# Efficient Cardiac Cell Solvers and Simulating Temperature Dependence in the Myocardium

by

Justin Grenier

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science

Department of Biomedical Engineering

University of Alberta

# Abstract

The modelling of the electrical activity in cardiac tissue can enable researchers to study heart phenomena such as arrhythmias which are difficult to observe in vivo. However, the nature of the mathematical equations used to represent these behaviors present multiple numerical difficulties which limit their large scale usage. In this thesis, we have suggested the application of a nested implicit Runge-Kutta method of order 4 (NIRK4) as a means of efficiently solving the stiff cardiac cell models. By comparing its performance to multiple common implicit and explicit solvers, we have established the advantages of using NIRK4 when solving multiple cell models of varying complexities. Due to recent experimental and modelling results, it was deemed advantageous to investigate the effects of the temperature on the transmembrane potential. Therefore, we developed a mathematical model by coupling Pennes' bioheat equation to the bidomain model to simulate the induced heat caused by cardiac action potential. The influence of the temperature on ionic conductances in the Aliev-Panfilov and Luo-Rudy cardiac cell models was also investigated. Furthermore, the induced heat caused by the transmembrane potential's propagation was studied as a potential method for detecting spiral-waves.

# Acknowledgements

This work would not have been possible without the hard work and support of many individuals. First off, I would like to sincerely thank my supervisor Dr. Belhamadia for his guidance and encouragement throughout the project. My gratitude must also be extended to my supervisory committee for offering advice on the direction of my studies. Moreover, I extend my appreciation to the GIREF group at l'Université Laval for allowing me the usage of their numerical and computational resources. Furthermore, I would like to thank the Government of the province of Alberta as well as the Heart and Stroke foundation for their generous financial support. Finally, without the love and support of my mother, father, sister and girlfriend Kayla I would have had little strength to complete this task.

# Contents

# List of Tables

# List of Figures

# Abbreviations

| | |
|---|---|
| ODE | Ordinary Differential Equation |
| PDE | Partial Differential Equation |
| RK4 | Runge-Kutta method of order 4 |
| NIRK4 | Nested Implicit Runge-Kutta method of order 4 |
| SDIRK4 | Singly Diagonally Implicit Runge-Kutta method of order 4 |
| ESDIRK3 | Explicit Singly Diagonally Implicit Runge-Kutta method of order 3 |
| RL | Rush-Larsen method |
| RL2 | Rush-Larsen method of order 2 |
| APD | Action Potential Duration |
| CV | Conduction Velocity |

# Chapter 1

# Introduction

In 2007 heart diseases were the leading cause of death in the United States with over 600 thousand deaths and a total mortality cost of over 80 billion dollars [1]. This exceeded the next cause of death, cancer, by nearly 50 thousand deaths. In particular, sudden cardiac death due to ventricular fibrillation (VF) is the most common and often the first manifestation of a coronary heart disease. Due to the long periods of clinical validation and often prohibitive cost of developing reliable new treatments for heart pathological conditions, the pace of successful cardiac research is slowed. With the recent developments in the fields of scientific computing and numerical techniques, the realistic applications of cardiac mathematical modelling is increasing in use in modern medicine as these models can allow researchers to study complex medical questions.

Electrocardiology models can be used to describe both the cellular and tissue domain of the electrical activity in the heart. Cellular models seek to accurately model the electrical activity of a single cardiac cell once a stimulus current has been applied. These models predict the resulting potential difference across the cell membrane based on a series of gating variables and multiple ionic currents. Such models are known as ionic models and they are often represented as electrical circuits which are then transformed into a system of ordinary differential equations (ODEs). This approach is based on the

1963 Nobel prize winning work of Hodgkin and Huxley which accurately modelled the electrical potential across a giant squid axon [2]. Since then, many models of varying complexity and accuracy have been developed. Hence, as more complex models induced more variables to be added to increase the physiological accuracy, the complexity in the numerical solving of these models is also increased. The majority of difficulties that arise in solving these ODE systems are due to the fact that they are numerically very stiff. This means that numerical stability is often the limiting factor in solving these equations. To overcome the computational difficulties for solving cardiac cell models efficient solvers must be used. The performance of these solvers will be heavily studied in this thesis.

