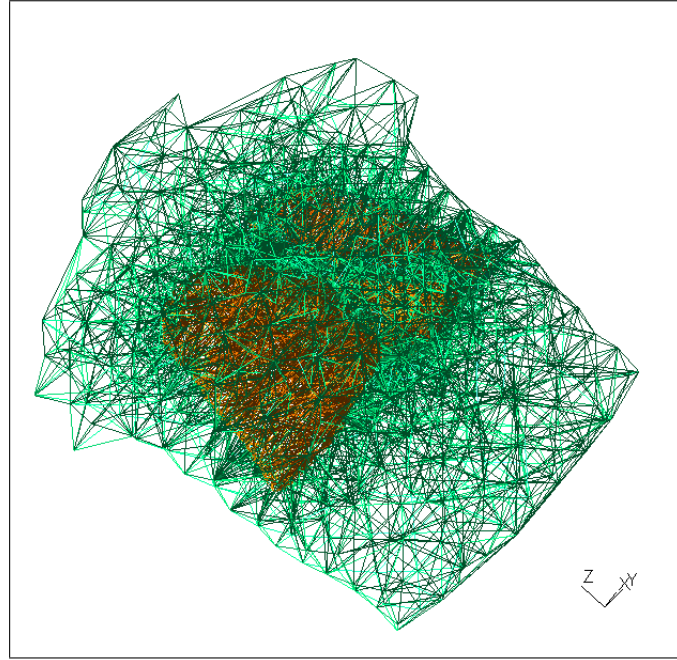


FIGURE 4.2: Example of the heart thorax and lungs for the FEM.

model [31], and then use it to solve the inverse problem. This is why the explanation of the FEM is so important as a numerical method for the problem we are considering.

4.2.4 Electrostatic formulation

When two objects near each other have different electrical charges an electrostatic field is formed between them. An electrostatic field also exists when an object is electrically charged with respect to its surrounding environment. These arise from a potential difference between the static charges in them.

The electric potential u in the domain Ω can be determined from conditions on its boundary $\partial\Omega = \overline{\Gamma_1} \cup \overline{\Gamma_2}$ (where $\Gamma_1 \cap \Gamma_2 = \emptyset$) via the partial differential equation

$$-\nabla \cdot (\kappa \nabla u) = f \text{ in } \Omega,$$

with boundary conditions

$$u = g \text{ on } \Gamma_1,$$

$$(\kappa \nabla u) \cdot \mathbf{n} = h \text{ on } \Gamma_2.$$

(4.2.7)

Equation 4.2.7 is known as the Poisson Equation. This equation is used to calculate the potential/ temperature/ displacement equilibrium, in a domain. The domain Ω can be

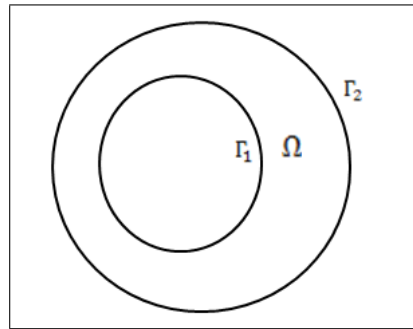


FIGURE 4.3: Example of Domain.

any solid (3D), plate (2D), or just a line (1D). The domain has to be closed; this means that in the whole boundary of the domain there has to be a boundary condition.

The domain is divided into *elements*, this is why the method is known as the finite element method. For the 2D case the domain is usually divided into triangles or rectangles. We will use here triangles in 2D, and tetrahedra in 3D. For example, in the following image we can see a square divided into triangles for the finite element analysis.

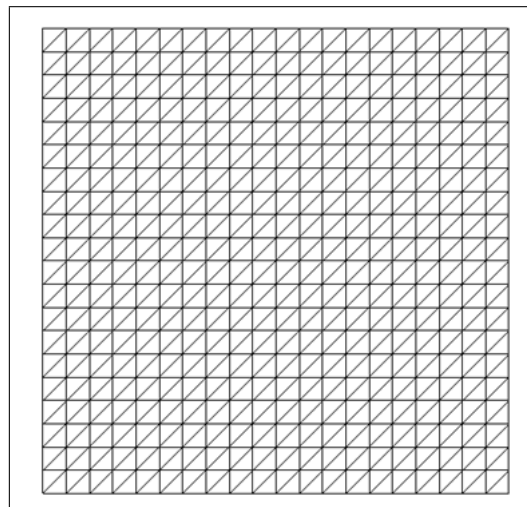


FIGURE 4.4: Domain divided into triangular elements in 2D.

In our first example the square will be the domain Ω . The sides of the square will be the boundary, on which the prescribed electrical activity conditions will be potential and flux. When considering potential on each element, we will be talking of potential at a node. The flux will be the current entering the element through an edge.

For example, the flux in 2D will enter an edge (the segment joining two consecutive nodes on the boundary). When we talk about the 3D case, it will be the flux entering a side of a tetrahedral element, i.e. a triangle.

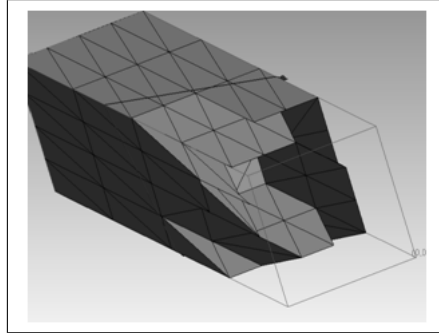


FIGURE 4.5: Detail of a 3D domain.

For the electrostatic field, we obtain from Maxwell's equation the following system of partial differential equations (assuming quasi-static fields):

$$\begin{aligned}\nabla \times E &= 0, \\ \nabla \cdot D &= \rho_v, \\ D &= \epsilon E,\end{aligned}\tag{4.2.8}$$

where E represents electric field intensity (V/m), D is electric flux density (As/m^2), ρ_v is charge density (As/m^3), and ϵ is electric permmissivity (F/m). Since the *curl* of the electric field intensity E is zero, we can express it as the gradient of a scalar potential u , called the electric scalar potential,

$$E = -\nabla u.\tag{4.2.9}$$

The electrostatic boundary value problem can be written as:

$$\begin{aligned}-\nabla \cdot (\epsilon \nabla u) &= \rho_v \text{ in } \Omega, \\ u &= g \text{ on } \Gamma_1, \\ (\epsilon \nabla u) \cdot \mathbf{n} &= h \text{ on } \Gamma_2.\end{aligned}\tag{4.2.10}$$

In general, permmissivity ϵ is a second order tensor.

4.2.5 Basics

We will consider the following boundary value problem

$$\begin{aligned} -\nabla \cdot (\kappa \nabla u) &= f \text{ in } \Omega, \\ u &= 0 \text{ on } \partial\Omega. \end{aligned} \tag{4.2.11}$$

If we multiply by the test function v , 4.2.11 yields:

$$-\nabla \cdot (\kappa \nabla u)v = fv \text{ in } \Omega. \tag{4.2.12}$$

Thus,

$$-\int_{\Omega} \nabla \cdot (\kappa \nabla u)v \, d\Omega = \int_{\Omega} fv \, d\Omega. \tag{4.2.13}$$

Next, we apply the Green's identity:

$$-\int_{\Omega} \nabla \cdot (\kappa \nabla u)v \, d\Omega = \int_{\Omega} \kappa \nabla u \nabla v \, d\Omega - \int_{\partial\Omega} v \kappa \nabla u \cdot \mathbf{n} \, d\partial\Omega. \tag{4.2.14}$$

Since $v = 0$ on $\partial\Omega$,

$$\int_{\partial\Omega} v(\kappa \nabla u) \cdot \mathbf{n} \, d\partial\Omega = 0. \tag{4.2.15}$$

Combining 4.2.13, 4.2.14 and 4.2.15 we see that

$$\int_{\Omega} \kappa \nabla u \cdot \nabla v \, d\Omega = \int_{\Omega} fv \, d\Omega. \tag{4.2.16}$$

This is known as the weak formulation. The domain Ω will be decomposed into a union of non-overlapping triangles (2D) or tetrahedra (3D), with a number of free nodes N_f . Free nodes means nodes that are not on the boundary. Let us write now the FEM approximation in terms of the basis functions ϕ_j as

$$w(x) = \sum_{j=1}^N u_j \phi_j. \tag{4.2.17}$$

The nodal basis function ϕ is the unique linear function defined by the values at the vertices x_k of each element having x_j as a vertex:

$$\phi_j(x_k) = \begin{cases} 1, & \text{if } j = k \\ 0, & \text{if } j \neq k. \end{cases} \tag{4.2.18}$$

Using $v = \phi_i$, for $i = 1, 2, \dots, N$:

$$\sum_{j=1}^N u_j \int_{\Omega} \kappa \nabla \phi_i \nabla \phi_j \, d\Omega = \int_{\Omega} \phi_i f \, d\Omega. \quad (4.2.19)$$

We now define the stiffness matrix K as:

$$K_{ij} = \int_{\Omega} \kappa \nabla \phi_i \nabla \phi_j \, d\Omega \quad i, j = 1, 2, \dots, N, \quad (4.2.20)$$

and the load vector F as:

$$F_i = \int_{\Omega} \phi_i f \, d\Omega \quad i = 1, 2, \dots, N. \quad (4.2.21)$$

if the domain is divided in N_t elements T_k then K will have the form:

$$K_{ij} = \sum_{k=1}^{N_t} \int_{T_k} \kappa \nabla \phi_i \nabla \phi_j \, d\Omega \quad i, j = 1, 2, \dots, N. \quad (4.2.22)$$

4.2.5.1 Inhomogeneous boundary conditions

When we have the system

$$\begin{aligned} -\nabla \cdot (\kappa \nabla u) &= f \text{ in } \Omega, \\ u &= g \text{ on } \Gamma_1, \\ \kappa \frac{\partial u}{\partial n} &= h \text{ on } \Gamma_2, \end{aligned} \quad (4.2.23)$$

u will be decomposed as $u = \bar{u} + G$, where \bar{u} is the solution of

$$\begin{aligned} -\nabla \cdot (\kappa \nabla \bar{u}) &= f \text{ in } \Omega, \\ \bar{u} &= 0 \text{ on } \Gamma_1, \\ \kappa \frac{\partial \bar{u}}{\partial n} &= h \text{ on } \Gamma_2, \end{aligned} \quad (4.2.24)$$

and G is the solution of

$$\begin{aligned} -\nabla \cdot (\kappa \nabla G) &= 0 \text{ in } \Omega, \\ G &= g \text{ on } \Gamma_1, \\ \kappa \frac{\partial G}{\partial n} &= 0 \text{ on } \Gamma_2. \end{aligned} \quad (4.2.25)$$

The weak formulation will then have the following form:

$$\int_{\Omega} \kappa \nabla \bar{u} \nabla v \, d\Omega = \int_{\Omega} f v \, d\Omega + \int_{\Gamma_2} h v \, d\Gamma_2 - \int_{\Omega} \kappa \nabla G \nabla v \, d\Omega. \quad (4.2.26)$$

The load vector is calculated as:

$$F_i = \int_{\Omega} f \phi_i d\Omega + \int_{\Gamma_2} h \phi_i d\Gamma_2 - \int_{\Omega} \kappa \nabla G \nabla \phi_i d\Omega. \quad (4.2.27)$$

The Neumann condition uses the flux, so in the 2D mesh composed by triangles the boundary will be an edge (line). For the 3D case, the integral will be calculated over a triangle. The value of the Neumann contribution for 2D case will be calculated using:

$$\int_{\Gamma_2} h \phi_i d\Gamma_2 = \int_{edge\ 1} h \phi_i d(edge\ 1) + \int_{edge\ 2} h \phi_i d(edge\ 2). \quad (4.2.28)$$

4.2.5.2 Another PDE

Consider the following boundary value problem (BVP):

$$\begin{aligned} -\nabla \cdot (\kappa \nabla u) + bu &= f \text{ in } \Omega, \\ \kappa \nabla u n &= h \text{ on } \partial\Omega. \end{aligned} \quad (4.2.29)$$

This discrete system will have the following components:

$$K_{ij} = \int_{\Omega} \kappa \nabla \phi_i \nabla \phi_j d\Omega, \quad (4.2.30)$$

$$M_{ij} = \int_{\Omega} b \phi_i \phi_j d\Omega, \quad (4.2.31)$$

$$F_i = \int_{\Omega} f \phi_i d\Omega + \int_{\partial\Omega} h \phi_i d\partial\Omega, \quad (4.2.32)$$

The matrix equation form it will be $(K+M)U=F$.

4.2.5.3 Stiffness Matrix

The Stiffness Matrix is calculated using the following:

$$K_{ij} = \sum_{k=1}^{N_t} \int_{T_k} \kappa \nabla \phi_i \nabla \phi_j d\Omega. \quad (4.2.33)$$

where ϕ_j is a basis function:

$$\phi_j(x_k) = \begin{cases} 1, & \text{if } j = k \\ 0, & \text{if } j \neq k \end{cases}. \quad (4.2.34)$$

The function ϕ_i in 3D is represented by:

$$\phi_i(x, y, z) = a_i + b_i x + c_i y + d_i z, \quad (4.2.35)$$

for $i, j = 1, 2, 3, 4$. The coefficients a_i, b_i, c_i, d_i can be found by solving a 4 x 4 system of equations. For example, for $i = 1$ we get the following system:

$$\begin{aligned} a_1 + b_1 x_1 + c_1 y_1 + d_1 z_1 &= 1 \\ a_1 + b_1 x_2 + c_1 y_2 + d_1 z_2 &= 0 \\ a_1 + b_1 x_3 + c_1 y_3 + d_1 z_3 &= 0 \\ a_1 + b_1 x_4 + c_1 y_4 + d_1 z_4 &= 0. \end{aligned} \quad (4.2.36)$$

If we combine the 4 systems and put them in matrix form we have the next system:

$$\begin{bmatrix} 1 & x_1 & y_1 & z_1 \\ 1 & x_2 & y_2 & z_2 \\ 1 & x_3 & y_3 & z_3 \\ 1 & x_4 & y_4 & z_4 \end{bmatrix} \begin{bmatrix} a_1 & a_2 & a_3 & a_4 \\ b_1 & b_2 & b_3 & b_4 \\ c_1 & c_2 & c_3 & c_4 \\ d_1 & d_2 & d_3 & d_4 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (4.2.37)$$

where

$$\begin{bmatrix} a_1 & a_2 & a_3 & a_4 \\ b_1 & b_2 & b_3 & b_4 \\ c_1 & c_2 & c_3 & c_4 \\ d_1 & d_2 & d_3 & d_4 \end{bmatrix}, \quad (4.2.38)$$

is the inverse of

$$\begin{bmatrix} 1 & x_1 & y_1 & z_1 \\ 1 & x_2 & y_2 & z_2 \\ 1 & x_3 & y_3 & z_3 \\ 1 & x_4 & y_4 & z_4 \end{bmatrix}. \quad (4.2.39)$$

To assemble the stiffness matrix is necessary to calculate the following integral:

$$\int_{T_k} \kappa \nabla \phi_i \cdot \nabla \phi_j dT_k. \quad (4.2.40)$$

Since ϕ_i is linear over T_k (the k -th Tetrahedron), the gradients are constant:

$$\nabla \phi_i = \begin{bmatrix} b_i \\ c_i \\ d_i \end{bmatrix}. \quad (4.2.41)$$

Therefore,

$$\int_{T_k} \kappa \nabla \phi_i \cdot \nabla \phi_j dT_k = (\nabla \phi_i \cdot \nabla \phi_j) \int_{T_k} \kappa dT_k, \quad (4.2.42)$$

and the integral of κ can be estimated with the 1-point rule.

$$\int_{T_k} \kappa dT_k \simeq V * \kappa(\bar{x}, \bar{y}, \bar{z}), \quad (4.2.43)$$

where

$$\begin{aligned} \bar{x} &= \frac{x_1 + x_2 + x_3 + x_4}{4}, \\ \bar{y} &= \frac{y_1 + y_2 + y_3 + y_4}{4}, \\ \bar{z} &= \frac{z_1 + z_2 + z_3 + z_4}{4}, \end{aligned} \quad (4.2.44)$$

and

$$V = \frac{1}{6} \det \begin{vmatrix} 1 & x_1 & y_1 & z_1 \\ 1 & x_2 & y_2 & z_2 \\ 1 & x_3 & y_3 & z_3 \\ 1 & x_4 & y_4 & z_4 \end{vmatrix}. \quad (4.2.45)$$

The 1-point rule approximates the integral using the value of κ at the center point of the tetrahedron. For the 2D case, the process is similar. The difference is that the integral has to be evaluated over a triangle. The basis function ϕ_i is equal to:

$$\phi_i(x, y) = a_i + b_i x + c_i y, \quad (4.2.46)$$

for $i, j = 1, 2, 3, 4$. The gradients will have the form:

$$\nabla \phi_i = \begin{bmatrix} b_i \\ c_i \end{bmatrix}. \quad (4.2.47)$$

Therefore,

$$\int_{T_k} \kappa \nabla \phi_i \cdot \nabla \phi_j dT_k = (\nabla \phi_i \cdot \nabla \phi_j) \int_{T_k} \kappa dT_k, \quad (4.2.48)$$

and the integral of κ can be estimated with the 1-point rule, as for the 3D case.

$$\int_{T_k} \kappa \simeq A * \kappa(\bar{x}, \bar{y}), \quad (4.2.49)$$

where

$$\begin{aligned} \bar{x} &= \frac{x_1 + x_2 + x_3}{3}, \\ \bar{y} &= \frac{y_1 + y_2 + y_3}{3}, \end{aligned} \quad (4.2.50)$$

and

$$A = \frac{1}{2} \det \begin{vmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{vmatrix}. \quad (4.2.51)$$

The Stiffness Matrix Algorithm for the 2D case is the following:

For each triangle in the domain

{

Generate the element matrix for each element

$$\begin{bmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{bmatrix}$$

Generate the coefficient matrix by the inverse of the element matrix

$$\begin{bmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{bmatrix}^{-1} = \begin{bmatrix} a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \\ c_1 & c_2 & c_3 \end{bmatrix}$$

Calculate the area

$$A = \frac{1}{2} \det \begin{vmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{vmatrix}$$

Evaluate $\kappa(\bar{x}, \bar{y})$

(4.2.52)

$$\bar{x} = \frac{x_1 + x_2 + x_3}{3}$$

$$\bar{y} = \frac{y_1 + y_2 + y_3}{3}$$

Calculate the element stiffness matrix

$$K_e = \begin{bmatrix} \nabla\phi_1\nabla\phi_1 & \nabla\phi_1\nabla\phi_2 & \nabla\phi_1\nabla\phi_3 \\ \nabla\phi_2\nabla\phi_1 & \nabla\phi_2\nabla\phi_2 & \nabla\phi_2\nabla\phi_3 \\ \nabla\phi_3\nabla\phi_1 & \nabla\phi_3\nabla\phi_2 & \nabla\phi_3\nabla\phi_3 \end{bmatrix} * A * \kappa(\bar{x}, \bar{y})$$

Where:

$$\begin{bmatrix} \nabla\phi_1\nabla\phi_1 & \nabla\phi_1\nabla\phi_2 & \nabla\phi_1\nabla\phi_3 \\ \nabla\phi_2\nabla\phi_1 & \nabla\phi_2\nabla\phi_2 & \nabla\phi_2\nabla\phi_3 \\ \nabla\phi_3\nabla\phi_1 & \nabla\phi_3\nabla\phi_2 & \nabla\phi_3\nabla\phi_3 \end{bmatrix} = \begin{bmatrix} b_1 & b_2 & b_3 \\ c_1 & c_2 & c_3 \end{bmatrix}^T * \begin{bmatrix} b_1 & b_2 & b_3 \\ c_1 & c_2 & c_3 \end{bmatrix}$$

Add the contributions to the global matrix

$K_e(i, j)$ for $i, j = 1, 2, 3 \rightarrow K(\text{node}_i, \text{node}_j)$ or

$K(\text{node}_i, \text{node}_j) + = K_e(i, j)$

}

4.2.5.4 Load Vector

To compute the load vector F in 3D the process is similar to that for the stiffness matrix:

$$F_i = \int_{T_1} f \phi_1 + \int_{T_2} f \phi_2 + \dots + \int_{T_n} f \phi_n, \quad (4.2.53)$$

where (using the 1-point integration rule):

$$\int_{T_k} f \phi_i \simeq V * f(\bar{x}, \bar{y}, \bar{z}) * \phi_i(\bar{x}, \bar{y}, \bar{z}). \quad (4.2.54)$$

and

$$\phi_i(\bar{x}, \bar{y}, \bar{z}) = 1/4. \quad (4.2.55)$$

Taking all of this into account the contribution in the load vector for each tetrahedron will be:

$$\int_{T_k} f \phi_i \simeq \frac{1}{4} V * f(\bar{x}, \bar{y}, \bar{z}). \quad (4.2.56)$$

For the 2D case the integral is evaluated with the following:

$$\int_{T_k} f \phi_i \simeq \frac{1}{3} A * f(\bar{x}, \bar{y}). \quad (4.2.57)$$

The load vector algorithm is the following:

For each triangle in the domain

{

Generate the element matrix for each element

$$\begin{bmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{bmatrix}$$

Calculate the area

$$A = \frac{1}{2} \det \begin{vmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{vmatrix} \quad (4.2.58)$$