Most clinical applications for modelling the electrical activity in the heart necessitate an accurate prediction of the propagation of this wave throughout the whole heart tissue. This includes modelling defibrillation which can be difficult to observe in vivo[3] and which is thus one of the prime target of cardiac clinicians. Since a large number of pathological conditions are associated with the synchronous and dynamically relevant representation of the large number of cells represented as tissue, simple cellular models are often not adequate to reveal pathological phenomena to clinical researches. This means that the cellular models must be coupled with partial differential equations (PDEs) in order to accurately describe the propagation of the voltage potential difference through the myocardial tissue. Two popular models are often used for this purpose: the monodomain and bidomain models. The bidomain model, which became popular in the late 1970s, accounts for both the extra and intracellular voltage potential. This model currently gives the best reflection of electrophysiological waves in cardiac tissue and is the closest approximation to the observed experimental data [4]. The bidomain model considers a system of elliptic and parabolic non-linear PDEs, one for each of the intra- and extracellular potentials, coupled with the ionic models. Unfortunately, this

complexity leads to a model which is computationally intensive and algorithmically challenging to be realized. To partially reduce these challenges, the monodomain model can be employed which consider equal isotropic ratios of the extra- and intracellular domains and thus the two non-linear PDEs are reduced to one parabolic PDE. This model is widely employed to describe, for instance, the dynamics of a general excitable cardiac tissue. At the expense of limited capabilities, the monodomain model is still adequate in some cases such as reproducing the electrocardiogram (ECG). By coupling the monodomain or bidomain model with the various cellular models available, it is possible to study the propagation of the electrical wave throughout the heart using realistic three dimensional geometries. However, the extreme computational demand of solving these equations to obtain accurate and realistic results present several challenges. These difficulties often necessitates the usage and the development of various numerical techniques such as splitting methods, parallel solvers and adaptive mesh techniques etc.

Previous research has also discovered the significant effect of the temperature on the behavior of the transmembrane potential. From the earliest models, such as Hodgkin-Huxley, it has been demonstrated that the temperature can have a definite influence on the movement of different ions through the membrane channels [2]. Due to these changes, the duration of which the action potential is stimulated can vary considerably depending on the temperature of the environment. This action potential duration, known as APD, is a very important characteristic of cardiac modelling. It is now believed that changes in APDs can cause electrical alternans which are waves of electric potentials of alternating durations. The reason for which these alternans are of special interest is that this behavior has been strongly linked with the development of cardiac arrhythmias[24]. Therefore it is meaningful to develop an accurate model of cardiac tissue which includes temperature effects in order to facilitate the study of these phenomena.

Two specific goals are therefore presented in this thesis. First we will present and investigate the performance of a new solver for cardiac cell models. Our results will be compared with many of the most popular solvers used in this application. Second, we will develop a realistic cardiac model in order to investigate the effects of temperature on the transmembrane potential.

This thesis is structure as follows. In chapter 2, we will present an overview of the modelling techniques for the cardiac action potential which includes both the cellular and tissue domains. In chapter 3, we will introduce multiple methods for solving the ODEs and PDEs which arise in the models as well as a possible coupling technique for these equations. Furthermore, in chapter 4 we will introduce a new numerical technique for solving cardiac cell models and compare its performance to that of many other popular methods. Moreover, in chapter 5, we will propose a model for cardiac tissue which includes temperature and demonstrate its effect on the action potential duration. Finally, chapter 6 will offer the conclusion.

# Chapter 2

# Electrical Activity in the Heart

In the heart, the contractions are regulated by the propagation of an electric signal through the whole myocardium. Problems and irregularities in this electric wave can lead to many potentially fatal changes in blood flow such as ventricular fibrillation. Therefore, it is important to diagnose and further study the mechanisms which can change this electrical signal. At the cellular level, the potential difference across the semi-permeable membrane is caused by differences in ionic concentrations across this structure. Furthermore, the passage of many ions are dictated by channels in the membrane which are often voltage controlled. Thus, a voltage above a certain threshold can allow the passage of ions which change the concentration and therefore the potential difference across this membrane. It is through this mechanism that the electric wave is propagated from cell to cell and finally cause the flow of contractions in the muscle which enable the pumping action of the heart. Figure 2.1 illustrates a simplified example of the potential difference caused by a difference in ionic concentration across a semi-permeable membrane.

In this chapter, we will introduce the fundamentals of cardiac cell modeling. An overview of most ionic and simplifed models which will be used further in the thesis will be presented in section 2.1. Meanwhile, section 2.2 will be devoted

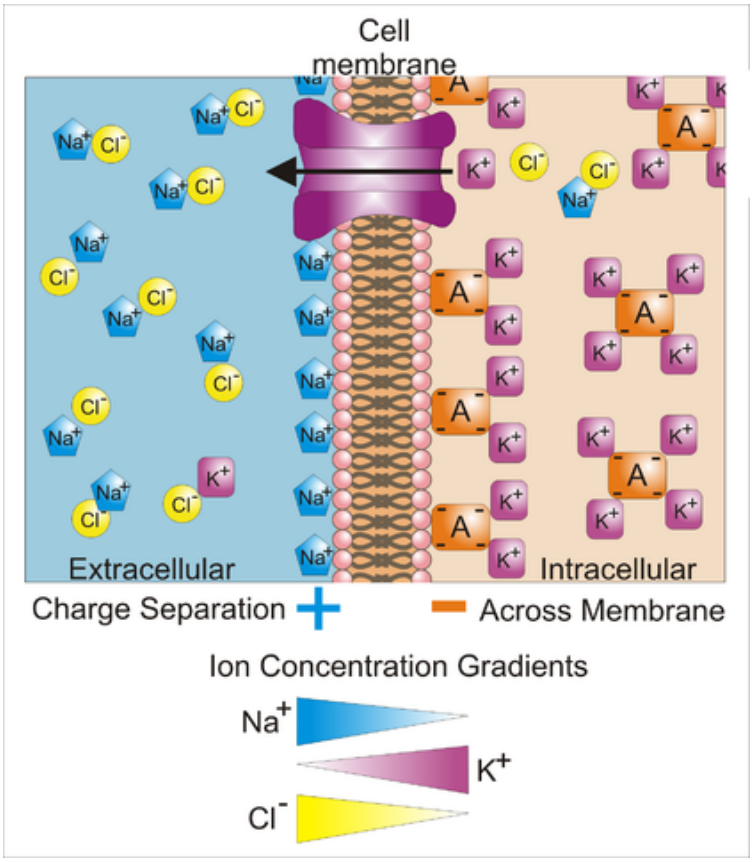



Figure 2.1: Illustration of the potential difference in a cell caused by flow of ionic currents[1]

to an introduction of the Bidomain and simplified Monodomain tissue models which must be coupled with the cell models in order to obtain accurate results for myocardial simulations.

## 2.1 Cardiac Cell Models

The most fundamental step when modeling the electrical activity in the heart is simulating the change in potential difference across the membrane at the cellular level. Various mathematical models exist which differ by complexity, accuracy and type of tissue. In this thesis, we will be using five different cell

[1] `http://en.wikipedia.org/wiki/Membrane_potential`

models and two simplified models: Hodgkin-Huxley, Fox, Luo-Rudy, Courtemanche, Tusscher, FitzHugh-Nagumo and Aliev-Panfilov. These models are used throughout literature and present different computational difficulties. A more complete description of these models are presented in the following sections.

### 2.1.1 Hodgkin-Huxley

In 1963, Alan Hodgkin and Andrew Huxley were awarded the Nobel prize in Physiology or Medicine "for their discoveries concerning the ionic mechanisms involved in excitation and inhibition in the peripheral and central portions of the nerve cell membrane"[5]. Using the axon of a giant squid, they devised a method for modelling the potential difference across a cell membrane which has become the basis for all subsequent modern cardiac cell models. Their methods involved first representing the cell membrane as an electric circuit as seen in Figure 2.2.