Evaluate $\kappa(\bar{x}, \bar{y})$

$$\bar{x} = \frac{x_1 + x_2 + x_3}{3}$$

$$\bar{y} = \frac{y_1 + y_2 + y_3}{3}$$

Calculate the element load contribution

$$F_e = A/3 * f(\bar{x}, \bar{y})$$

Add the contributions to the global matrix

$$F_e(i) \text{ for } i, j = 1, 2, 3 \rightarrow F(\text{node}_i) \text{ or}$$

$$F(\text{node}_i) + = F_e(i)$$

}

4.2.6 Mass Matrix

The mass matrix is calculated from the following equation:

$$M_{ij} = \int_{\Omega} b\phi_i\phi_j d\Omega = \sum_{k=1}^{N_t} \int_{T_k} b\phi_i\phi_j dT_k, \quad (4.2.59)$$

The basis functions give a relationship of the areas of the elements in the 2D case. In the 3D case, it is a relationship of volumes. The value of ϕ_i is the relationship between the area of two triangles. One made by the points $(x_1, y_1)(x_2, y_2)(x_3, y_3)$, and one made using an extra point (x_e, y_e) substituting the point i for the extra point. For the case $i = 1$, the second triangle will be $(x_e, y_e)(x_2, y_2)(x_3, y_3)$. The value of ϕ_i will be the area of the triangle made by the points $(x_e, y_e)(x_2, y_2)(x_3, y_3)$ (A_e) over the area of the

original triangle (A_t),

$$\phi_1(x_e, y_e) = \frac{A_e}{A_t}, \quad (4.2.60)$$

or

$$\phi_1 = \frac{\frac{1}{2} \det \begin{vmatrix} 1 & x_e & y_e \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{vmatrix}}{\frac{1}{2} \det \begin{vmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{vmatrix}}. \quad (4.2.61)$$

For the 2D case we can build an element mass matrix. We will consider the element in the triangle of the following figure:

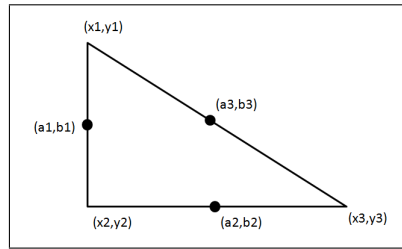


FIGURE 4.6: Triangle Element (2D case)

For the different points in the triangle we will have the following values:

	ϕ_1	ϕ_2	ϕ_3
(x_1, y_1)	1	0	0
(x_2, y_2)	0	1	0
(x_3, y_3)	0	0	1
(a_1, b_1)	1/2	1/2	0
(a_2, b_2)	0	1/2	1/2
(a_3, b_3)	1/2	0	1/2
(\bar{x}, \bar{y})	1/3	1/3	1/3

(4.2.62)

Using the three-point rule, and using the points $(a_1, b_1)(a_2, b_2)(a_3, b_3)$. We can build the values for the whole element:

$$\phi_1 \cdot \phi_1 = (\phi_1(a_1, b_1) \cdot \phi_1(a_1, b_1) + \phi_1(a_2, b_2) \cdot \phi_1(a_2, b_2) + \phi_1(a_3, b_3) \cdot \phi_1(a_3, b_3)) \cdot 1/3, \quad (4.2.63)$$

$$\phi_1 \cdot \phi_1 = 1/2 \cdot 1/3, \quad (4.2.64)$$

$$\phi_1 \cdot \phi_2 = (\phi_1(a_1, b_1) \cdot \phi_2(a_1, b_1) + \phi_1(a_2, b_2) \cdot \phi_2(a_2, b_2) + \phi_1(a_3, b_3) \cdot \phi_2(a_3, b_3)) \cdot 1/3, \quad (4.2.65)$$

$$\phi_1 \cdot \phi_2 = 1/4 \cdot 1/3. \quad (4.2.66)$$

From this we can build a matrix to store the interactions between the different basis functions in the element:

$$\begin{bmatrix} \phi_1 \cdot \phi_1 & \phi_1 \cdot \phi_2 & \phi_1 \cdot \phi_3 \\ \phi_2 \cdot \phi_1 & \phi_2 \cdot \phi_2 & \phi_2 \cdot \phi_3 \\ \phi_3 \cdot \phi_1 & \phi_3 \cdot \phi_2 & \phi_3 \cdot \phi_3 \end{bmatrix} = \begin{bmatrix} 1/2 & 1/4 & 1/4 \\ 1/4 & 1/2 & 1/4 \\ 1/4 & 1/4 & 1/2 \end{bmatrix} \cdot 1/3. \quad (4.2.67)$$

The 3D case is similar, but instead of area we will calculate the volume.

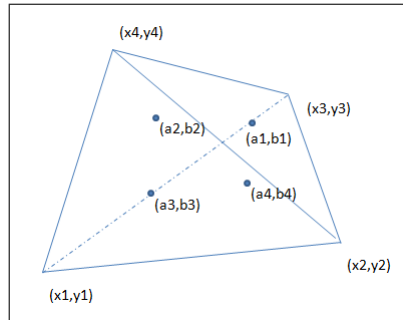


FIGURE 4.7: Tetrahedral Element (3D case)

The resulting matrix will have the following form:

$$\begin{bmatrix} \phi_1 \cdot \phi_1 & \phi_1 \cdot \phi_2 & \phi_1 \cdot \phi_3 & \phi_1 \cdot \phi_4 \\ \phi_2 \cdot \phi_1 & \phi_2 \cdot \phi_2 & \phi_2 \cdot \phi_3 & \phi_2 \cdot \phi_4 \\ \phi_3 \cdot \phi_1 & \phi_3 \cdot \phi_2 & \phi_3 \cdot \phi_3 & \phi_3 \cdot \phi_4 \\ \phi_4 \cdot \phi_1 & \phi_4 \cdot \phi_2 & \phi_4 \cdot \phi_3 & \phi_4 \cdot \phi_4 \end{bmatrix} = \begin{bmatrix} 3/9 & 2/9 & 2/9 & 2/9 \\ 2/9 & 3/9 & 2/9 & 2/9 \\ 2/9 & 2/9 & 3/9 & 2/9 \\ 2/9 & 2/9 & 2/9 & 3/9 \end{bmatrix} \cdot 1/4. \quad (4.2.68)$$

4.2.7 Inhomogeneous Dirichlet conditions

To deal with the Inhomogeneous Dirichlet conditions the Load Vector F becomes:

$$F_i = \int_{\Omega} f \phi_i d\Omega - \int_{\Omega} \kappa \nabla G \cdot \nabla \phi_i d\Omega, i = 1, 2, \dots, N, \quad (4.2.69)$$

The function G will be 0 in all the elements that do not have constrained nodes. In elements that have at least one constrained node, the calculation has the following form:

$$- \int_T \kappa \nabla G \nabla \phi_i dT = - \nabla G \nabla \phi_i \int_T \kappa dT. \quad (4.2.70)$$

For the 3D case

$$G = \sum_{i=1}^4 w_i \phi_i \implies \nabla G = \sum_{i=1}^4 w_i \nabla \phi_i, \quad (4.2.71)$$

and w_i are the nodal values $g(x)$ at the node i . The contributions to F_i will have the form

$$\int_T \kappa \nabla G \cdot \nabla \phi_i dT = V * \kappa(\bar{x}, \bar{y}, \bar{z}) * \nabla G \cdot \nabla \phi_i, \quad (4.2.72)$$

where $\nabla G \cdot \nabla \phi_i$ is calculated numerically as:

$$\nabla G \cdot \nabla \phi_i = \begin{bmatrix} b_1 & c_1 & d_1 \\ b_2 & c_2 & d_2 \\ b_3 & c_3 & d_3 \\ b_4 & c_4 & d_4 \end{bmatrix} \cdot \begin{bmatrix} b_1 & b_2 & b_3 & b_4 \\ c_1 & c_2 & c_3 & c_4 \\ d_1 & d_2 & d_3 & d_4 \end{bmatrix} \cdot \begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ w_4 \end{bmatrix}. \quad (4.2.73)$$

This will contribute to node i in the tetrahedron. For the 2D case, $\nabla G \nabla \phi_i$ is calculated as follows:

$$\nabla G \cdot \nabla \phi_i = \begin{bmatrix} b_1 & c_1 \\ b_2 & c_2 \\ b_3 & c_3 \end{bmatrix} \cdot \begin{bmatrix} b_1 & b_2 & b_3 \\ c_1 & c_2 & c_3 \end{bmatrix} \cdot \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix}. \quad (4.2.74)$$

In the 2D case the load vector algorithm is modified as follows to take into account the Dirichlet conditions:

For each triangle in the domain:

{

Generate the element matrix for each element,

$$\begin{bmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{bmatrix}.$$

Calculate the area,

$$A = \frac{1}{2} \det \begin{vmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{vmatrix}.$$

Evaluate $\kappa(\bar{x}, \bar{y})$,

$$\bar{x} = \frac{x_1 + x_2 + x_3}{3},$$

$$\bar{y} = \frac{y_1 + y_2 + y_3}{3}.$$

Calculate the element load contribution

$$F_e = A/3 * f(\bar{x}, \bar{y}).$$

Add the contributions to the global matrix,

$$F_e(i) \text{ for } i, j = 1, 2, 3 \rightarrow F(\text{node}_i) \text{ or}$$

$$F(\text{node}_i) + = F_e(i)$$

If any of the vertices of the triangle has a Dirichlet boundary condition then

{

Assemble the vector $w[i], i = 1, 2, 3$; where the values are the nodal values of the Dirichlet data and zero for the free

nodes (nodes that does not have a Dirichlet condition).

Generate the element matrix for the element,

$$\begin{bmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{bmatrix}.$$

Generate the coefficients matrix by the inverse of the element matrix,

$$\begin{bmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{bmatrix}^{-1} = \begin{bmatrix} a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \\ c_1 & c_2 & c_3 \end{bmatrix}.$$

Calculate the area,

$$A = \frac{1}{2} \det \begin{vmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{vmatrix}. \quad (4.2.75)$$

Evaluate $\kappa(\bar{x}, \bar{y})$,

$$\bar{x} = \frac{x_1 + x_2 + x_3}{3},$$

$$\bar{y} = \frac{y_1 + y_2 + y_3}{3}.$$

Calculate the contribution G to the load vector,

$$\begin{bmatrix} b_1 & b_2 & b_3 \\ c_1 & c_2 & c_3 \end{bmatrix}^T \cdot \begin{bmatrix} b_1 & b_2 & b_3 \\ c_1 & c_2 & c_3 \end{bmatrix} \cdot w \cdot A \cdot \kappa(\bar{x}, \bar{y}).$$

Subtract the contributions to the global load vector

$$- G(i) \text{ for } i, j = 1, 2, 3 \rightarrow F(\text{node}_i) \text{ or}$$

$$G(\text{node}_i) - = F_e(i)$$

}

}

4.2.8 Inhomogeneous Neumann conditions

To add the Inhomogeneous Neumann conditions the load vector F becomes

$$F_i = \int_{\Omega} f \phi_i - \int_{\Omega} c \nabla G \cdot \nabla \phi_i + \int_{\Gamma_2} h \phi_i, \quad i = 1, 2, \dots, N, \quad (4.2.76)$$

where

$$\int_{\Gamma_2} h \phi_i = \int_{T_1} h \phi_i + \int_{T_2} h \phi_i + \dots + \int_{T_n} h \phi_i. \quad (4.2.77)$$

where the T_i denotes the triangles of the boundary Γ_2 .

Where evaluated in each triangle with vertices (x_1, y_1, z_1) , (x_2, y_2, z_2) , and (x_3, y_3, z_3) , the value of the integral at each point becomes

$$\int h\phi_i = \frac{h_1 * \phi_i + h_2 * \phi_i + h_3 * \phi_i}{3} \cdot A. \quad (4.2.78)$$

The contributions at each node to F_i will be

$$F_{i+} = \frac{h \cdot A}{3}. \quad (4.2.79)$$

To calculate the area of the triangle, we calculate the normal vector \mathbf{n} , then we generate two temporal vectors in the following form

$$\begin{bmatrix} x_1 - x_2 \\ y_1 - y_2 \\ z_1 - z_2 \end{bmatrix}, \quad (4.2.80)$$

$$\begin{bmatrix} x_1 - x_3 \\ y_1 - y_3 \\ z_1 - z_3 \end{bmatrix}. \quad (4.2.81)$$

Then we perform the cross product of the vectors 4.2.80 and 4.2.81, and calculate the Euclidean norm ($\|x\| = \sqrt{x^2 + y^2 + z^2}$) of the vector. The area is then

$$A = \frac{\|x\|}{2}. \quad (4.2.82)$$

To calculate h we need the gradient of u , and the normal vector. To calculate the normal vector, we perform the cross product of the vectors 4.2.80 and 4.2.81, and normalize it. For each point in the mesh, we consider each triangle that has the node as a vertex, and then average of all normals on those triangles; this will be the node normal,

$$n = \begin{bmatrix} n_x \\ n_y \\ n_z \end{bmatrix} \frac{1}{\|x\|}. \quad (4.2.83)$$

The gradient vector will have the following form:

$$\nabla u = \begin{bmatrix} \frac{\partial u}{\partial x} \\ \frac{\partial u}{\partial y} \\ \frac{\partial u}{\partial z} \end{bmatrix}. \quad (4.2.84)$$

We have to evaluate numerically the functions at each point. The value of h will be the following:

$$h_i = \kappa \nabla u_i \cdot n_i. \quad (4.2.85)$$

The algorithm for the contributions in 3D to the load vector are the following:

For each 3D triangle surface with Neuman Boundary Conditions

{

Add the Neumann data of the points and divide it by 3

$$hval = \frac{Neu(x_1) + Neu(x_2) + Neu(x_3)}{3}$$

Calculate the area by the following procedure:

$$v_1 = \begin{bmatrix} x_1 - x_2 \\ y_1 - y_2 \\ z_1 - z_2 \end{bmatrix}, v_2 = \begin{bmatrix} x_1 - x_3 \\ y_1 - y_3 \\ z_1 - z_3 \end{bmatrix},$$

$$n_v = \frac{v_1 \times v_2}{\|v_1 \times v_2\|}, \quad (4.2.86)$$

$$A = \frac{\|n_v\|}{2}.$$

Calculate the contribution

$$N_c = \frac{hval \cdot A}{3}$$

Add the contribution to the load vector,

$$F(x_1, y_1, z_1) + = N_c,$$

$$F(x_2, y_2, z_2) + = N_c,$$

$$F(x_3, y_3, z_3) + = N_c.$$

}

4.2.9 Analytical Example

Hence follows a test to show that the code works correctly, following the example in [32]. In these examples the value of N means the quantity of squares by which the domain will be divided. Also the graphs are the values in each of the nodes; this means the node (axis x), the value (axis y). For the following system:

$$\begin{aligned} -\nabla(\kappa \nabla u) &= f \text{ in } \Omega, \\ u &= g \text{ on } \Gamma_1, \\ (\kappa \nabla u) \cdot n &= h \text{ on } \Gamma_2. \end{aligned} \quad (4.2.87)$$

The domain is a unit square with the following boundary conditions. The boundary with Dirichlet boundary conditions is Γ_1 , the boundary for the Neumann conditions is Γ_2 .

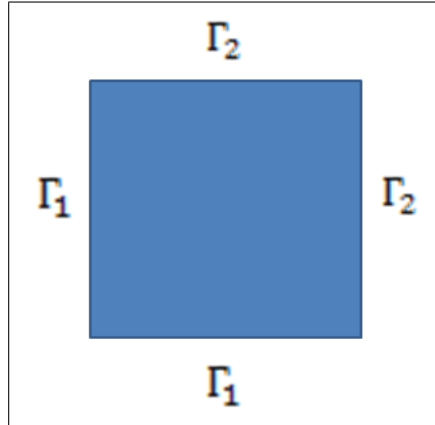
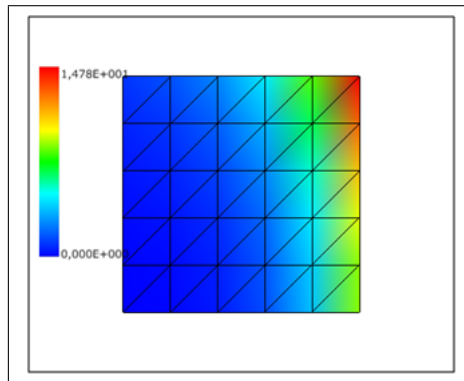


FIGURE 4.8: Boundary conditions for the analytical problem.

We create the data with the following equations:

$$\begin{aligned}
 \kappa &= 1 + x^2y \\
 -\nabla(\kappa\nabla u) &= -2xy(2e^{2x}(x^2 + y^2) + 2e^{2x}x) \\
 - (1 + x^2y)(4e^{2x}(x^2 + y^2) + 8e^{2x}x + 2e^{2x}) - 2x^2e^{2x}y - 2(1 + x^2y)e^{2x} & \quad \text{in } \Omega, \\
 u &= e^{2x}(x^2 + y^2) & \quad \text{on } \Gamma_1, \\
 (\kappa\nabla u)n &= \begin{bmatrix} 2e^{2x}(x^2 + y^2) + 2e^{2x}x \\ 2e^{2x}y \end{bmatrix} (1 + x^2y)n & \quad \text{on } \Gamma_2.
 \end{aligned}
 \tag{4.2.88}$$

The analytic solution is $u = e^{2x}(x^2 + y^2)$.



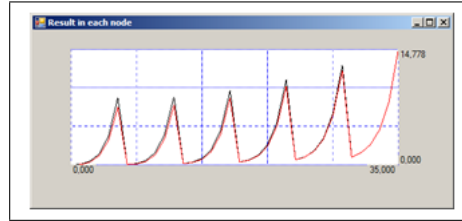


FIGURE 4.9: Results $N=5$. Above we can see the comparison between the calculated solution (black) with the exact solution (red). In this graph the x axis is node n , and the y axis the value of the potential. It is a representation of the 3D potential in 2D.

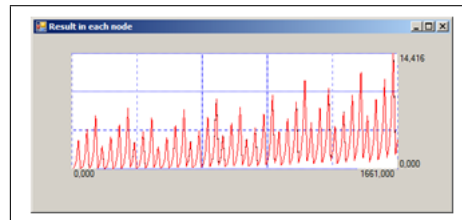
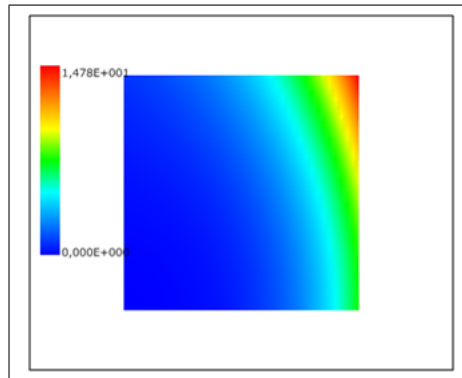


FIGURE 4.10: Results $N=40$. Above we can see the comparison between the calculated solution (black) with the exact solution (red).

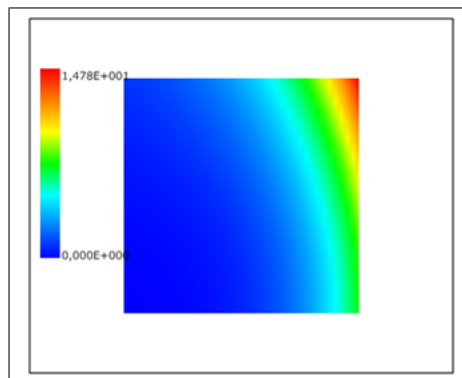


FIGURE 4.11: Exact Solution for $N=40$.

4.2.10 Electrostatic Example

This code is an example from [33]. The square has Dirichlet boundary conditions in the 4 sides.

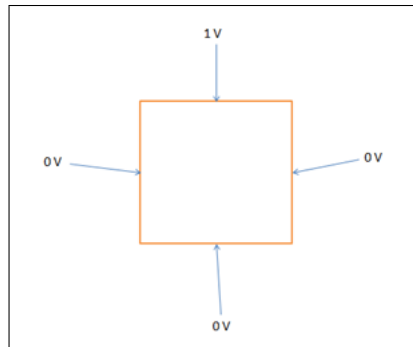


FIGURE 4.12: Boundary conditions for the electrostatic problem.

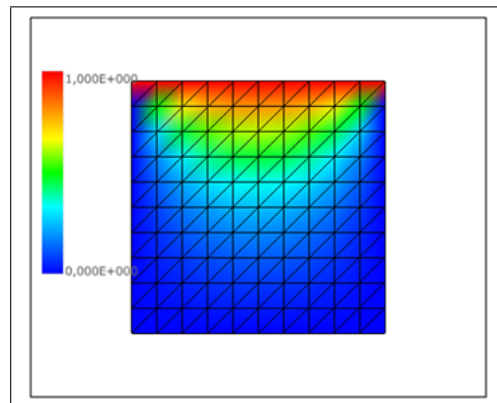
The value ϵ is equal to 1.

$$\begin{aligned} -\nabla(\epsilon\nabla u) &= 0 && \text{in } \Omega, \\ u &= g && \text{on } \partial\Omega. \end{aligned} \tag{4.2.89}$$

The analytical solution is:

$$u(x, y) = \frac{4V_0}{\pi} \sum_{k=1}^{\infty} \frac{\sin\frac{(2k-1)\pi x}{width} \sinh\frac{(2k-1)\pi y}{width}}{(2k-1)\sinh\frac{(2k-1)\pi x}{width}}, \tag{4.2.90}$$

and $height = width = 1$.



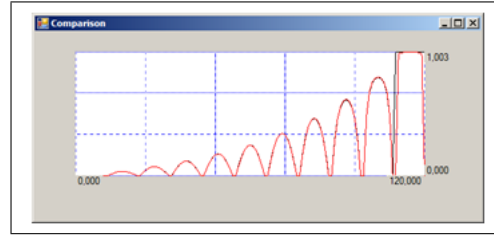


FIGURE 4.13: Results $N=5$. In the right we can see the comparison between the calculated solution (black) with the exact solution (red).

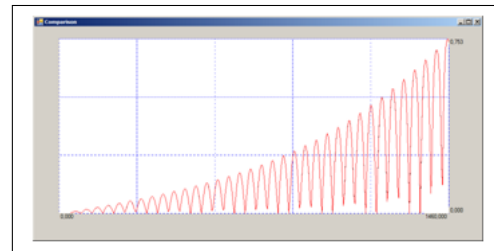
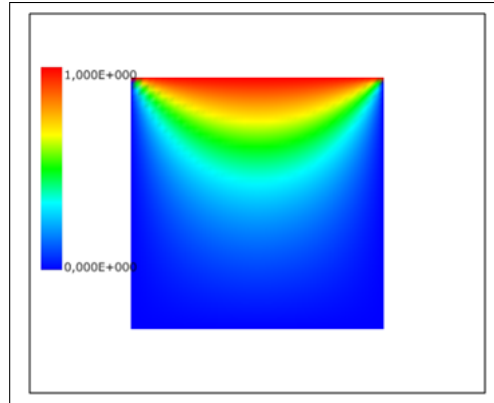


FIGURE 4.14: Results $N=40$. In the right we can see the comparison between the calculated solution (black) with the exact solution (red).

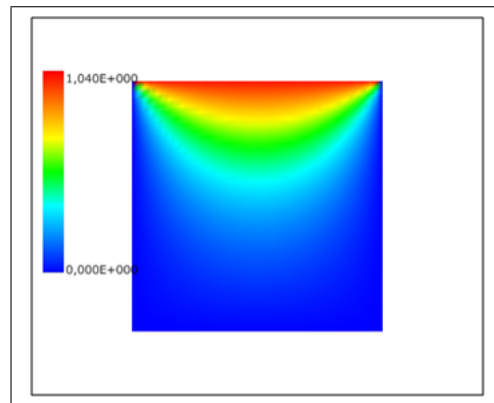


FIGURE 4.15: Exact solution for $N=40$.

4.2.11 Boundary Element Method

Another option to create and solve the inverse problem in ECG is to use the Boundary Element Method (BEM) to create the operators. The BEM only requires the discretization of the surfaces of heart and thorax; which is an advantage from the FEM where is necessary the whole volume. The problem is that the generated matrix will have almost full entries. Another problem of the BEM is the implementation in comparison with the FEM is very complicated. For example in Figure 4.16 we can see the necessary surfaces for the direct and inverse problem.

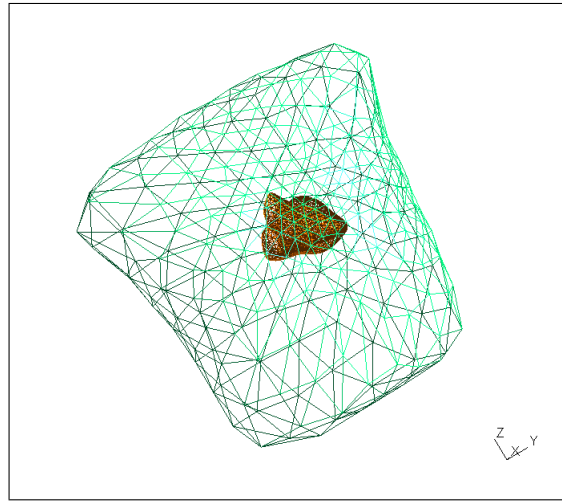


FIGURE 4.16: Example of the heart thorax and lungs for the BEM.

4.2.11.1 Boundary Element Method for Laplace in 3D

If we take the Laplace equation system:

$$\begin{aligned} -\nabla(M_T(x)\nabla u) &= 0, & x \in \Omega, \\ u &= u_e, & x \text{ on } \partial\Omega_h, \\ (M_T(x)\nabla u \cdot n) &= 0, & x \text{ on } \Sigma. \end{aligned} \quad (4.2.91)$$

For a volume contained by a surface $S = S_1 \cup S_2$, where $\Sigma = S_1$ and $\partial\Omega_h = S_2$. Using a weighted integral, and applying the Green's first identity, now we have the following formulation, which is the same as the one used for the FEM.

$$0 = - \int_{\Omega} \nabla \cdot (M_T(x)\nabla u) w d\Omega = \int_{\Omega} (M_T(x)\nabla u) \nabla w d\Omega, - \int_{\partial\Omega} w (M_T(x)\nabla u) n d\partial\Omega. \quad (4.2.92)$$

Applying the Green's first identity again, and if $M_T(x)$ is a constant the second term becomes

$$\int_{\Omega} \nabla u(M_T(x) \nabla w) d\Omega = \int_{\Omega} \nabla [u(M_T(x) \nabla w)] d\Omega - \int_{\Omega} u \nabla (M_T(x) \nabla w) d\Omega. \quad (4.2.93)$$

and equation 4.2.92 becomes

$$- \int_{\Omega} u \nabla (M_T(x) \nabla w) d\Omega = \int_{\partial\Omega} u(M_T(x) \nabla w) n d\partial\Omega - \int_{\partial\Omega} w(M_T(x) \nabla u) n d\partial\Omega, \quad (4.2.94)$$

or for the cases of two non-overlapping surfaces (as in the case of heart and the thorax):

$$- \int_{\Omega} u \nabla (M_T(x) \nabla w) d\Omega = \int_{\Sigma} u(M_T(x) \nabla w) \cdot n d\Sigma - \int_{\partial\Omega_h} w(M_T(x) \nabla u) \cdot n d\partial\Omega_h. \quad (4.2.95)$$

This is known as the boundary integral equation (BIE).

4.2.11.2 Implementation

The following implementation is from [34] and [35]. The boundary integral equation is, evaluated at a point (ξ, η, ζ) :

$$\gamma u(\xi, \eta, \zeta) = \int_{\partial\Omega} (u \cdot \nabla w \cdot n - M_T(x) \nabla u \cdot n \cdot w) d\partial\Omega. \quad (4.2.96)$$

where

$$\gamma = \begin{cases} 1 & \text{if } (\xi, \eta, \zeta) \in \Omega \\ \frac{1}{2} & \text{if } (\xi, \eta, \zeta) \in \partial\Omega \\ 0 & \text{if } (\xi, \eta, \zeta) \notin \partial\Omega \cup \Omega \end{cases}. \quad (4.2.97)$$

The boundary $\partial\Omega$ is discretized using N triangles (Tr), so from 4.2.96 we can change into:

$$\gamma u(\xi, \eta, \zeta) = \sum_{k=0}^N (u_k \int_{Tr_k} (M_T(x) \nabla w \cdot n) dTr_k - (M_T(x) \nabla u_k \cdot n) \int_{Tr_k} w dTr_k). \quad (4.2.98)$$

Because we want to know the potential in the boundary $\partial\Omega$, then we change the formula into:

$$\frac{1}{2} u_i(\xi, \eta, \zeta) = \sum_{k=0}^N ((u_k(\xi, \eta, \zeta) \int_{Tr_k} (M_T(x) \nabla w \cdot n) dTr_k - (M_T(x) \nabla u_k(\xi, \eta, \zeta) \cdot n) \int_{Tr_k} w dTr_k). \quad (4.2.99)$$

where w , and $(\nabla w \cdot n)$ are equal to:

$$\omega = -\frac{1}{4\pi\sqrt{(x-\xi)^2 + (y-\eta)^2 + (z-\zeta)^2}} \quad (4.2.100)$$

$$(4.2.101)$$

$$(\nabla w \cdot n) = \frac{((x-\xi) * n_x + (y-\eta) * n_y + (z-\zeta) * n_z)}{4\pi((x-\xi)^2 + (y-\eta)^2 + (z-\zeta)^2)^{\frac{3}{2}}} \quad (4.2.102)$$

4.2.11.3 Explicit Expressions for 3D in integrals in potential theory

This implementation is more complicated as the one explained above, but the results for the forward and inverse problem are better. The complete explanation can be found in [36–39] to calculate the Boundary Integral Equation BIE, using explicit expressions, I will omit the conductivity M_T , considering is a constant;

$$0 = \lim_{x_\epsilon \rightarrow x \in \partial\Omega} \left(\int_{\partial\Omega} w(x_\epsilon, y) (\nabla u(y) \cdot n) d\partial\Omega - \int_{\partial\Omega} u(y) \nabla w(x_\epsilon, y) \cdot n \right), \quad (4.2.103)$$

x is called the source point and y the field point. We will declare:

$$\begin{aligned} w(x_\epsilon, y) &= \frac{1}{\|x - y\|}, \\ (\nabla w(x_\epsilon, y)) &= \frac{1}{4\pi} \frac{x - y}{\|x - y\|^3}, \\ u(y) &= \sum_{j=1}^N u(y_j) \psi_j(y), \\ \nabla u(y) \cdot n &= \sum_{j=1}^N \nabla \cdot u(y_j) \cdot n \psi_j(y). \end{aligned} \quad (4.2.104)$$

From this we can declare the matrix vector system as

$$G \nabla u(y_j) \cdot n = H u(y_j), \quad (4.2.105)$$

where the matrix G , and H are given by:

$$\begin{aligned} G_{ij} &= \sum_{i=1}^{N_e} \lim_{x_\epsilon \rightarrow x_i} \int_{Tr_i} w(x_\epsilon, y_j) \psi(y_j) dTr_i, \\ H_{ij} &= \sum_{i=1}^{N_e} \lim_{x_\epsilon \rightarrow x_i} \int_{Tr_i} \nabla w(x_\epsilon, y_j) \cdot n \psi(y_j) dTr_i, \end{aligned} \quad (4.2.106)$$

where N_e is the number of elements with support ψ_j . We will have contributions to each element i from the support j . We will define:

$$G_j^i(x) = \int_{Tr_i} w(x, y) \psi(y_j) dTr_i, \quad (4.2.107)$$

$$H_j^i(x) = \int_{Tr_i} \nabla w(x, y) \cdot n \psi(y_j) dTr_i. \quad (4.2.108)$$

This single layer and double layer potentials can be decomposed in generic integrals:

$$G_j^i(x) = \frac{1}{4\pi} [a_j I_1^{\xi_e} + b_j I_1^{\zeta_e} + c_j I_3], \quad (4.2.109)$$

$$H_j^i(x) = \frac{-\eta_e}{4\pi} [a_j I_3^{\xi_e} + b_j I_3^{\zeta_e} + c_j I_3]. \quad (4.2.110)$$

$$(4.2.111)$$

The generic integrals are given by:

$$\begin{aligned} I_1 &= \int_{E_i} \frac{1}{r} & I_1^{\xi_e} &= \int_{E_i} \frac{\xi_e}{r} & I_1^{\zeta_e} &= \int_{E_i} \frac{\zeta_e}{r} \\ I_3 &= \int_{E_i} \frac{1}{r^3} & I_3^{\xi_e} &= \int_{E_i} \frac{\xi_e}{r^3} & I_3^{\zeta_e} &= \int_{E_i} \frac{\zeta_e}{r^3} \end{aligned} \quad (4.2.112)$$

The values ξ_e, η_e, ζ_e are given by

$$r = y - x = \xi_e e_1 + \zeta_e e_2 + \eta_e e_3, \quad (4.2.113)$$

where e_1, e_2, e_3 are the vectors that form the orthonormal basis companion of each triangle such that e_1 is parallel to the line created by the point 1 and 2 of the triangle and e_3 is the outward normal vector to the triangle. For the procedure to calculate exactly the integrals, please refer to the references.

4.2.12 Example of Boundary Element Method

We will have the following system;

$$-\nabla(\kappa \nabla u) = 0 \text{ in } \Omega, \quad (4.2.114)$$

$$u = g \text{ on } \Gamma_1, \quad (4.2.115)$$

$$\kappa \nabla u n = h \text{ on } \Gamma_2, \quad (4.2.116)$$

with:

$$u = e^x \sin(z) + e^z \cos(y), \quad (4.2.117)$$

$$\kappa = 100, \quad (4.2.118)$$

$$\nabla u = \begin{bmatrix} e^x \sin(z) \\ -e^z \sin(y) \\ e^x \cos(z) + e^z \cos(y) \end{bmatrix}. \quad (4.2.119)$$

The geometry will be the following:

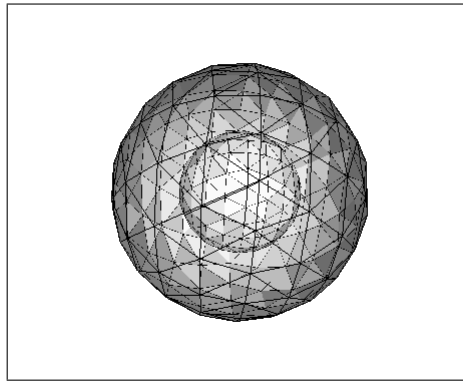


FIGURE 4.17: Used geometry for BEM example.

where the inner sphere will be Γ_1 , and the outer sphere Γ_2 . A graph of the comparison between the calculated and the analytical solution in each node is displayed next:

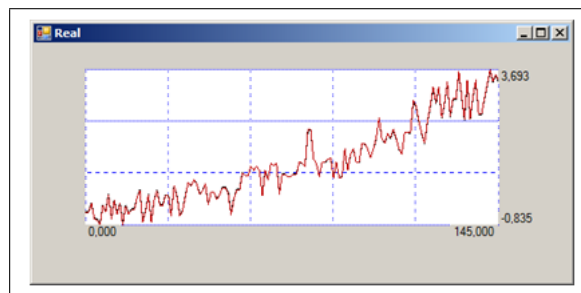


FIGURE 4.18: Comparison between analytical (black) and calculated (red) solution in each node.

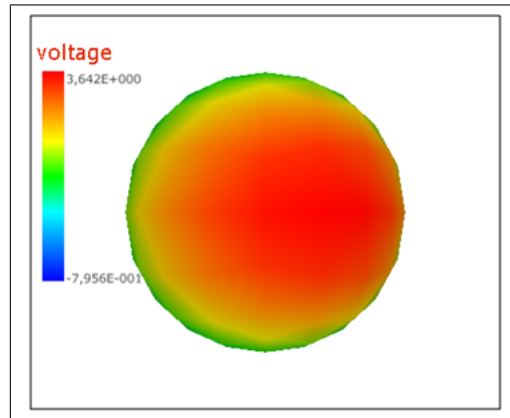


FIGURE 4.19: 3D solution for the system using BEM.

The same system is solved using the finite element method (FEM).

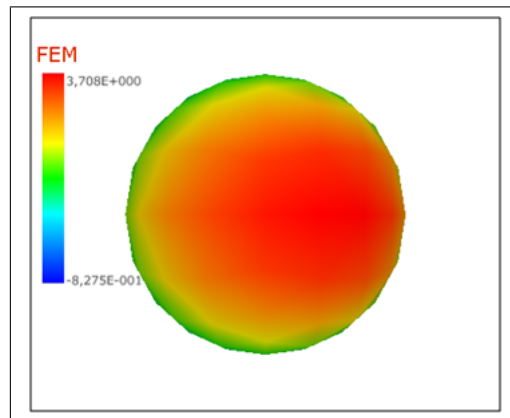


FIGURE 4.20: 3D solution for the system using FEM.

Chapter 5

Chapter 5. Regularization

5.1 Inverse problem and regularization

In this section, we prove the existence of the solution to the inverse problem.

We can define the following bounded linear operator:

$$\Theta : H^{1/2}(\partial\Omega_H) \rightarrow H^1(\Omega) \text{ such that } \Theta(u_e) = \mathbf{u}.$$

Observe that by the uniqueness of the solution \mathbf{u} to the direct problem, the operator Θ is well-defined.

Next we define Θ_H the operator mapping the electrical potential heart surface u_e to the electrical potential thorax surface \bar{u}

$$\Theta_H : H^{1/2}(\partial\Omega_H) \rightarrow H^1(\Sigma) \text{ such that } \Theta_H(u_e) = \bar{u} \text{ on } \Sigma.$$

Following the Hilbert Uniqueness Method (HUM) developed by Lions [40], we introduce the adjoint operator:

$$\Theta_H^* : H^{1/2}(\Sigma) \rightarrow H^{1/2}(\partial\Omega_H) \text{ such that } \Theta_H^*(g) = (\mathbf{M}_T(x)\nabla p_T) \cdot \mathbf{n}_T.$$

$$\left\{ \begin{array}{l} -\nabla \cdot (\mathbf{M}_k(x)\nabla p_k) = 0, \quad x \in \Omega_k, \\ p_L = p_T \text{ and } \mathbf{M}_L(x)\nabla p_L \cdot \mathbf{n}_L = \mathbf{M}_T(x)\nabla p_T \cdot \mathbf{n}_L \text{ on } \partial\Omega_L, \\ p_R = p_T \text{ and } \mathbf{M}_R(x)\nabla p_R \cdot \mathbf{n}_R = \mathbf{M}_T(x)\nabla p_T \cdot \mathbf{n}_R \text{ on } \partial\Omega_R, \\ (\mathbf{M}_T(x)\nabla p_T) \cdot \mathbf{n}_T = g \text{ on } \Sigma, \\ p_T = 0 \text{ on } \partial\Omega_h, \end{array} \right. \quad (5.1.1)$$

For g given we define the operator $\Lambda : H^{1/2}(\Sigma) \rightarrow H^{1/2}(\Sigma)$

$$\left\{ \begin{array}{l} -\nabla \cdot (\mathbf{M}_k(x) \nabla u_k) = 0, \quad x \in \Omega_k, \\ u_L = u_T \text{ and } \mathbf{M}_L(x) \nabla u_L \cdot \mathbf{n}_L = \mathbf{M}_T(x) \nabla u_T \cdot \mathbf{n}_L \text{ on } \partial\Omega_L, \\ u_R = u_T \text{ and } \mathbf{M}_R(x) \nabla u_R \cdot \mathbf{n}_R = \mathbf{M}_T(x) \nabla u_T \cdot \mathbf{n}_R \text{ on } \partial\Omega_R, \\ (\mathbf{M}_T(x) \nabla u_T) \cdot \mathbf{n}_T = 0 \text{ on } \Sigma, \\ u_T = (\mathbf{M}_T(x) \nabla p_T) \cdot \mathbf{n}_T \text{ on } \partial\Omega_h, \end{array} \right. \quad (5.1.2)$$

$$\Lambda(g) = u_T \text{ on } \Sigma.$$

Multiplying the first equation in 5.1.2 by p_k solution of 5.1.1, integrating over Ω_k , and summing over k one gets:

$$\int_{\Sigma} u_T(x) g(x) dx = \int_{\partial\Omega_h} (\mathbf{M}_T(x) \nabla p_T \cdot \mathbf{n}_T)^2 dx.$$

Thus the semi-norm induced by the operator Λ

$$\langle \Lambda g, g \rangle = \int_{\partial\Omega_h} (\mathbf{M}_T(x) \nabla p_T \cdot \mathbf{n}_T)^2 dx$$

is in fact a norm (Note that $\Lambda = \Theta_H \Theta_H^*$).

We can now introduce a Hilbert space G as the completion of $H^{1/2}(\Sigma)$ with respect to the semi-norm induced by Λ . Then if $\bar{u} \in G'$, there exist a unique $g \in G$ s.t. $\Lambda(g) = \bar{u}$, and the inverse problem admits a unique solution (Λ is an isomorphism from G to its dual space G').

Since G' is an unknown space it's difficult to deal with data in this space. Thus we choose to work in more common spaces for the inverse problem. Namely data in $H^{1/2}(\Sigma)$ or in $L^2(\Sigma)$.

In the following, we show the ill-posedness of the inverse problem by showing that the operator Θ_H is compact and then the inverse operator Θ_H^{-1} is an unbounded operator. The following theorem is a consequence of the compactness of the operator Θ_H (see [40] for e.g.): The inverse problem is ill-posed.

$$\begin{aligned} -\nabla \cdot (\mathbf{M}_s \nabla u_s) &= 0 \in \Omega_B, \\ (\mathbf{M}_s \nabla u_s) \cdot \mathbf{n} &= 0 \text{ on } \partial\Omega_B - \partial\Omega_H, \\ u_s &= u_t \text{ on } \partial\Omega_B - \partial\Omega_H. \end{aligned} \quad (5.1.3)$$

Since the inverse problem is ill-posed, the inverse problem (11) is usually solved by regularization methods. We introduce the regularized cost function: $J_{\lambda}(u_e) = \frac{1}{2} \|u_T - \bar{u}\|_{L^2(\Sigma)}^2 + \frac{\lambda}{2} \|u_e\|_{L^2(\partial\Omega_h)}^2$ and look for its minimum over u_e (recall that in the direct

problem (5.1.3), $u_T = u_e$ on $\partial\Omega_h$). Using the Lagrangian method we can show that the optimality system is given by:

$$\left\{ \begin{array}{l} -\nabla \cdot (\mathbf{M}_k(x)\nabla p_k) = 0, \quad x \in \Omega_k, \\ p_L = p_T \text{ and } \mathbf{M}_L(x)\nabla p_L \cdot \mathbf{n}_L = \mathbf{M}_T(x)\nabla p_T \cdot \mathbf{n}_L \text{ on } \partial\Omega_L, \\ p_R = p_T \text{ and } \mathbf{M}_R(x)\nabla p_R \cdot \mathbf{n}_R = \mathbf{M}_T(x)\nabla p_T \cdot \mathbf{n}_R \text{ on } \partial\Omega_R, \\ (\mathbf{M}_T(x)\nabla p_T) \cdot \mathbf{n}_T = u_T - \bar{u} \text{ on } \Sigma, \\ p_T = 0 \text{ on } \partial\Omega_h, \end{array} \right. \quad (5.1.4)$$

$$\left\{ \begin{array}{l} -\nabla \cdot (\mathbf{M}_k(x)\nabla u_k) = 0, \quad x \in \Omega_k, \\ u_L = u_T \text{ and } \mathbf{M}_L(x)\nabla u_L \cdot \mathbf{n}_L = \mathbf{M}_T(x)\nabla u_T \cdot \mathbf{n}_L \text{ on } \partial\Omega_L, \\ u_R = u_T \text{ and } \mathbf{M}_R(x)\nabla u_R \cdot \mathbf{n}_R = \mathbf{M}_T(x)\nabla u_T \cdot \mathbf{n}_R \text{ on } \partial\Omega_R, \\ (\mathbf{M}_T(x)\nabla u_T) \cdot \mathbf{n}_T = 0 \text{ on } \Sigma, \\ u_T = \frac{1}{\lambda}(\mathbf{M}_T(x)\nabla p_T) \cdot \mathbf{n}_T \text{ on } \partial\Omega_h, \end{array} \right. \quad (5.1.5)$$

Note that the solution of this optimality system is equivalent to

$$u_e^\lambda = (\Theta_H^* \Theta_H + \lambda I)^{-1} \Theta_H^* \bar{u}, \quad (5.1.6)$$

which is the Tikhonov regularization introduced originally in [41].