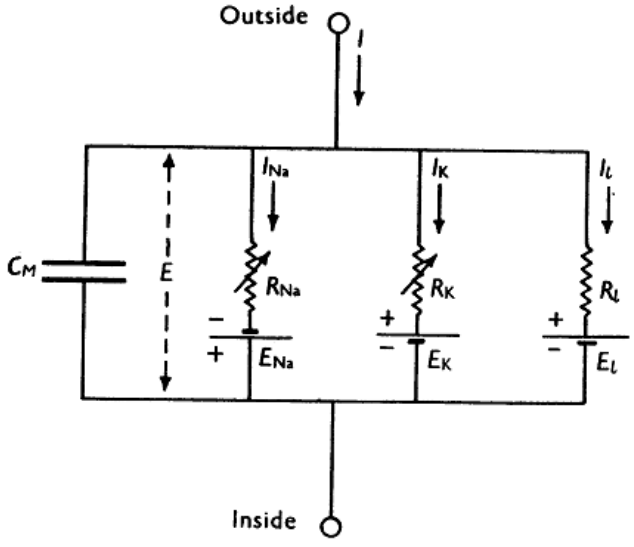



Figure 2.2: Shows a visualization of the circuit used by Hodgkin and Huxley when modeling the membrane potential of a squid giant axon [2]

From basic circuit analysis we can see that the total current density $I_{tot}$ is equal to:

$$I_{tot} = C_m \frac{\partial E}{\partial t} + I_{ion} \tag{2.1}$$

where $I_{ion} = I_{Na} + I_K + I_l$, of which $I_{Na}$ corresponds to the sodium current, $I_K$ to the potassium current and $I_l$ to a leftover leak current. $E$ represents the potential difference across the membrane which, in order to retain consistency with other notations, we will name $V$ for the remainder of the derivation. According to [6] after $200\mu s$ the $I_{tot}$ becomes negligible and therefore equation 2.1 can be simplified as:

$$\frac{\partial V}{\partial t} = -\frac{I_{ion} + I_{stim}}{C_m} \tag{2.2}$$

where $I_{stim}$ is some initial stimulus current. From the circuit diagram 2.2 we can also see that the three ionic currents can be represented as:

$$I_{Na} = g_{Na}(V - E_{Na}) \tag{2.3}$$

$$I_K = g_K(V - E_K) \tag{2.4}$$

$$I_l = g_l(V - E_l) \tag{2.5}$$

Furthermore, it was considered that the currents $I_Na$ and $I_K$ were time-dependent whereas $I_l$ was time independent and therefore had a linear relationship with the potential difference. However, for the time-independent currents we can express the conductances as a function of gating variables which represent the probability of the opening or closing of the ion specific voltage-activated channels. This can be represented in the following form:

$$g = g_{max}x \tag{2.6}$$

where $g_{max}$ is the maximum conductance and $x$ is a gating variable which fluctuates between 0 (fully closed channel) and 1 (fully opened channel) and

can be described in the form of:

$$\frac{dx}{dt} = \alpha_x(1-x) - \beta_x x \tag{2.7}$$

where $\alpha_x$ and $\beta_x$ are voltage dependent rate constants. For the Hodgkin-Huxley model it was determined that the experimental data best fit when using three gating variables. Therefore, the full mathematical representation is described by the following four differential equations:

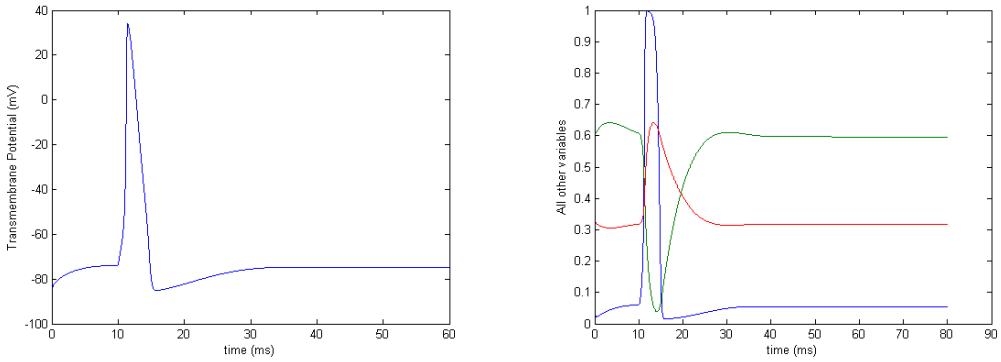
$$\frac{dV}{dt} = \frac{-(I_{Na} + I_K + I_l + I_{stim})}{C_m} \tag{2.8}$$