5.1.1 Create the Operators

As mentioned in the introduction the inverse problem is to reconstruct the *image* from a set of measures with a linear or non-linear operator. In this chapter we explain the creation of the operators in the case the heart is considered with a null flux conductivity and model by the Laplace equation for volume conductor model in 3D. The operators are created consider a quasi-static relationship; this means that the change in time is not important from measure to measure, that we can consider each time step as static. The operators for the Laplace equation will be created with the theory explained of finite element method and boundary element method. The creation of the operators is similar as explained and used for the inverse problem in [13, 28, 31].

5.1.1.1 Matrix Form FEM

If we take the Laplace equation system:

$$\begin{aligned} -\nabla \cdot (\kappa \nabla u) &= 0 && \in \Omega_T, \\ u &= g(x) && \text{on } \partial\Omega_H, \\ (\kappa \nabla u) \cdot \mathbf{n} &= h(x) && \text{on } \Sigma, \end{aligned} \quad (5.1.7)$$

for the geometry in Figure 5.1.

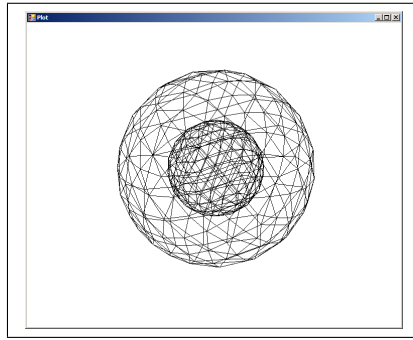


FIGURE 5.1: Tetrahedra Element (3D case)

Where the nodes in the outer sphere will be indicated with sub-index t , the nodes in the inner sphere h , and the nodes in between v . This equivalent if we are using the heart - thorax geometry. The nodes created from the discretization in 3D (tetrahedra). We can build the matrix-vector system in the following form. First we calculate the Stiffness Matrix; we calculate the stiffness matrix which entries are equal to

$$K_{ij} = \int_{\Omega} \kappa \nabla \phi_i \nabla \phi_j d\Omega \quad i, j = 1, 2, \dots, N(\text{number of nodes}). \quad (5.1.8)$$

The resulting matrix vector equation will be:

$$\begin{bmatrix} K_{hh} & K_{hv} & K_{ht} \\ K_{vh} & K_{vv} & K_{vt} \\ K_{th} & K_{tv} & K_{tt} \end{bmatrix} \begin{bmatrix} U_h \\ U_v \\ U_t \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}. \quad (5.1.9)$$

The vectors with the nodal values of the potential are U_h, U_v, U_t for the inner sphere, volume, and outer sphere respectively. Then we apply the Dirichlet condition; we consider U_h is given, and considering no overlapping between the surfaces of inner and outer

sphere the system becomes

$$\begin{bmatrix} K_{vh} & K_{vv} & K_{vt} \\ 0 & K_{tv} & K_{tt} \end{bmatrix} \begin{bmatrix} U_h \\ U_v \\ U_t \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad (5.1.10)$$

or

$$\begin{bmatrix} K_{vv} & K_{vt} \\ K_{tv} & K_{tt} \end{bmatrix} \begin{bmatrix} U_v \\ U_t \end{bmatrix} = \begin{bmatrix} -K_{vh}U_h \\ 0 \end{bmatrix}. \quad (5.1.11)$$

Next, we apply the Neumann condition. For each triangle in the outer surface (where the Neumann condition is applied), we calculate the following coefficient;

$$Neumann_c = (A)/3.0. \quad (5.1.12)$$

Then, we apply this contributions to each of the nodes of the triangle in a vector the size of the nodes of the outer sphere. This vector we will call it N_v . The contributions will be in the form $N_{v_i+} = Neumann_{c_i}$. Creating this vector the global Laplace matrix-equation will be;

$$\begin{bmatrix} K_{vv} & K_{vt} \\ K_{tv} & K_{tt} \end{bmatrix} \begin{bmatrix} U_v \\ U_t \end{bmatrix} = \begin{bmatrix} -K_{vh}U_h \\ N_v^t dU_t \end{bmatrix}, \quad (5.1.13)$$

where dU_t is a vector with the nodal values of the flux in the normal direction.

5.1.1.2 Operator Form FEM

If the values in the volume in between are not from our interest; from the equation [5.1.13](#) we can build a direct relationship between the potentials in the two surfaces:

$$\begin{aligned} K_{vv}U_v + K_{vt}U_t &= -K_{vh}U_h, \\ K_{tv}U_v + K_{tt}U_t &= N_v^t dU_t. \end{aligned} \quad (5.1.14)$$

Solving for U_v

$$U_v = -K_{vv}^{-1}(K_{vh}U_h + K_{vt}U_t), \quad (5.1.15)$$

and substituting in [5.1.14](#), we will have the following form;

$$-K_{tv}K_{vv}^{-1}K_{vh}U_h - K_{tv}K_{vv}^{-1}K_{vt}U_t + K_{tt}U_t = N_v^t dU_t, \quad (5.1.16)$$

or

$$(K_{tt} - K_{tv}K_{vv}^{-1}K_{vt})U_t = N_v^t dU_t + K_{tv}K_{vv}^{-1}K_{vh}U_h. \quad (5.1.17)$$

We will define the operators P , and Q as:

$$\begin{aligned} P &= (K_{tt} - K_{tv}K_{vv}^{-1}K_{vt})^{-1}K_{tv}K_{vv}^{-1}K_{vh}, \\ Q &= (K_{tt} - K_{tv}K_{vv}^{-1}K_{vt})^{-1}N_v^t. \end{aligned} \quad (5.1.18)$$

Then substituting in 5.1.14 the system by operators will be

$$U_t = PU_h + QdU_t. \quad (5.1.19)$$

5.1.1.3 Example Finite Element Method Using Operators

We will have the following system;

$$\begin{aligned} -\nabla \cdot (\kappa \nabla u) &= f(x) && \in \Omega_T, \\ u &= g(x) && \text{on } \partial\Omega_H, \\ (\kappa \nabla u) \cdot \mathbf{n} &= h(x) && \text{on } \Sigma. \end{aligned} \quad (5.1.20)$$

With:

$$u(x) = e^x \sin(z) + e^z \cos(y), \quad (5.1.21)$$

$$\kappa = 100, \quad (5.1.22)$$

$$\nabla u = \begin{bmatrix} e^x \sin(z) \\ -e^z \sin(y) \\ e^x \cos(z) + e^z \cos(y) \end{bmatrix}. \quad (5.1.23)$$

The used geometry is the following, where Γ_1 is the heart's surface, and Γ_2 the thorax surface:

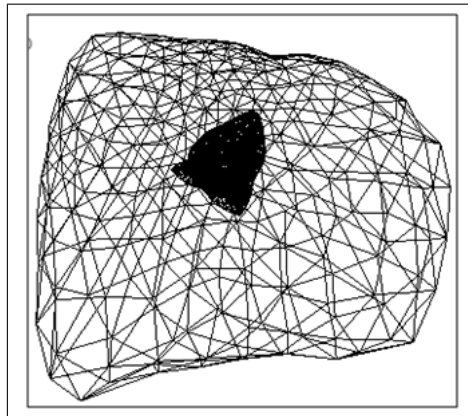


FIGURE 5.2: Heart - Thorax Geometry.

The comparison between the analytical solution, and the calculated are shown in the following figures:

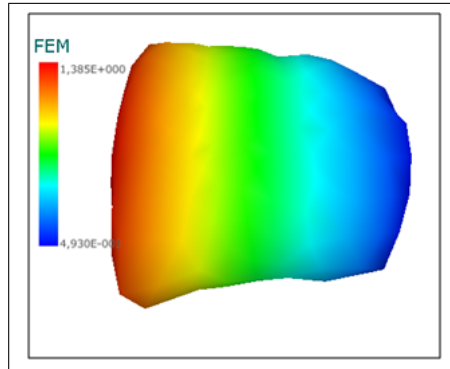


FIGURE 5.3: Calculated solution.

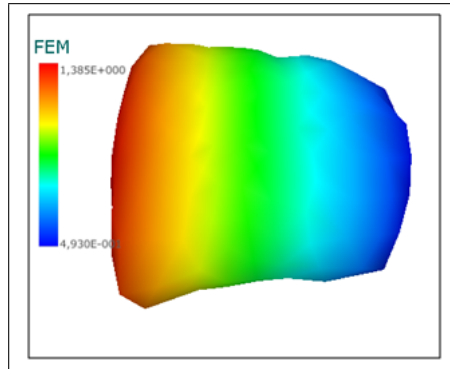


FIGURE 5.4: Analytical solution.

5.1.1.4 Operator Form BEM

Using BEM

$$\frac{1}{2}u_i(\xi, \eta, \zeta) = \sum_{k=0}^N ((u_k(\xi, \eta, \zeta) \int_{Tr_k} (\kappa \nabla w n) dTr_k - (\kappa \nabla u_k(\xi, \eta, \zeta) n) \int_{Tr_k} w dTr_k). \quad (5.1.24)$$

We can readjust the terms into:

$$\sum_{k=1}^N A_{ik} \cdot x_i = \sum_{k=1}^N B_{ik} \quad (5.1.25)$$

where N is the number of triangles:

$$A_{ik} = \begin{cases} -\int_{Tr_k} w dTr_k & \text{if } u_i \text{ is given} \\ \int_{Tr_k} (\kappa \nabla w n) dTr_k & \text{if } (\kappa \nabla u_i n) \text{ is given and } i \neq k \\ \int_{Tr_k} (\kappa \nabla w n) dTr_k - \frac{1}{2} & \text{if } (\kappa \nabla u_i n) \text{ is given and } i = k \end{cases} \quad (5.1.26)$$

$$B_{ik} = \begin{cases} u_i (-\int_{Tr_k} (\kappa \nabla w n) dTr_k) & \text{if } u_i \text{ is given and } i \neq k \\ u_i (-\int_{Tr_k} (\kappa \nabla w n) dTr_k + \frac{1}{2}) & \text{if } u_i \text{ is given and } i = k \\ (\int_{Tr_k} w dTr_k) (\kappa \nabla u_i n) & \text{if } (\kappa \nabla u_i n) \text{ is given} \end{cases} \quad (5.1.27)$$

$$x_i = \begin{cases} U_i & \text{if } dU_i \text{ is given} \\ \frac{\partial w_i}{\partial n} & \text{if } U_i \text{ is given} \end{cases} \quad (5.1.28)$$

From the equation

$$\int_{\partial\Omega_H} \left(\kappa \frac{\partial w}{\partial n} \right) \cdot u \, dx - \int_{\Sigma} w \cdot \frac{\partial u}{\partial n} \, dx = 0. \quad (5.1.29)$$

We will have the following relationship:

$$[G]U = [H]dU, \quad (5.1.30)$$

where

$$G_{ij} = \sum_{k=1}^N \int_{Tri_k} w \psi dTri_k, \quad (5.1.31)$$

$$H_{ij} = \sum_{k=1}^N \int_{Tri_k} (\kappa \nabla w n) \psi dTri_k, \quad (5.1.32)$$

U is a vector with the nodal values of the potential in the geometry, dU a vector with the nodal values of the flux, and ψ will be piecewise linear functions. Applying the formulas 5.1.31, and 5.1.32 for the inverse problem in ECG we have the matrix vector equation

$$\begin{bmatrix} G_{hh} & -H_{ht} \\ G_{th} & -H_{tt} \end{bmatrix} \begin{bmatrix} dU_h \\ U_t \end{bmatrix} = \begin{bmatrix} H_{hh} \\ H_{th} \end{bmatrix} \begin{bmatrix} U_h \end{bmatrix} - \begin{bmatrix} G_{ht} \\ G_{tt} \end{bmatrix} \begin{bmatrix} dU_t \end{bmatrix}, \quad (5.1.33)$$

where h refers to the surface of the heart, and t to the surface of the thorax. To simplify the notation, we will consider

$$H_{ht} = -H_{th} \quad (5.1.34)$$

$$H_{tt} = -H_{th}. \quad (5.1.35)$$

If we solve for dU_h , and substitute in the second equation we built the following equation

$$[H_{tt} - G_{th} G_{hh}^{-1} H_{ht}] U_t = [H_{th} - G_{th} G_{hh}^{-1} H_{hh}] U_h + [G_{th} G_{hh}^{-1} G_{ht} - G_{tt}] dU_t. \quad (5.1.36)$$

From the equation 5.1.36 we built the following operators

$$P = [H_{tt} - G_{th}G_{hh}^{-1}H_{ht}]^{-1}[H_{th} - G_{th}G_{hh}^{-1}H_{hh}], \quad (5.1.37)$$

$$Q = [H_{tt} - G_{th}G_{hh}^{-1}H_{ht}]^{-1}[G_{th}G_{hh}^{-1}G_{ht} - G_{tt}], \quad (5.1.38)$$

which gives

$$U_t = PU_h + QdU_t. \quad (5.1.39)$$

5.1.1.5 Example of Boundary Element Method in Matrix Form

We make the same test as for the BEM-FEM comparison, the result is the following figure

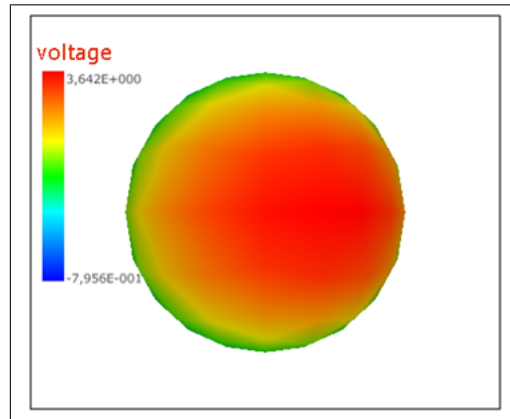


FIGURE 5.5: Calculated solution using BEM operators.

5.1.2 Regularization

The discretized inverse system of equations 5.1.7, considering a null flux condition will be summarized into the equation:

$$PU_h = U_t, \quad (5.1.40)$$

Where U_h are the nodal values in $\partial\Omega_H$, U_t are the nodal values in Σ , and P is transfer matrix of the size $N_h \times N_t$ (number of nodes in surface i). A regularization technique is necessary, because it is an ill-posed system. The regularization technique used in this study is a global Tikhonov scheme. For this global scheme, U_h , can be estimated by minimizing a generalized form of the Tikhonov functional:

$$\min(\| PU_h - U_t \|^2 + \lambda \| C(U_h - U'_h) \|^2) \quad \lambda > 0, \quad (5.1.41)$$

where C is a constrained matrix (the identity matrix), λ is the regularization parameter, and U'_0 is priori information.

5.1.2.1 Regularization Parameter

To find the optimal value of regularization parameter the L-Curve Method was used. The L-Curve is a parametric plot of $(\log_{10}(\| Ax_\lambda - b \|_2), \log_{10}(\| x_\lambda \|_2))$ for different values of the regularization parameter λ . The optimal value of Lambda for Thikonov-0 regularization can be obtained by the maximum value of the curvature given by the following formula:

$$\kappa(\lambda) = \frac{\rho'' * \eta' - \rho' * \eta''}{((\rho')^2 + (\eta')^2)^{3/2}} \quad (5.1.42)$$

where:

$$\rho = \log_{10}(\| Ax_\lambda - b \|_2) \quad (5.1.43)$$

$$\eta = \log_{10}(\| x_\lambda \|_2) \quad (5.1.44)$$

5.1.3 L-Curve algorithm

If we have a system $Ax = b$, in this case A will be P the transfer matrix of the system:

For a rectangular matrix A size m x n

if (m >= n)

(U,S,V)=Singular Value Decomposition(A)

L-curve(U,S,b) . (5.1.45)

If (m < n)

(U,S,V)= Singular Value Decomposition (A^T)

L-curve(V,S,b)

Where the L-Curve function is the following:

$$\begin{aligned}
\beta &= U^T b \\
\beta_2 &= \|b\| * \|b\| - \|\beta\| * \|\beta\| \\
\chi_i &= \frac{\chi_i}{s_i} \\
\text{reg_param}_{n\text{points}-1} &= \text{Max}(s_n, s_0 * \text{Smallest regularization parameter}) \\
\text{ratio} &= \frac{s_0}{\text{reg_param}_{n\text{points}-1}} \frac{1}{n\text{points}-1} \\
\text{for}(i = n\text{points} - 2; i \geq 0; i --) \\
\{ \\
&\quad \text{reg_param}_i = \text{ratio} * \text{reg_param}_{i+1} \\
\} \\
\text{for}(i = 0; i < n\text{points}; i ++) \\
\{ \\
&\quad \text{for}(j = 0; j < \text{size of } s; j ++) \\
&\quad \{ \\
&\quad \quad f_i = \frac{s_j^2}{s_j^2 + \text{reg_param}_i^2} \\
&\quad \quad \eta'_j = f_j * \chi_j \\
&\quad \quad \rho'_j = (1 - f_j) * \beta_j \\
&\quad \} \\
&\quad \eta_i = \|\eta'_j\| \\
&\quad \rho_i = \|\rho'_j\| \\
&\} \\
\} \\
\text{If } U \text{ is size } m \times n \\
\text{if}(m > n \qquad \qquad \qquad \beta_2 > 0) \\
\{ \\
&\quad \text{for}(i = 0; i < n\text{points}; i ++) \\
&\quad \{ \\
&\quad \quad \rho_i = \sqrt{\rho_i^2 + \beta_2} \\
&\quad \} \\
&\} \\
\rho &= \log(\rho) \\
\eta &= \log(\eta)
\end{aligned} \tag{5.1.46}$$

Then using the values of ρ and η and the curvature formula we can choose the regularization parameter λ . To apply the regularization we define the system as

$$Ax = b, \quad (5.1.47)$$

where

$$A = P^T P + \lambda C^T C, \quad (5.1.48)$$

$$b = P^T U_t + \lambda C^T C U'_h, \quad (5.1.49)$$

$$x = U_h. \quad (5.1.50)$$

This system can be minimized using gradient methods or non linear methods like the genetic algorithms.

For the following geometry:

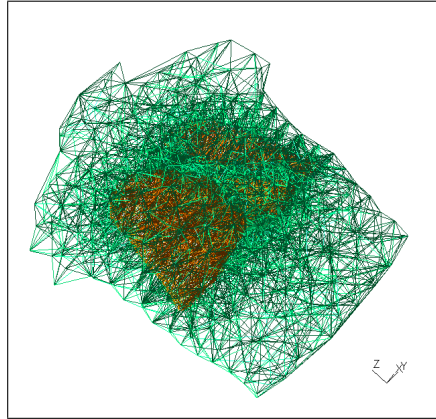


FIGURE 5.6: Heart thorax Geometry.

for the first value in the thorax using FEM operator, we calculate the value of regularization for the L-Curve, which will be 0.0006.

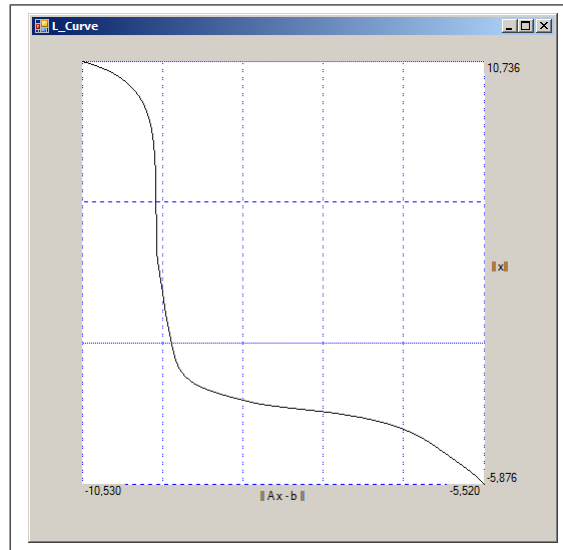


FIGURE 5.7: L-Curve.

5.1.3.1 Gradient Methods

5.1.3.2 Conjugate Gradient Method

Conjugate gradient will solve the matrix vector equation $Ax = b$. The matrix A has to be a real, symmetric, positive definite matrix. A matrix created using the FEM will be considered positive definite, also using the transformations from the last section we can minimize for the inverse problem. The input vector x_0 is an approximation of the solution; if we do not have an approximation then $x_0 = 0$.

$$r_0 = b - Ax_0$$

$$p_0 = r_0$$

$$k = 0$$

for $k = 0$ to *Max iterations* or $r_{k+1} > \text{residual}$

$$\alpha_k = \frac{r_k^T r_k}{p_k^T A p_k}$$

$$x_{k+1} = x_k + \alpha_k p_k \tag{5.1.51}$$

$$r_{k+1} = r_k - \alpha_k A p_k$$

$$\beta_k = \frac{r_{k+1}^T r_{k+1}}{r_k^T r_k}$$

$$p_{k+1} = r_{k+1} + \beta_k p_k$$

$$k = k + 1$$

end loop

5.1.3.3 Preconditioned Conjugate Gradient

To speed up the convergence of the algorithm, sometimes preconditioning is applied. The algorithm for the preconditioned conjugate gradient is the following:

$$\begin{aligned}
 r_0 &= b - Ax_0 \\
 z_0 &= M^{-1}r_0 \\
 p_0 &= z_0 \\
 k &= 0 \\
 \text{for } k &= 0 \text{ to Max Iterations or } r_{k+1} < \text{residual} \\
 \alpha_k &= \frac{r_k^T r_k}{p_k^T A p_k} \\
 x_{k+1} &= x_k + \alpha_k p_k \\
 r_{k+1} &= r_k - \alpha_k A p_k \\
 z_{k+1} &= M^{-1}r_{k+1} \\
 \beta_k &= \frac{z_{k+1}^T r_{k+1}}{z_k^T r_k} \\
 p_{k+1} &= z_{k+1} + \beta_k p_k \\
 k &= k + 1 \\
 \text{end loop}
 \end{aligned} \tag{5.1.52}$$

The algorithm for the preconditioned conjugate gradient, is almost the same as the same one as for the conjugate gradient. The main difference is the preconditioned matrix M . The used pre-conditioner was created using the following formula (from the matrix A).

$$\text{for } i = 0 \text{ to } N \quad M_{ii} = \sum_{j=0}^N A_{ij} \text{ end loop} \tag{5.1.53}$$

To make the inverse from this matrix we apply

$$M_{ii}^{-1} = \frac{1}{M_{ii}}. \tag{5.1.54}$$

5.1.3.4 Biconjugate Gradient

When the A matrix is not positive definite (like in the case of the Boundary Element Method) other methods should be applied like the bi-conjugate gradient method.

$$\begin{aligned}
r_0 &= b - Ax_0 \\
r'_0 &= b' - x'_0 A \\
p_0 &= M^{-1} r_0 \\
p'_0 &= r'_0 M^{-1} \\
k &= 0 \\
&\text{for } k = 0 \text{ to Max Iterations or } r_{k+1} > \text{residual} \\
\alpha_k &= \frac{r'_k M^{-1} r_k}{p'_k A p_k} \\
x_{k+1} &= x_k + \alpha_k p_k \\
x'_{k+1} &= x'_k + \alpha_k p'_k \\
r_{k+1} &= r_k - \alpha_k A p_k \\
r'_{k+1} &= r'_k - \alpha_k p'_k A \\
z_{k+1} &= M^{-1} r_{k+1} \\
\beta_k &= \frac{r'_{k+1} M^{-1} r_{k+1}}{r'_k M^{-1} r_k} \\
p_{k+1} &= M^{-1} r_{k+1} + \beta_k p_k \\
p'_{k+1} &= r'_{k+1} M^{-1} + \beta_k p'_k \\
k &= k + 1 \\
&\text{end loop}
\end{aligned} \tag{5.1.55}$$

5.1.3.5 Algorithm for the Conjugate Gradient Method for the Inverse Problem

For a series of vectors U_t^i which describe the voltage of the thorax at the sample i , and a P which will be the transfer operator, we will have the following algorithm, to get a

series of vectors U_h^i .

$$A = P^T P + \lambda C^T C,$$

for $i = 0$ to the quantity of the series of vectors

$$b = P^T U_t + \lambda C^T C U_h^{i-1},$$

$$r_0 = b - A U_h^{i-1}$$

$$p_0 = r_0$$

$$k = 0$$

for $k = 0$ to Max iterations or $r_{k+1} < \text{residual}$

$$\alpha_k = \frac{r_k^T r_k}{p_k^T A p_k} \tag{5.1.56}$$

$$x_{k+1} = x_k + \alpha_k p_k$$

$$r_{k+1} = r_k - \alpha_k A p_k$$

$$\beta_k = \frac{r_{k+1}^T r_{k+1}}{r_k^T r_k}$$

$$p_{k+1} = r_{k+1} + \beta_k p_k$$

$$k = k + 1$$

end loop

$$U_h^i = x$$

end loop

If we consider $U_h^{i-1} = 0$, then we can modify the algorithm using parallel programming:

$$A = P^T P + \lambda C^T C,$$

Parallel for $i = 0$ *to the quantity of the series of vectors*

$$b = P^T U_t,$$

$$r_0 = b$$

$$p_0 = r_0$$

$$k = 0$$

for $k = 0$ *to Max iterations or* $r_{k+1} < \text{residual}$

$$\alpha_k = \frac{r_k^T r_k}{p_k^T A p_k} \tag{5.1.57}$$

$$x_{k+1} = x_k + \alpha_k p_k$$

$$r_{k+1} = r_k - \alpha_k A p_k$$

$$\beta_k = \frac{r_{k+1}^T r_{k+1}}{r_k^T r_k}$$

$$p_{k+1} = r_{k+1} + \beta_k p_k$$

$$k = k + 1$$

end loop

$$U_h^i = x$$

end loop

5.1.3.6 Tests

5.1.3.7 Test1

Using the Monodomain a pulse is generated in the top of the left ventricle. Next, the voltage in the thorax is created using an operator from the Laplace model in 3D. The model consists of the surfaces of heart and thorax. Then, the inverse solution is calculated using the static regularization, with the minimum energy norm. The regularization parameter was chosen using the L-Curve method. The value is $\lambda = 0.00001$.

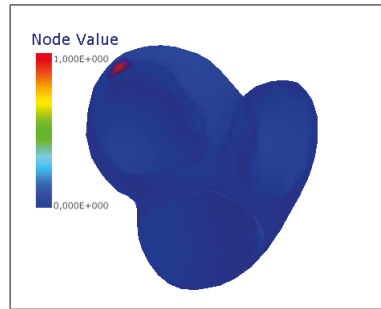


FIGURE 5.8: Nodes where the impulse was applied.

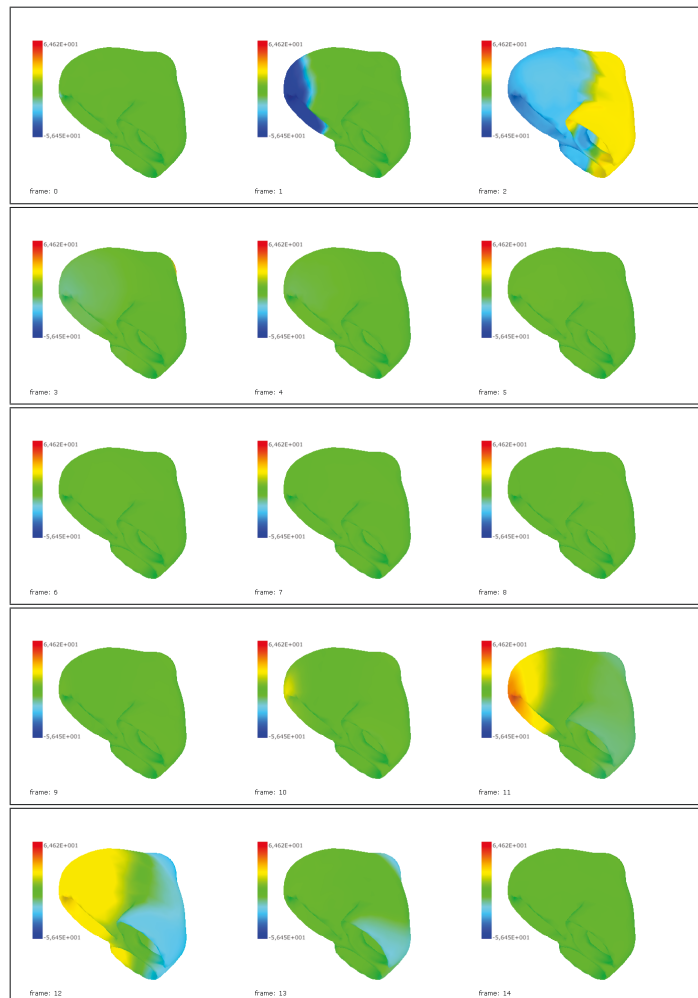


FIGURE 5.9: Voltage Distribution for the complete heart cycle with one pulse generated using an homogeneous torso.

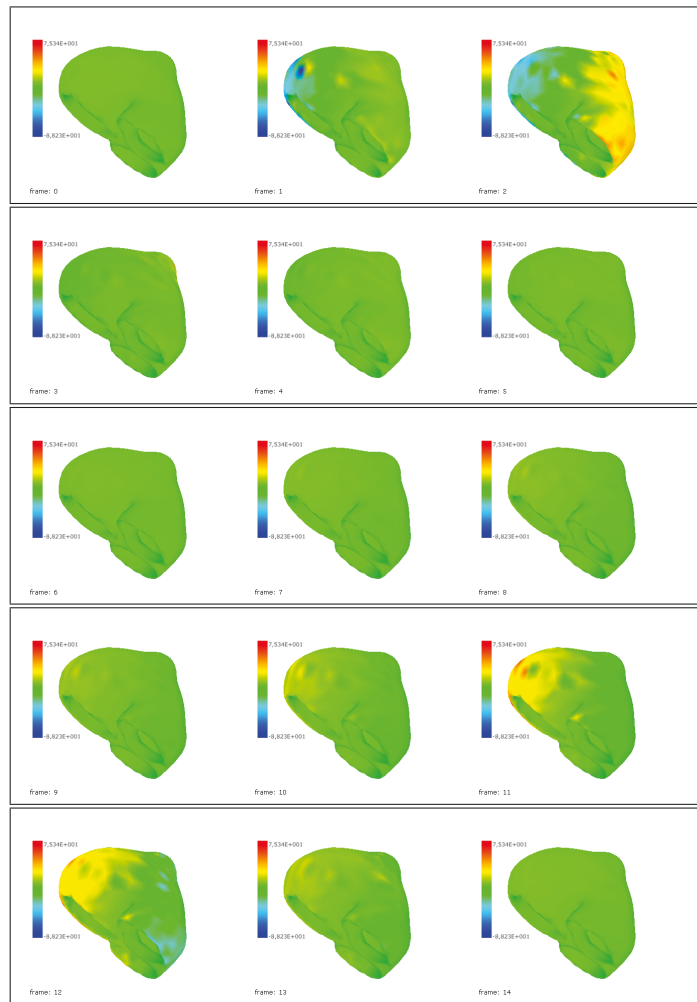


FIGURE 5.10: Voltage Distribution for the complete cycle using static regularization $\lambda = 0.00001$ using an homogeneous torso.



FIGURE 5.11: Voltage Distribution for the complete cycle using static regularization $\lambda = 0.00001$ using a non-homogeneous torso.

There are some nodes where the difference between the homogeneous and non-homogeneous case is considerable.

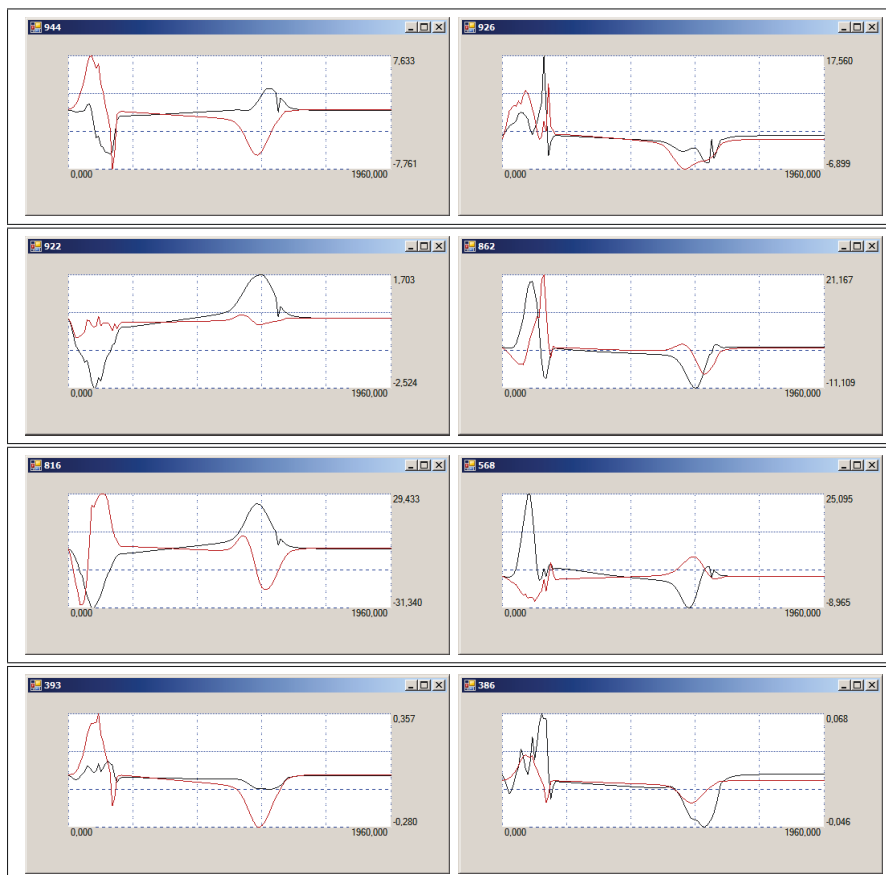


FIGURE 5.12: Comparison in some nodes between the homogeneous (black) and non-homogeneous (red) case.

5.1.3.8 Test2

Using the Monodomain two pulses are generated: in the right , and left on the top of the ventricles. Next, the voltage in the thorax is created, as in the last test. Then, the inverse solution is calculated with the static regularization.

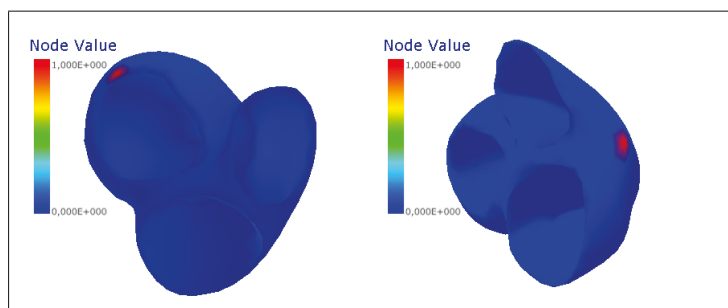


FIGURE 5.13: Nodes where the impulse was applied for test 2.

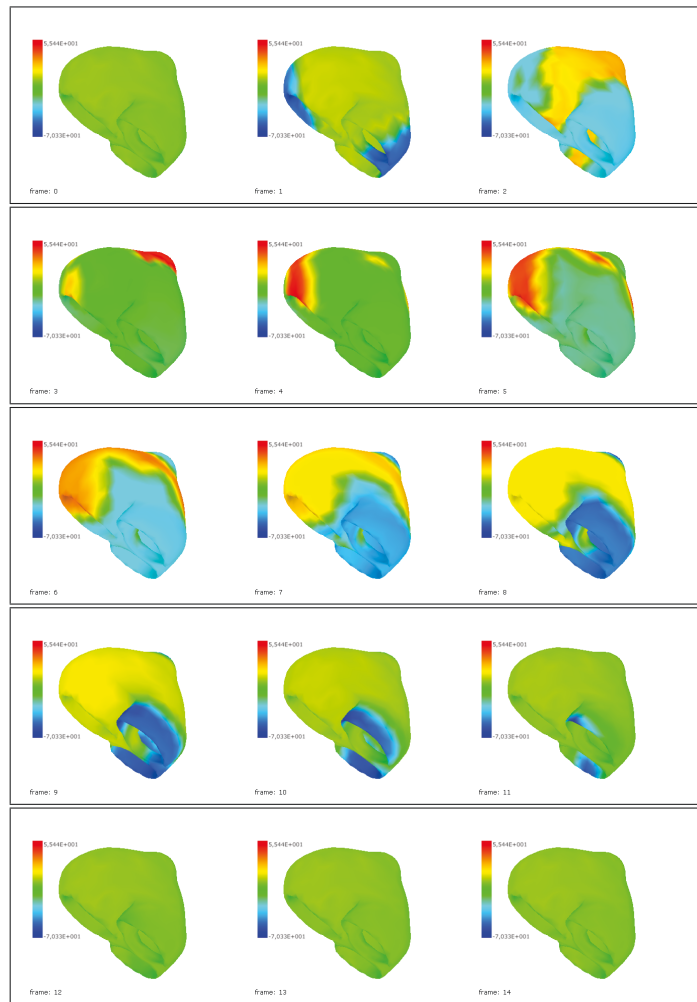


FIGURE 5.14: Voltage Distribution for the complete cycle for the test 2.

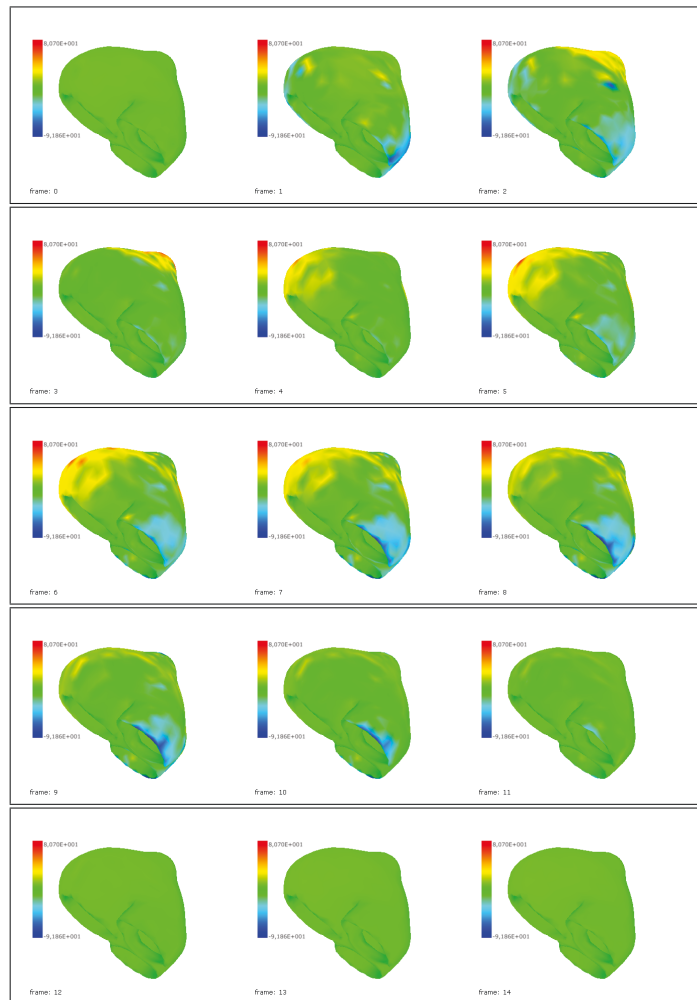


FIGURE 5.15: Voltage Distribution for the complete cycle using static regularization $\lambda = 0.00001$ using an homogeneous torso.

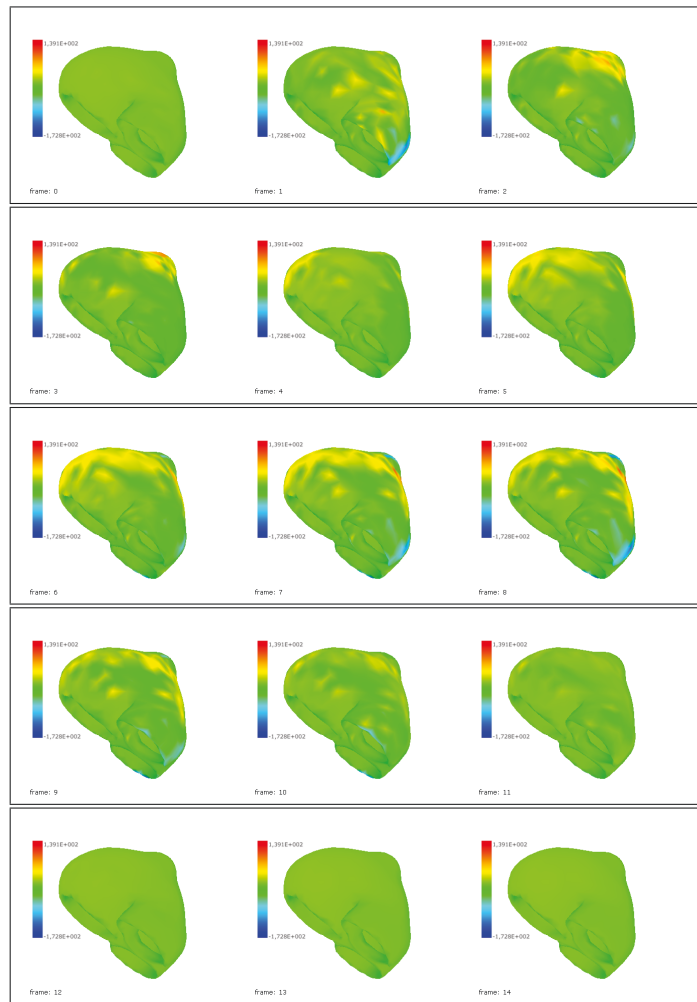


FIGURE 5.16: Voltage Distribution for the complete cycle using static regularization $\lambda = 0.00001$ using a non-homogeneous torso.