$$\frac{dm}{dt} = \alpha_m(1-m) - \beta_m m \tag{2.9}$$

$$\frac{dh}{dt} = \alpha_h(1-h) - \beta_h h \tag{2.10}$$

$$\frac{dn}{dt} = \alpha_n(1-n) - \beta_n n \tag{2.11}$$

For the full model please refer to Appendix A and for the waveform of the transmembrane potential please see Figure 2.3.




(a) Transmembrane potential (b) Recovery variables

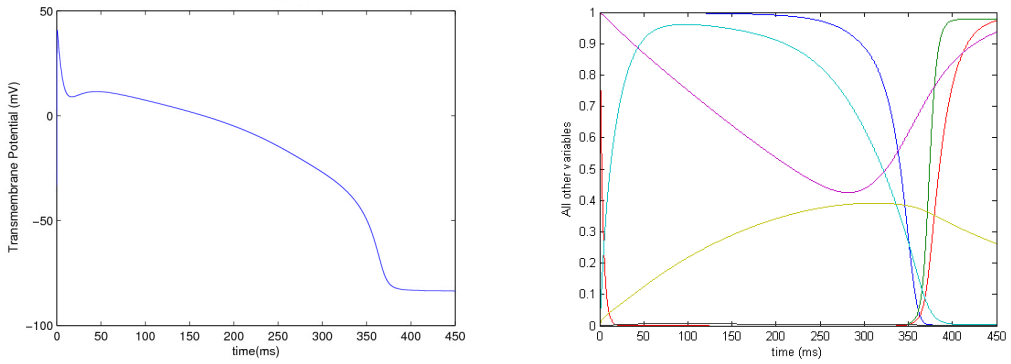
Figure 2.3: Waveforms for the Hodgkin-Huxley Model

### 2.1.2 Luo-Rudy I

In 1991, Ching-hsing Luo and Yoram Rudy developed the more complex Luo-Rudy I model based on the most recent experimental data obtained on mammalian ventricular tissue [7]. Their approach was similar to the that of Hodgkin and Huxley, however, more ionic currents were considered:

$$I_{ion} = I_{Na} + I_{si} + I_K + I_{K1} + I_{Kp} + I_b \tag{2.12}$$

Where $I_{K1}$, $I_{Kp}$ and $I_b$ are time independent components of the potassium current. Furthermore, $I_{si}$ is the slow intake current which is the result of the movement of calcium ions across the membrane. In total, the Luo-Rudy model consists of a system of 8 ordinary differential equations and is presented more fully in Appendix A. The waveform of the transmembrane potential and recovery variables can be seen in Figure 2.4.




(a) Transmembrane potential (b) All other variables

Figure 2.4: Waveforms for the Luo-Rudy Model

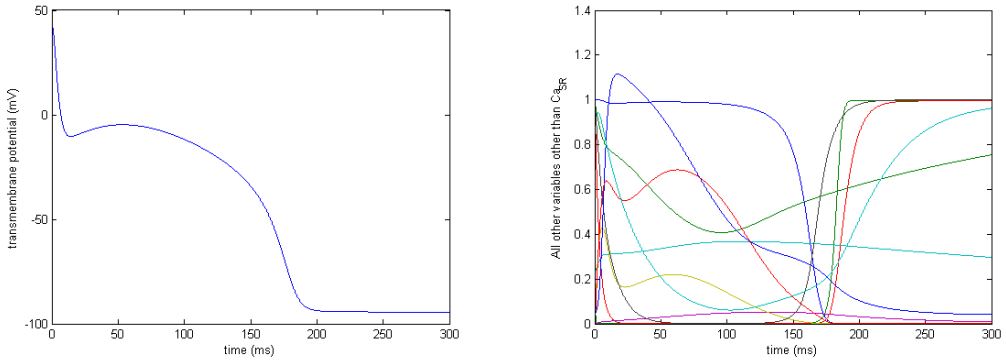
### 2.1.3 Fox model

In 2001, Jeffrey Fox, Jennifer McHarg and Robert Gilmour presented the Fox model of canine myocyte [9]. Built upon previous canine models, the Fox model