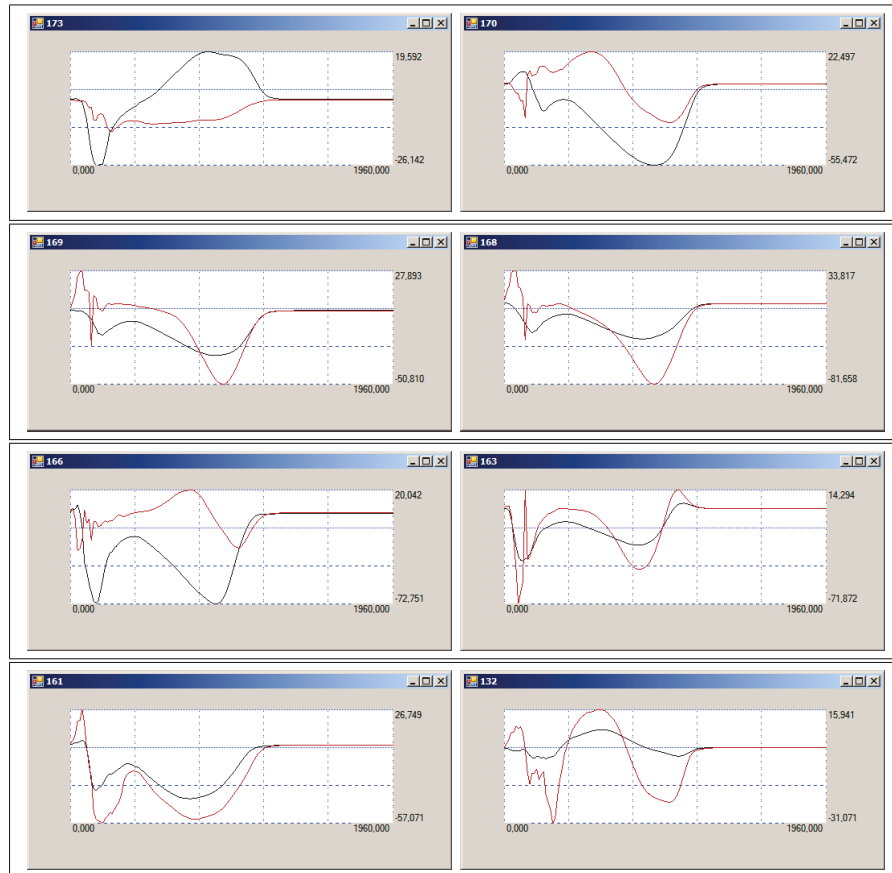


FIGURE 5.17: Comparison in some nodes between the homogeneous (black) and non-homogeneous (red) case.

5.1.3.9 Test3

Using the Monodomain three pulses are generated: in the center, right, and left on the top of the ventricles. Next, the voltage in the thorax is created, as in the last test. Then, the inverse solution is calculated with the static regularization.

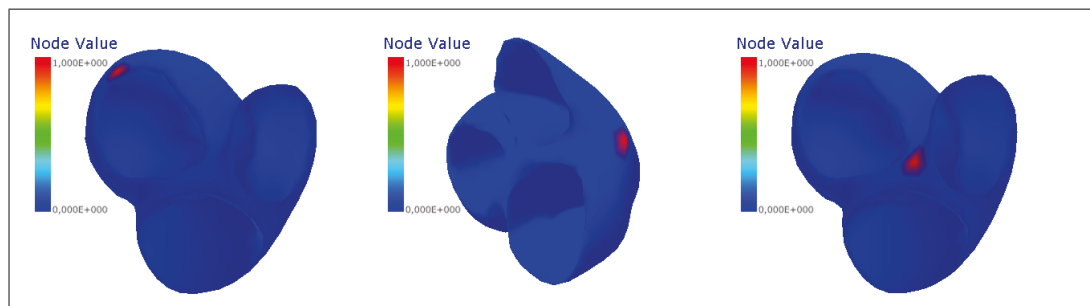


FIGURE 5.18: Nodes where the impulse was applied.

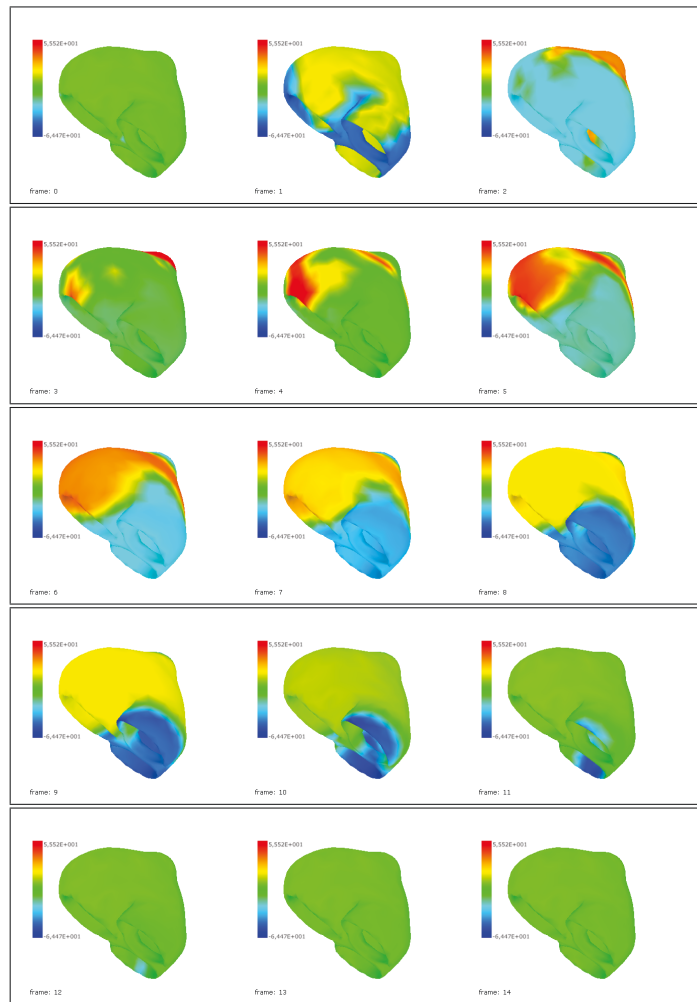


FIGURE 5.19: Voltage Distribution for the complete heart cycle with three pulses generated using an homogeneous torso.

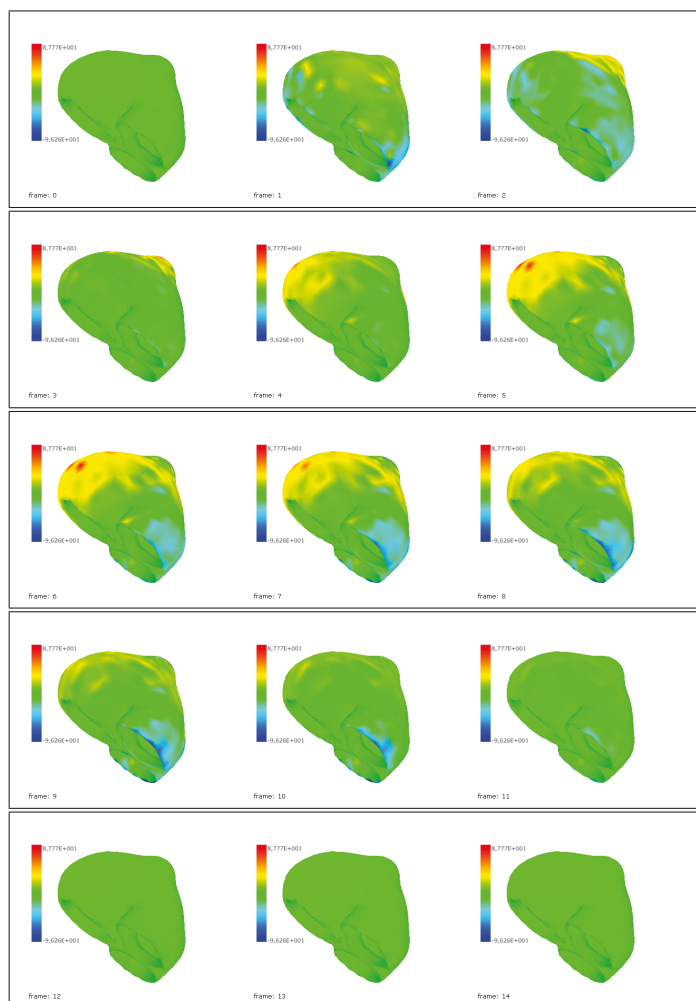


FIGURE 5.20: Voltage Distribution for the complete cycle using static regularization $\lambda = 0.00001$ using an homogeneous torso.

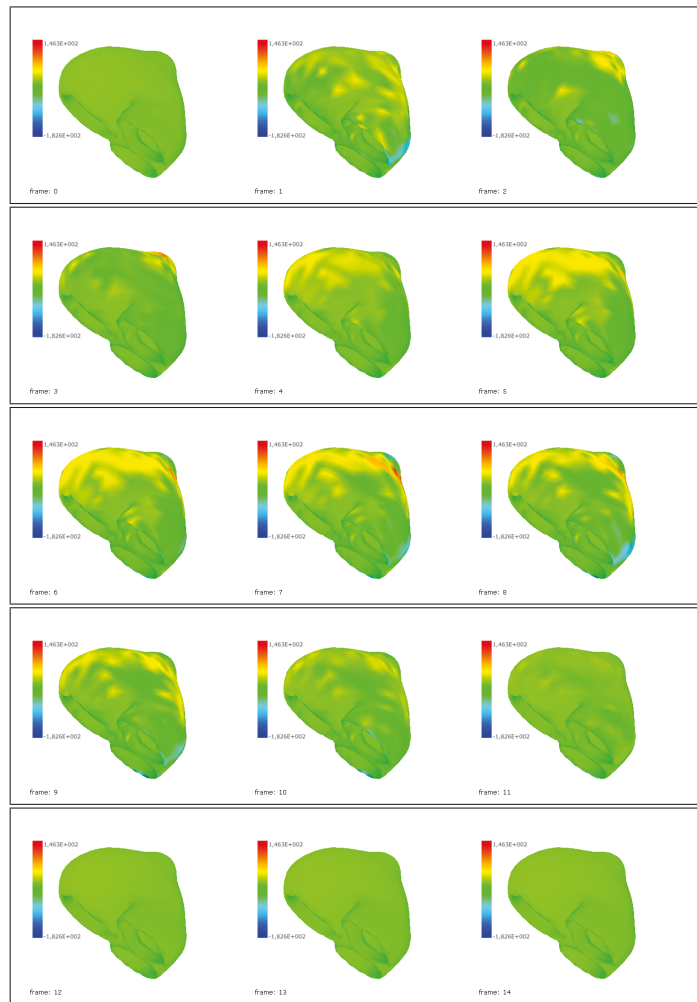


FIGURE 5.21: Voltage Distribution for the complete cycle using static regularization $\lambda = 0.00001$ using a non-homogeneous torso.

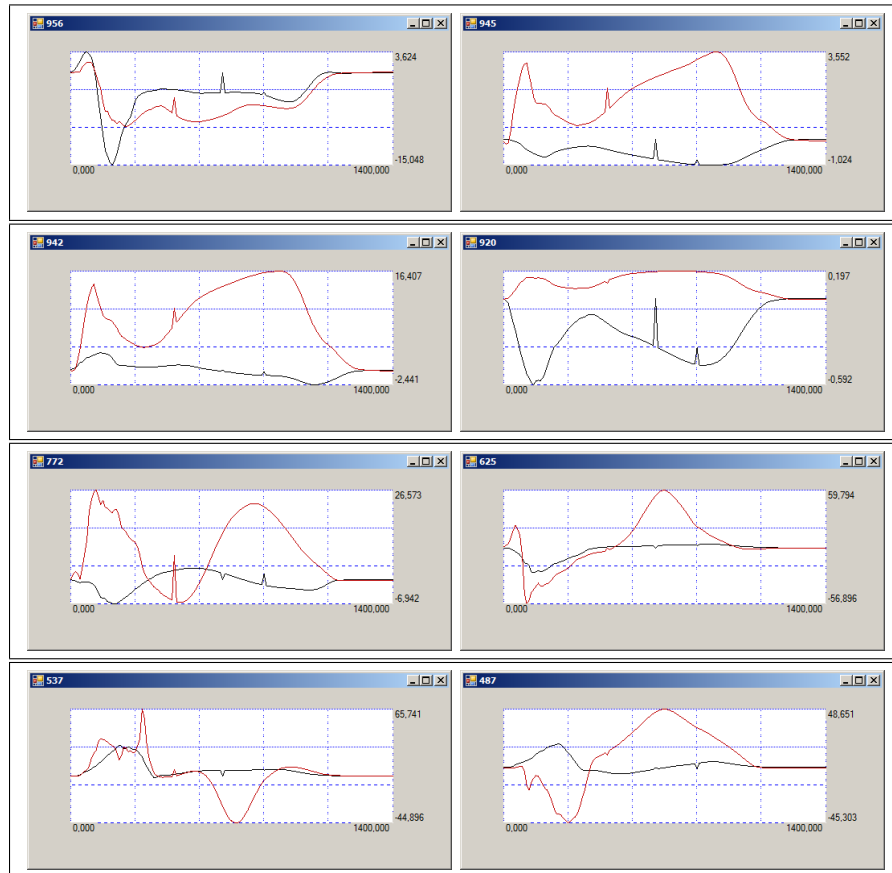


FIGURE 5.22: Comparison in some nodes between the homogeneous (black) and non-homogeneous (red) case.

5.1.3.10 Discussion

For each of the tests the difference between two distributions was calculated with,

$$error = \frac{\sum(U_{h_i} - U_{c_i})^2}{\sum(U_{h_i})^2} \quad (5.1.58)$$

The time simulated is 1 second.

	Original-H	Original-NH	H-NH
1 Node Activation	0.617	0.771	0.748
2 Node Activation	0.686	0.842	1.131
3 Node Activation	0.676	0.845	1.083

In test 1 when there is only one pulse, we can guess the origin of the impulse by the propagation of the wave. But if we compare test 2, and 3 it is difficult to say where the wave is originating from. If we compare our results to the ones obtained by Ramanathan

[42], we can see that there is a considerable difference between the homogeneous and non-homogeneous case, for the inverse problem, although the direct problem is the same. The difference between the original (homogeneous) and non homogeneous in the thorax is 0.0018, using the same criteria as the other tests.

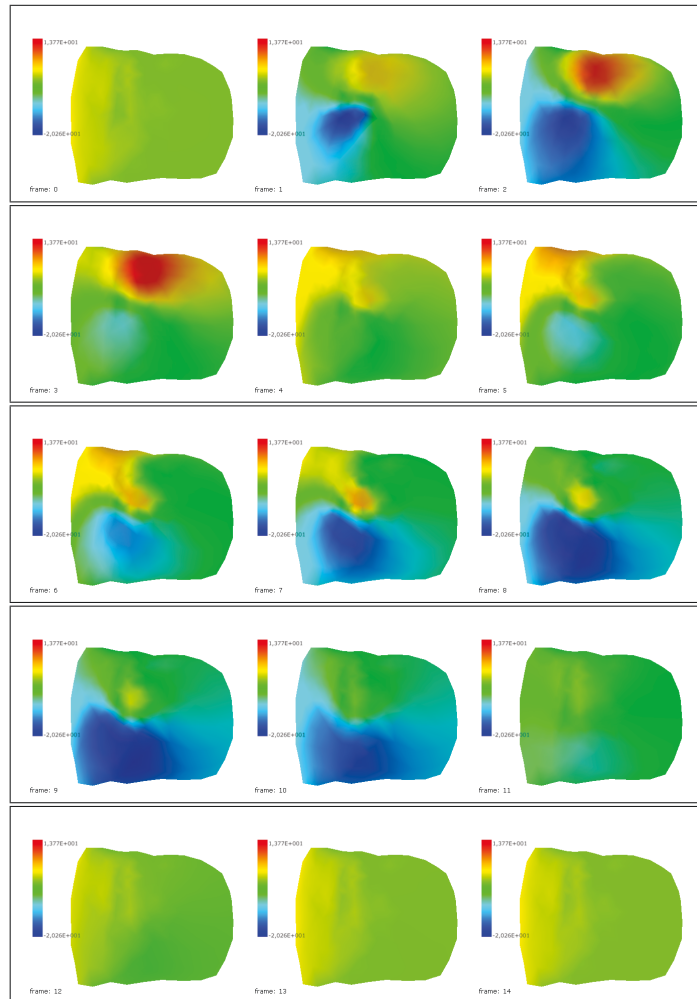


FIGURE 5.23: Voltage Distribution for the complete cycle in the thorax for the homogeneous case .

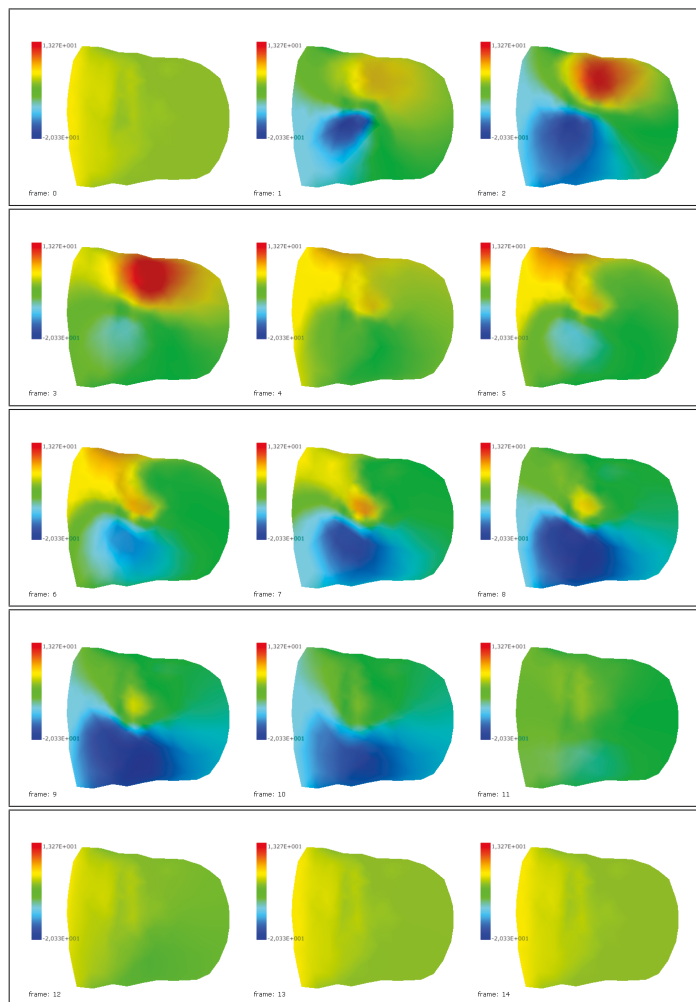


FIGURE 5.24: Voltage Distribution for the complete cycle in the thorax for the non-homogeneous case .

The difference is the used geometry. The mesh used is closed, and the activation points are on the surface of the ventricles, not on the top of the ventricles. When we applied the impulses in the top of the ventricles (90 degrees from the long axis [43]) the minimum energy norm is not able to reconstruct them (figure 5.25).

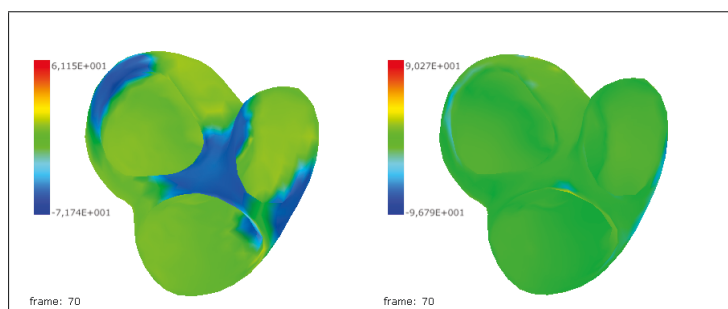


FIGURE 5.25: Comparison in the top of the ventricles.

For example if we make the same procedure in a closed geometry. The results are different (the reconstruction is better).

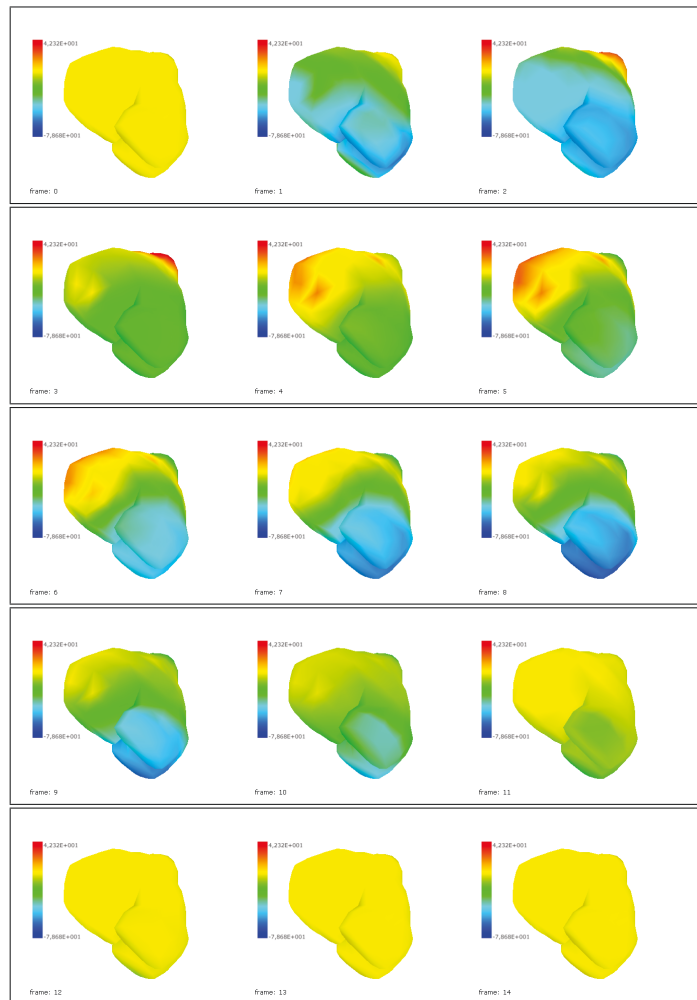


FIGURE 5.26: Voltage Distribution for the complete cycle in a closed heart.

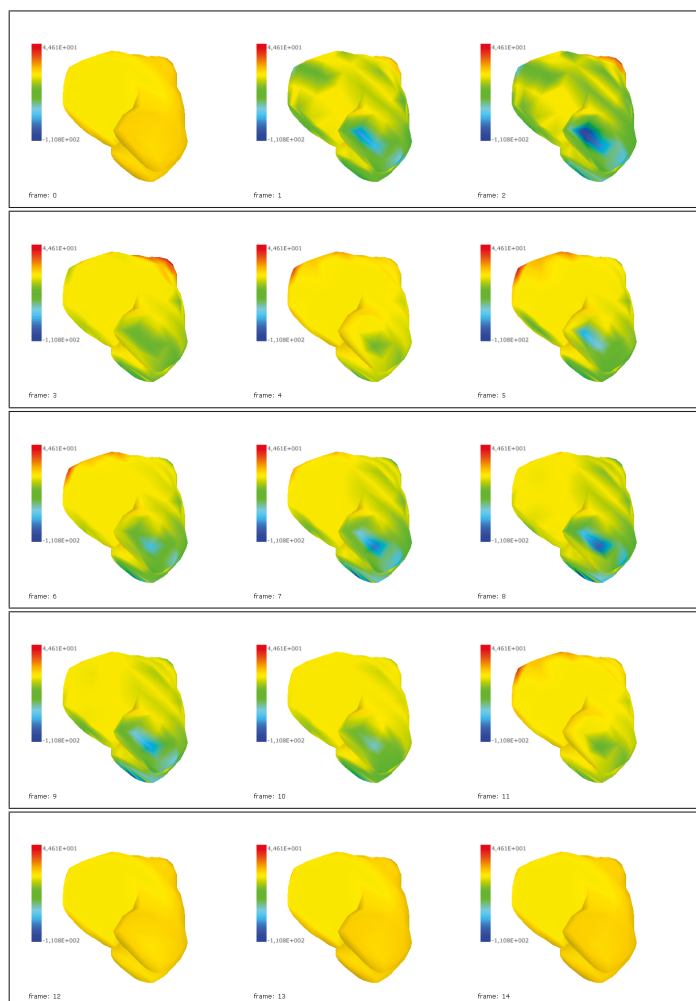


FIGURE 5.27: Voltage Distribution for the complete cycle using static regularization $\lambda = 0.007$ using a homogeneous torso.

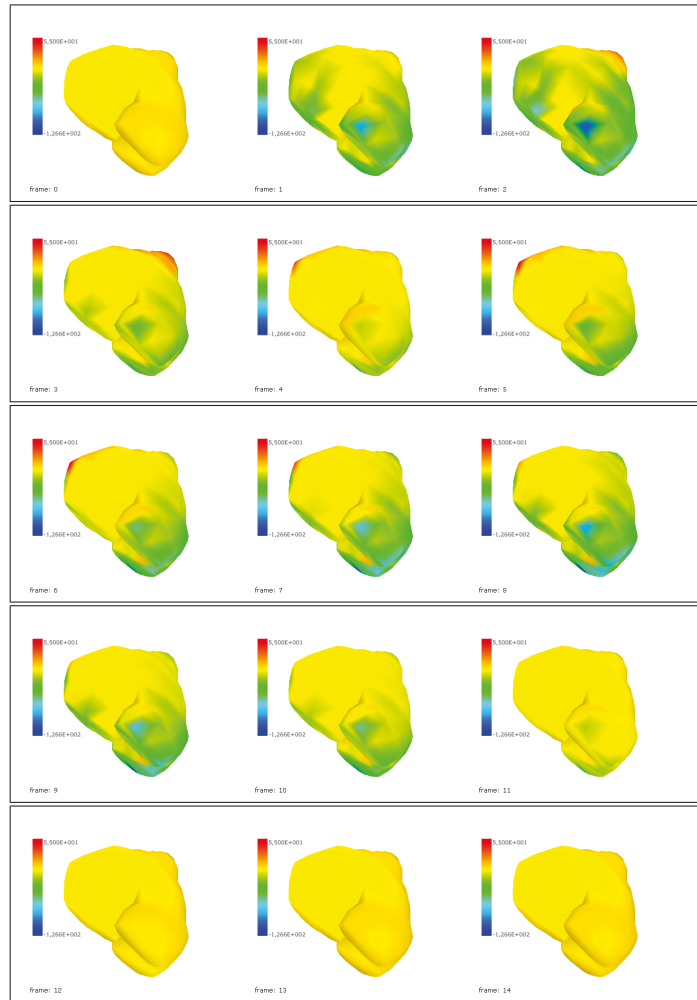


FIGURE 5.28: Voltage Distribution for the complete cycle using static regularization $\lambda = 0.007$ using a non-homogeneous torso.

	Original-H	Original-NH	H-NH
Activation	0.463	0.444	0.244

From the tests we conclude that for the direct problem the homogeneous and non-homogeneous case is the same thing, but for the inverse problem the calculated solutions using the minimum energy norm are different when using a more complex geometry.

If we make the comparison to the examples in Chapter 2, we get the following over all results using the same geometries. In this table we make the comparison of the two procedures. We call volume conductor extracellular the reconstruction of the extracellular potential on the heart's surface using the operator given by the Laplace's equation

only. It is not possible with this method to make the reconstruction of the transmembrane potential. The other two values are for the reconstruction using the monodomain operator.

	One Pulse	3 Pulse	1% Noise	Homogeneous-Non Homo.
<i>VolumeConductorExtracellular</i>	0.6564	0.6947	0.9664	0.8106
<i>MonodomainExtracellular</i>	0.0790	0.2318	0.0841	0.3375
<i>MonodomainTransmembrane</i>	0.0091	0.0219	0.0097	0.0386

Chapter 6

Chapter 6. Genetic Algorithms

6.1 Genetic Algorithms

The genetic algorithms (GA) optimization process begins by setting a random set of possible solutions, called the population, with a fixed initial number of individuals. The GA is a global optimization method which is used for searching the best solution for an ill-posed problem. Each individual is defined by optimization variables and is represented as a numerical type string or chromosome. Note that, this method has the ability to find the global optimum of a problem with a lot of unknown variables.

6.1.1 General Genetic algorithms

Usually, for the inverse problem in electrocardiography, in the form $PU_h = U_t$, a solution is found by minimizing the Tikhonov functional in [44] using the conjugate gradient method. Another way to approximate the solution vector U_h is by using GA, a good explanation of the genetic algorithms can be found in [45].

First it is necessary to define a maximum limit (*Max*) and a minimum limit (*Min*) for the values the genes (nodal values in this case) can take. Next, a random generated first family of solutions (chromosomes) will be created. For example, 100 chromosomes, in which each chromosome will be a vector of the size of the nodes in the heart's surface. Next, the solution vectors are evaluated by the norm

$$\min_{U_h} \|PU_h - U_t\|^2 \quad (6.1.1)$$

and then ordered, being the first vector the smallest norm value and the last vector the greatest norm value. Then we proceed to mate the chromosomes. The first 25 chromosomes in the ranking will mate by choosing randomly from the first 50 chromosomes.

The first 25 are called mothers, and the randomly chosen fathers. We have to do this process for the 25 first values. The mating process works in the following way: • Select Mother and Father. • Create two new chromosomes of the same length (*son0* and *son1*). • Choose a random cut that can go from the position 0 to the position N of the chromosome vector. • Exchange the values as in Figure 6.1

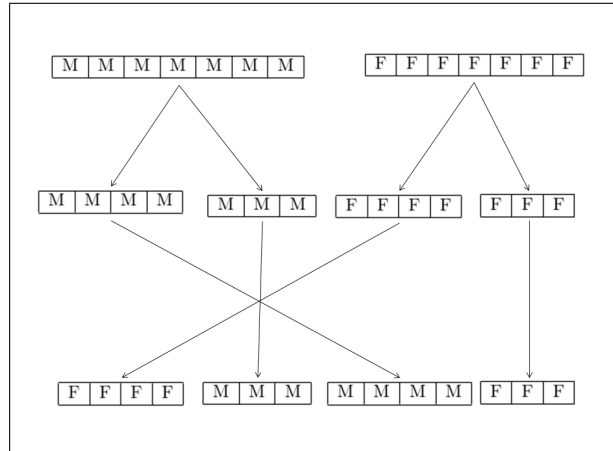


FIGURE 6.1: Chromosome Mating.

Once the two sons are created, the mutation value is introduced. In this case, the mutation can go from 0 to 10 • Choose randomly the percentage of mutation (0-10%). • Repeat for the chosen percentage the choice of a gene. • Randomly choose a new value for this gene. Once finished the mating and the mutation, there will be once again a family of 100 chromosomes. The procedure is repeated until

$$\|PU_0 - U_t\|^2 \quad (6.1.2)$$

where β is the stop parameter value. Normally this approach is not used for solving inverse problems. The GA solution will be completely random, with no restrictions, and could take a lot of time to calculate.

6.1.2 General Genetic Algorithm Example

To demonstrate the genetic algorithms, we will use the traveling salesman problem. The traveling salesman problem is the following: Given a number of cities N , with a distance between them; which is the smallest route to visit all of them once and return to the first city. The traveling salesman problem is a $O(n!)$ problem so trying to solve it using brute force is not practical.

To test the system we will have 20 cities put around a circle. We do this because the circle

will be the exact solution an easy to demonstrate. To create the genetic algorithm we will create a set of chromosomes with 20 values each one. The genes in the chromosome are the order of the cities. Then each of the chromosomes also will have a cost; which is the total distance from all of the cities.

The algorithm is the following:

```
    Create 100 random chromosomes (create the population).
    For (costjstop parameter)
    {
        Sort the chromosomes (Evaluate).
        For (the first 25 chromosomes) (Mate)
        {
            Select a father from the first 50.
            Create Son0, Son1.
            Mutate Son0, Son1 by changing order of the cities for k times.
        }
        Display Results
    }
```

(6.1.3)

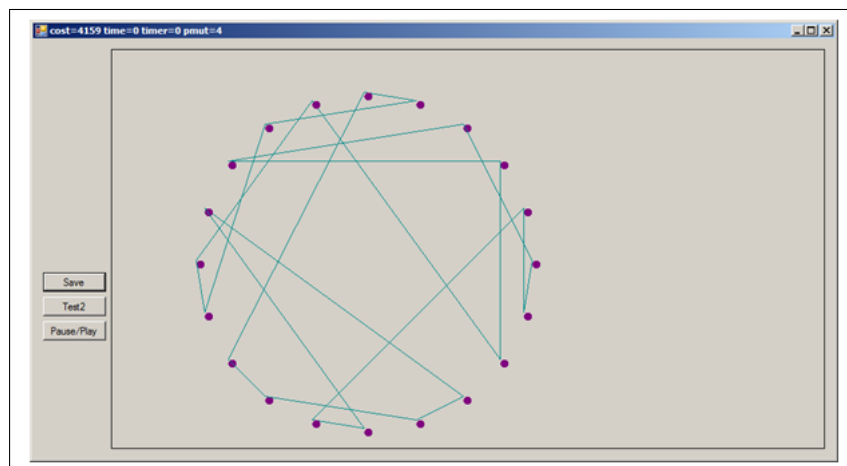


FIGURE 6.2: Initial solution.

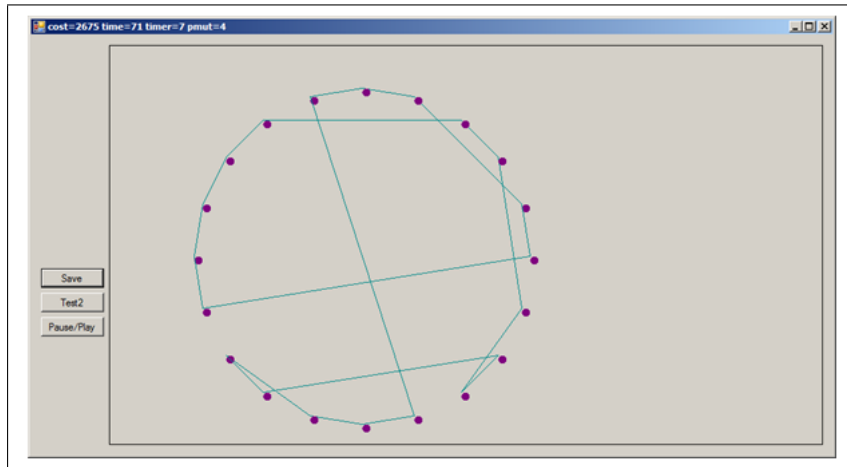


FIGURE 6.3: Iteration 71.

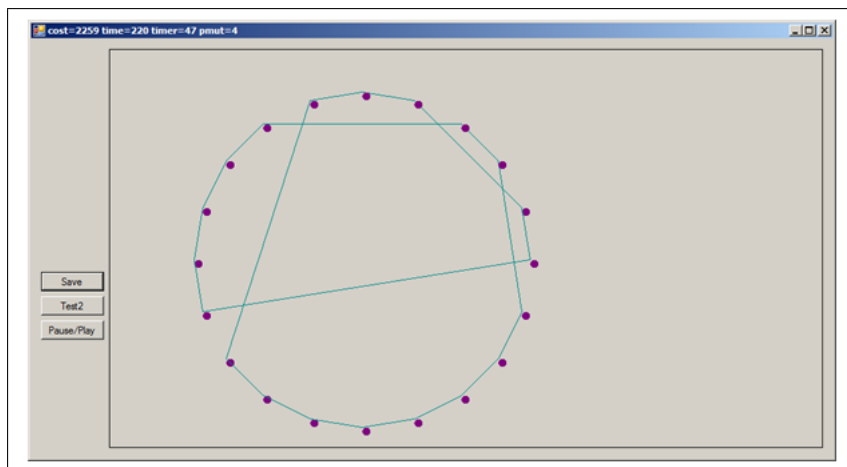


FIGURE 6.4: Iteration 220.

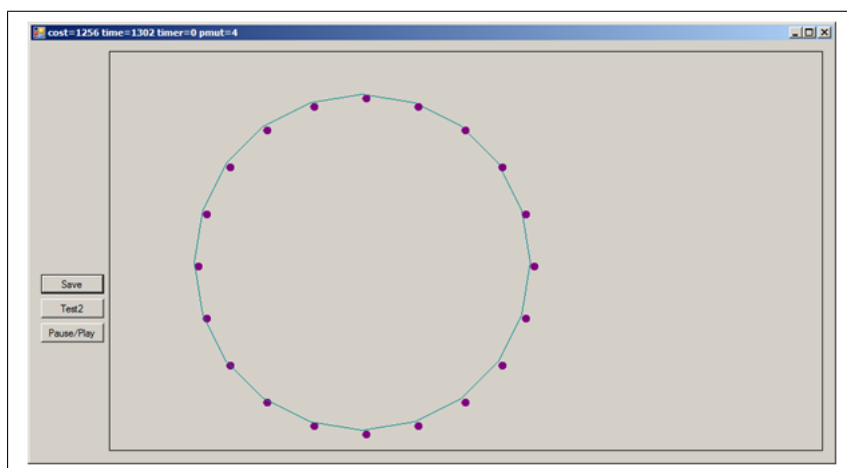


FIGURE 6.5: Iteration 1302.

6.1.3 Two-Step Algorithm

In this section the two-step algorithm using genetics and Tikhonov scheme will be explained. The fundamental solution of Laplace in 3D centered at a point (ξ, η, ζ) , is given by

$$\omega = -\frac{A}{4\pi\sqrt{(x-\xi)^2 + (y-\eta)^2 + (z-\zeta)^2}} + B \quad (6.1.4)$$

Considering $B = 0$, there will be 4 coefficients to search (ξ, η, ζ, A) . Using genetic algorithms, we look for the coefficients, to create a solution U_{hg} until the relationship (199) is satisfied or 1000 iterations are reached. The chromosomes will have a length of 28 integer type values. In the chromosome the values 0 to 6 will be the values of the coefficient A. The values from 7 to 13 will be the values of ξ , 14 to 20 η , and 21 to 27 ζ . Each parameter will consist of 7 genes, for example as in the following Figure.

0	1	2	3	4	5	6
---	---	---	---	---	---	---

FIGURE 6.6: Genes for a positive parameter.

The values from table 2 , will give a value of 123.456, being the first value, the sign value. For a negative value (-123.456), the same genes will be as in the following Figure.

1	1	2	3	4	5	6
---	---	---	---	---	---	---

FIGURE 6.7: Genes for a negative parameter.

Each chromosome will have the following form:

1	1	2	3	4	5	6	1	1	2	3	4	5	6	1	1	2	3	4	5	6	1	1	2	3	4	5	6
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

FIGURE 6.8: Chromosome Example.

Using this method, the family of solutions for Genetic Algorithms is created, and evaluated. The algorithm runs until the condition

$$\|PU_0 - U_t\|^2 < \beta \text{ for } \beta = 0.05 \quad (6.1.5)$$

is satisfied, or 1000 iterations are achieved. Once, the restrictions are satisfied, the answer is used as priori information in the Tikhonov regularization scheme. The Tikhonov functional to be minimized will be (5.1.41), where U'_h is the solution built using the GA

scheme. The following is the pseudo-code of the two-step algorithm.

```

Initialization
Create family of chromosomes
While ( $\beta > 0.05$  or iterations  $\leq 1000$ )
{
    Evaluate  $\|PU_0 - U_t\|^2 < \beta$  for  $\beta = 0.05$  (6.1.6)
    Create new family of chromosomes
    Mutate
}
Solve :  $\min_{U_0} = \|PU_0 - U_t\|^2 + \lambda \|C(U_h - U'_h)\|^2$ 

```

6.1.4 Tests

To test the two-step algorithm we create a voltage distribution in the thorax by using the Monodomain model (to resemble the real electrical activity of the heart). Then, we reconstruct the membrane potential using the minimum energy norm $V'_m = 0$, and the genetic algorithms solution $v'_m = v_g$. To evaluate the precision of the reconstructed solution we use the following formula

$$difference = \frac{\sum (V_{m_i} - V_{m_i*})^2}{\sum (V_{m_i})^2} \quad (6.1.7)$$

for the difference between the original distribution V_m and the calculated V_{m*} .

6.1.4.1 Test 1

In our first test we apply an impulse in the basal plane over the left ventricle. The original membrane potential generated with the cardiac model is in Figure 6.9. The reconstruction using the minimum energy norm is found in figure 6.10, and using the two-step algorithm in Figure 6.11. The heart is inverted showing the basal plane in the bottom, and the apex on the top for visualization purposes.

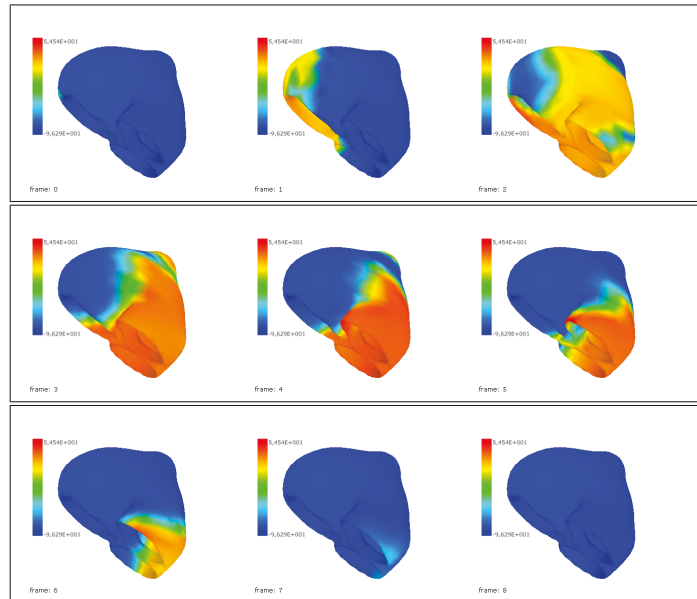


FIGURE 6.9: Original Membrane Potential Distribution for 0 ms, 50 ms, 150 ms, 200, ms, 250 ms, 300 ms, 350 ms, 400 ms created using the Monodomain model for one pulse.

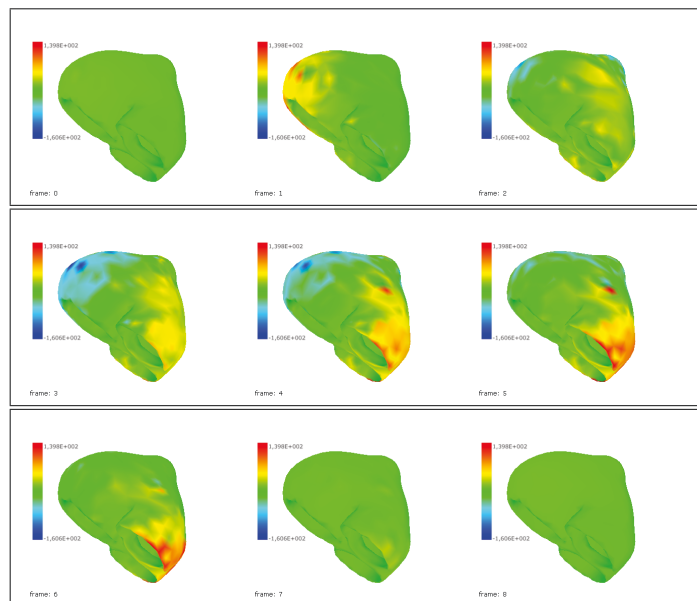


FIGURE 6.10: Membrane Potential Distribution for 0 ms, 50 ms, 150 ms, 200, ms, 250 ms, 300 ms, 350 ms, 400 ms originated using the minimum energy norm.

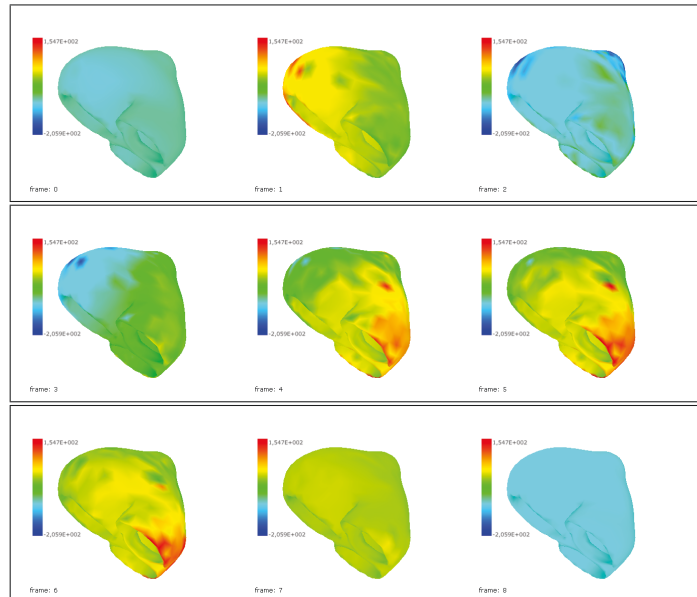


FIGURE 6.11: Membrane Potential Distribution for 0 ms, 50 ms, 150 ms, 200 ms, 250 ms, 300 ms, 350 ms, 400 ms originated using the two-step algorithm.

6.1.4.2 Test 2

In this experiment we apply impulses in three points over the basal plane; in the left ventricle, right ventricle and the wall that divides them. The original membrane potential distribution is in Figure 6.12. The reconstruction using minimum energy norm is in Figure 6.13, and using the two-step in Figure 6.14.

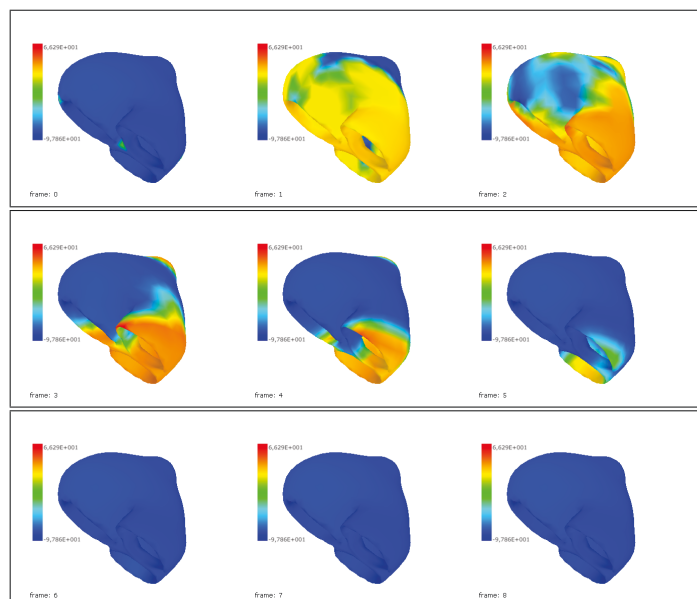


FIGURE 6.12: Original Membrane Potential Distribution for 0 ms, 50 ms, 150 ms, 200 ms, 250 ms, 300 ms, 350 ms, 400 ms created using the Monodomain model for three pulses over the basal plane.

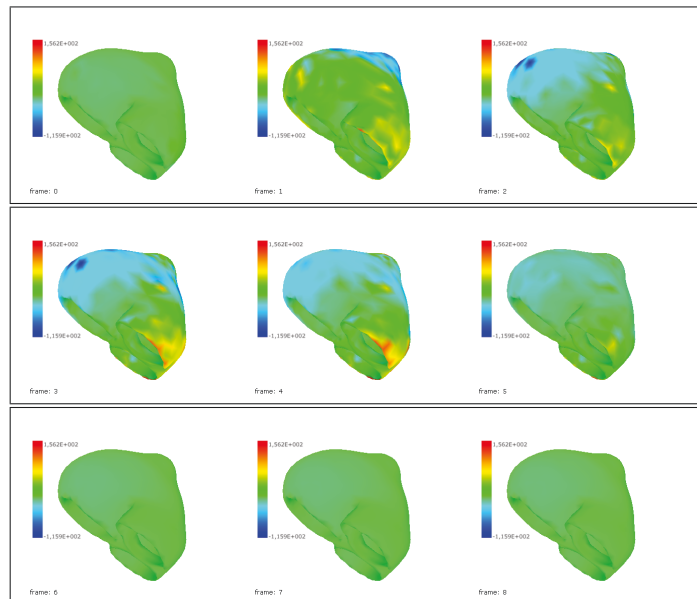


FIGURE 6.13: Membrane Potential Distribution for 0 ms, 50 ms, 150 ms, 200 ms, 250 ms, 300 ms, 350 ms, 400 ms created using the minimum energy norm.

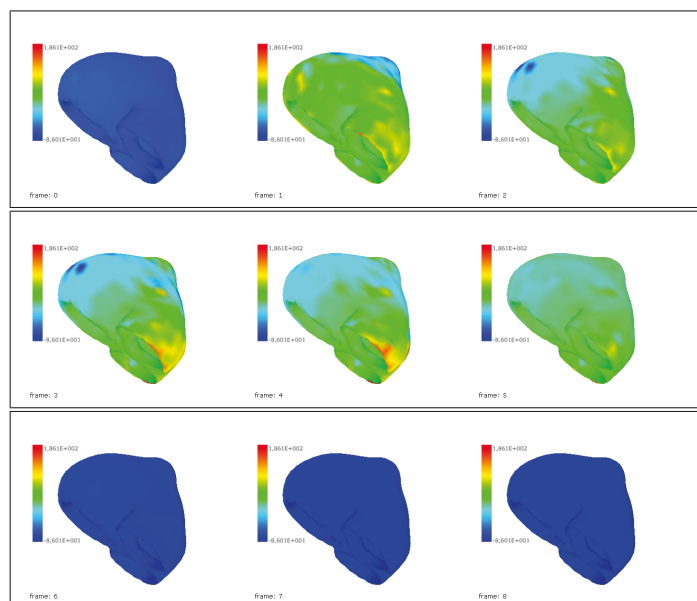


FIGURE 6.14: Membrane Potential Distribution for 0 ms, 50 ms, 150 ms, 200 ms, 250 ms, 300 ms, 350 ms, 400 ms created using the two-step algorithm.

TABLE 6.1: Comparison between the distributions from experiments 1 and 2, and the calculated using the inverse problem for these different methods; Minimum Energy , Genetic Algorithms and Two-Step Algorithm.

	1 Pulse	3 Pulse
Minimum Energy	0.9674	0.9770
Genetic Algorithms	0.5244	0.3874
Two-Step Algorithm	0.4949	0.3645

If we compare the different methods, and considering a computer with Intel(R) Core(TM) i7-2860QM CPU @ 2.50Hz and 8.00 (RAM) the reconstruction time in *ms* is the following programmed using *C#* and OpenGL for the graphics:

	1 Point	3 Point
Volume Conductor Parallel	1108054	1195967
Volume Conductor	7110906	7111203
Monodomain Operator	13510721	13511285
Genetics	29865805	29867052

Chapter 7

Chapter 7. Conclusion

7.1 Conclusion

As mentioned before the problem of inverse electrocardiography is an ill-posed problem. This means that a small perturbation can change the response in the heart. This is the reason why it is necessary to make some type of regularization. In this thesis we made a comparison between the calculated surface voltage nodal values with different values for the Tikhonov regularization parameter λ , and generated thorax distributions using models of electrical activity of the heart as explained in the chapter 3. One of the points in the inverse problem is the choice of the regularization parameter λ , as seen in the simulations a small change in λ changes drastically the result in the heart surface. To choose the value of λ , the L-curve method was used, but after a series of tests the value of λ calculated using only this method can produce over or under regularization. Once we had the value of λ , we made the test for the homogeneous and non homogeneous case. From the numerical tests we conclude that the homogeneous and non homogeneous case is the same for the direct problem, but the results in the surface of the heart are different for the inverse problem. We can see that differences depend of the geometry: the differences are presented in the nodes closest to the back, where the consideration of the different conductivity of the lungs affects the more. Actually we can see these non homogeneities as some noise add to the model. To not consider the lungs for example prove to be a mistake for the reconstructed signals, no matter which method was used. Other problem to consider is that the calculation of the transfer matrix, which changes from person to person, because of the geometry. During the calculation of this matrix there is a computational problem because it is necessary as one of the steps to make the calculation of the inverse from the sub-matrix A_{vv} , which depends on the density of the geometry and has the size of the nodes in the volume in between the surface of the heart

and the surface of the thorax. Even if the matrix is created using explicit expressions and BEM it will be necessary to calculate a huge inverse matrix. We can say that the number of nodes which can be used for the calculations of the inverse problem depend of the calculation of this inverse matrix or the equivalent in BEM. Another important thing to consider is how fast we want a solution to the inverse problem. For example if we want to have a close to real-time system we should use the volume conductor model with quasi-static condition. Once the geometry is processed and operators created, we can use parallelization to solve very fast the system by partitioning the samples and solving them by multi threading. This solution is fast, but the tests showed that the quality of the reconstruction is very low. Following this line of thinking to improve the quality of the reconstruction we can constrain the model by saying that the solution over the heart's surface is a solution of the Laplace model. To do this we will need non-linear minimization thus that is why we used genetic algorithms. The trade off is that in theory we can get to a very good solution, but the time will be huge for each sample (as the tests show), although the quality will be better. So with these tests we showed that including constrains to the inverse solution improves considerably the reconstruction on the heart's surface. That is why we use a model of electrical activity of the heart, and then the solution will have to fall in one of the possible solutions of the model. If we make the comparison in time, as in the table 6.1.4.2, and we use genetic algorithms they take around 8 hours to give a solution for one simulated second. Using dynamic constrains we will have a solution in around 2 hours, but this reconstruction is the closest to the original distribution than any other method. Although fast in comparison, is not fast enough to be considered real-time because it needs the calculation of the last sample, and last sample's solution to calculate the new one; therefore it cannot be parallelized. So the type of regularization will depend on the application (time and quality).

7.2 Technical Notes

All of the code is made using C#. The code is divided in several libraries; the code for the forward FEM was traduced and tested from [32, 33]. The code for the forward BEM was based from [36, 37]. The different used meshes were created using the geometries from the *.Tri (ECGSIM format [44]), *.STL, *.Msh files (GMESH format). The Tetrahedral elements are created from the given surfaces using GMESH [4], or by programming a specified geometry, like two spheres, and transforming it into a *.Msh File, changing the surface mesh into a volumetric mesh.

The software has the necessary tools to transform the *.Tri files into *.Msh Files and also if necessary create its own geometry files and save them in GMESH format. To represent the geometries we programmed a graphical environment using OpenGL(Open

Graphics Library) and a visual interface of 30 fps (frames per second). The 3D engine was created using the codes from [46] to implement OpenGL. The system was tested up to 1 000,000 elements using sparse matrix techniques.

7.2.1 Geometry Format

We use the *.Msh file format because is open, and easy to implement. The *.Msh files have the following format:

```

$MeshFormat
2.2 0 8
$EndMeshFormat
$Nodes
Number of Nodes (n)
1 x1 y1 z1
-----
-----
n xn yn zn
$EndNodes
$Elements
Number of Elements (n)
1 2 2 0 Surfacei Nodei Nodej Nodek
-----
-----
Elementi 4 2 0 Volumei Nodei Nodej Nodek Nodel
-----
-----
Elementn 4 2 0 Volumei Nodei Nodej Nodek Nodel
$EndElements

```

Bibliography

- [1] World Health Organization. *Global status report on noncommunicable diseases 2010*. Geneva, 2011. URL <http://www.who.int/mediacentre/factsheets/fs317/en/>.
- [2] World Health Organization. *Global atlas on cardiovascular disease prevention and control*. Geneva, 2011. URL <http://www.who.int/mediacentre/factsheets/fs317/en/>.
- [3] Azuaje F. Clifford, G.D. and P.E. McSharry. *Advanced methods and tools for ECG data analysis*. Artech House engineering in medicine & biology series. Artech House, 2006. ISBN 9781580539661.
- [4] B. Y. Kogan. *Introduction to computational cardiology : mathematical modeling and computer simulation*. Springer, New York, 2010. ISBN 978-0-387-76685-0. D’après le cours de l’auteur Introduction to computational cardiology donné à l’Université de Californie, Los Angeles.
- [5] A. L. Hodgkin and A. F. Huxley. A quantitative description of membrane current and its application to conduction and excitation in nerve. *The Journal of physiology*, 117(4):500–544, August 1952. ISSN 0022-3751.
- [6] D. Noble. A modification of the Hodgkin-Huxley equations applicable to Purkinje fibre action and pacemaker potentials. *The Journal of Physiology*, 160(2):317–352, February 1962.
- [7] V.C. Scanlon and T. Sanders. *Essentials of Anatomy and Physiology*. F.A. Davis Company, 2010. ISBN 9780803626515.
- [8] P. Davidovits. *Physics in Biology and Medicine*. Complementary Science. Elsevier Science, 2007. ISBN 9780080555935.
- [9] L. Tung. *A bi-domain model for describing ischemic myocardial D-C potentials*. PhD thesis, MIT, Cambridge, MA, 1978.

- [10] Nielsen B.F. Cai, X. and Lysaker; M. On the possibility for computing the transmembrane potential in the heart with a one shot method: An inverse problem. *Math Bioscience*, (210(2)):523–553, 2007.
- [11] Kirby R.M. MacLeod R.S. Johnson, C.R. and D Wang. Inverse electrocardiographic source localization of ischemia: An optimization framework and finite element solution. *Journal of computational physics*, (250(0)):403–24, 2013.
- [12] C. Ramanathan, R. N. Ghanem, P. Jia, K. Ryu, and Y. Rudy. Noninvasive electrocardiographic imaging for cardiac electrophysiology and arrhythmia. *Nat. Med.*, 10(4):422–428, Apr 2004.
- [13] Xin Zhang, Indiresha Ramachandra, Zhongming Liu, Basharat Muneer, Steven M Pogwizd, and Bin He. Noninvasive three-dimensional electrocardiographic imaging of ventricular activation sequence. *Am. J. Physiol. Heart Circ. Physiol.*, 289(6): H2724–H2732, December 2005. doi: 10.1152/ajpheart.00639.2005.
- [14] Bendahmane M. Lopez Rincon, A. and B. Ainseba. Two-step genetic algorithm to solve the inverse problem in electrocardiography for cardiac sources. *CMBBE: Imaging and Visualization.*
- [15] Cai X. Lines G.T. Sundnes, J. and B.F. Nielsen. *Computing the Electrical Activity in the Heart*. Monographs in computational science and engineering. Springer, 2007. ISBN 9783540334378.
- [16] C. Mitchell and D. Schaeffer. A two-current model for the dynamics of cardiac membrane. *Bulletin of Mathematical Biology*, 65(5):767–793, September 2003. ISSN 00928240. doi: 10.1016/s0092-8240(03)00041-7.
- [17] R. FitzHugh. Mathematical models of threshold phenomena in the nerve membrane. *Bulletin of Mathematical Biology*, 17(4):257–278, December 1955. ISSN 0007-4985. doi: 10.1007/bf02477753.
- [18] Taccardi B. P. Colli F., Pavarino L.F. Simulating patterns of excitation, repolarization and action potential duration with cardiac bidomain and monodomain models. *Mathematical Bioscience J.* 197, 2005.
- [19] F. Piero Colli and S. Giuseppe. Degenerate evolution systems modeling the cardiac electric field at micro-and macroscopic level. In *Evolution equations, semigroups and functional analysis*, pages 49–78. Springer, 2002.
- [20] Bendahmane M. and Karlsen K.H. Analysis of a class of degenerate reaction-diffusion systems and the bidomain model of cardiac tissue. *Netw. Heterog. Media* 1, 2006.

- [21] Igarashi T. Haraguchi R. Takayama, K. and K. Nakazawa. A sketch-based interface for modeling myocardial fiber orientation. In Andreas Butz, Brian D. Fisher, Antonio Krüger, Patrick Olivier, and Shigeru Owada, editors, *Smart Graphics*, volume 4569 of *Lecture Notes in Computer Science*, pages 1–9. Springer, 2007. ISBN 978-3-540-73213-6.
- [22] Dubé B. Richer J. Vinet A. Potse, M. and R. M. Gulrajani. A comparison of monodomain and bidomain reaction-diffusion models for action potential propagation in the human heart. *IEEE Trans. Biomed. Engineering*, 53(12):2425–2435, 2006.
- [23] F. Fenton and A. Karma. Vortex dynamics in three-dimensional continuous myocardium with fiber rotation: Filament instability and fibrillation. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 8(1):20–47, March 1998. ISSN 1089-7682. doi: 10.1063/1.166311.
- [24] H. Zhang W.W. Hager. A survey of nonlinear conjugate gradient methods. *Pacific J. Optim.*, 2:35–58, 2006.
- [25] E.L. Stiefel M.R. Hestenes. Methods of conjugate gradients for solving linear systems. *J. Research Nat. Bur. Standards*, 49:409–436, 1952.
- [26] J. Sundnes, G. T. Lines, and A. Tveito. An operator splitting method for solving the bidomain equations coupled to a volume conductor model for the torso. *Mathematical biosciences*, 194(2):233–248, April 2005.
- [27] P. C. Hansen. The l-curve and its use in the numerical treatment of inverse problems. In *Computational Inverse Problems in Electrocardiology*, ed. P. Johnston, *Advances in Computational Bioengineering*, pages 119–142. WIT Press, 2000.
- [28] Kirby R. M. Wang, D. and C. R. Johnson. Resolution strategies for the finite-element-based solution of the ecg inverse problem. *IEEE Trans. Biomed. Engineering*, 57(2):220–237, 2010.
- [29] O. A. Ladyzhenskaya and Ural'tseva N. N. *Linear and quasilinear elliptic equations*. Academic Press, New York, 1968.
- [30] Amaral L. A. N. Glass L. Hausdorff J. M. Ivanov P. Ch. Mark R. G. Mietus J. E. Moody G. B. Peng C.-K. Goldberger, A. L. and H. E. Stanley. PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals. *Circulation*, 101(23):e215–e220, 2000 (June 13). Circulation Electronic Pages: <http://circ.ahajournals.org/cgi/content/full/101/23/e215> PMID:1085218; doi: 10.1161/01.CIR.101.23.e215.

-
- [31] Bendahmane M. Ainseba, B. and A. Lopez. Software for inverse voltage calculations in the heart's surface. In *Electrical Communications and Computers (CONIELECOMP), 2012 22nd International Conference on*, pages 34–39, 2012.
- [32] *Understanding and Implementing the Finite Element Method*. PA: Society for Industrial and Applied Mathematics, 2006.
- [33] *Introduction to the Finite Element Method in Electromagnetics*. Morgan and Claypool, 2006.
- [34] W.T. Ang. *A Beginner's Course in Boundary Element Methods*. Universal Publishers, 2007. ISBN 9781581129748.
- [35] P. Hunter. Fem/bem notes. Technical report, 1998.
- [36] S. Nintcheu Fata and L. J. Gray. Semi-analytic integration of hypersingular galerkin bies for three-dimensional potential problems. *J. Computational Applied Mathematics*, 231(2):561–576, 2009.
- [37] S. Nintcheu Fata. Boundary integral approximation of volume potentials in three-dimensional linear elasticity. *J. Computational Applied Mathematics*, 242:275–284, 2013.
- [38] S. Nintcheu Fata. Explicit expressions for three-dimensional boundary integrals in linear elasticity. *J. Computational Applied Mathematics*, 235(15):4480–4495, 2011.
- [39] S. Nintcheu Fata. Semi-analytic treatment of the three-dimensional poisson equation via a galerkin bie method. *J. Computational Applied Mathematics*, 236(6):1216–1225, 2011.
- [40] J. L Lions. *Contrôlabilité Exacte, Perturbation et Stabilisation de Systèmes Distribués*. Masson, 1986.
- [41] A. N. Tikhonov. *Regularization of incorrectly posed problems*. Soviet Math, Dokl, 1963.
- [42] C. Ramanathan and Y. Rudy. Electrocardiographic imaging: II. Effect of torso inhomogeneities on noninvasive reconstruction of epicardial potentials, electrograms, and isochrones. *J. Cardiovasc. Electrophysiol.*, 12(2):241–252, Feb 2001.
- [43] Iskandrian A.E. Cerqueira, M.D. and E.V. Garcia. *Atlas of Nuclear Cardiology. Imaging Techniques to Braunwald's Heart Disease Series*. Elsevier - Health Sciences Division, 2011. ISBN 9781416061342.

-
- [44] A. van Oosterom and G. J. Huiskamp. The effect of torso inhomogeneities on body surface potentials quantified using "tailored" geometry. *Journal of Electrocardiology*, 22(1):53–72, January 1989. ISSN 00220736. doi: 10.1016/0022-0736(89)90023-x.
- [45] S. N. Sivanandam and S. N. Deepa. *Introduction to Genetic Algorithms*. Springer Publishing Company, Incorporated, 1st edition, 2007. ISBN 354073189X, 9783540731894.
- [46] Colin P. Fahey. C sharp opengl interface. URL http://www.colinfahey.com/csharp_wrapper_for_opengl/csharp_wrapper_for_opengl.html.