specifically addresses the inability of previous models in simulating stable electrical alternans. Alternans describe a phenomena where the duration of the action potential alternates between cycles. This is an especially interesting abnormality for cardiac modelers as there is evidence that alternans are a precursor to ventricular arrhythmia and therefore ventricular fibrillation [8]. This model uses 13 currents to describe the movement of ions across the membrane:

$$\begin{aligned} I_{ion} =& I_{Na} + I_{Kl} + I_{Kr} + I_{Ks} + I_{to} + I_{Kp} + I_{NaK} \\ &+ I_{NaCa} + I_{Nab} + I_{Cab} + I_{pCa} + I_{Ca} + I_{CaK} \end{aligned} \tag{2.13}$$

where $I_{Na}$ and $I_{Nab}$ describe the sodium current, $I_{K1}$, $I_{Kr}$, $I_{Ks}$, $I_{to}$ and $I_{Kp}$ describe the potassium current and $I_{Ca}$, $I_{pCa}$ and $I_{Cab}$ describe the calcium current. Furthermore, $I_{CaK}$ is the potassium current through the calcium channel, $I_{NaCa}$ is the sodium and calcium exchange current and $I_{NaK}$ is the sodium and potassium pump current. The model consists of a system of 13 ODEs and the full model can be found in Appendix A. The waveform of the transmembrane potential are recovery variables are presented in Figure 2.5.




(a) Transmembrane potential (b) All other variables excluding $Ca_{SR}$

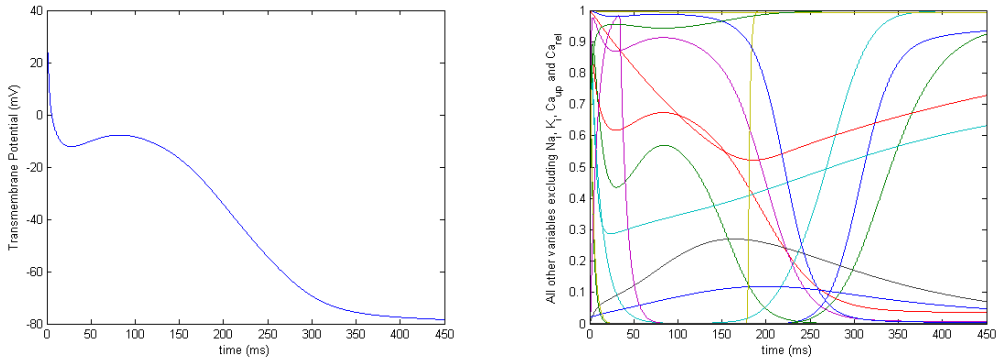
Figure 2.5: Waveforms for the Fox Model

### 2.1.4 Courtemanche model

In 1999, Marc Courtemanche, Rafael Ramirez and Stanley Nattel presented their human atrial myocyte model. The motivation of this model was to address significant differences in physiology between human and animal data which had previously been used when modelling atrial tissue [10]. This could help further understand various different disorders in this heart region such as atrial fibrillation. Their model consists of 12 ionic currents:

$$\begin{aligned} I_{ion} =& I_{Na} + I_{K1} + I_{to} + I_{Kur} + I_{Kr} + I_{Ks} + I_{CaL} \\ &+ I_{pCa} + I_{NaK} + I_{NaCa} + I_{Nab} + I_{Cab} \end{aligned} \tag{2.14}$$

These currents have similar definitions as found in the Fox model other than $I_{Kur}$ which corresponds to ultrarapid potassium current. This model consists of 21 ODEs and can be found in Appendix A. The waveforms of the transmembrane potential and recovery variables can be found in Figure 2.6.




(a) Transmembrane potential

(b) All other variables excluding $Na_i$, $K_i$, $Ca_{up}$ and $Ca_{rel}$

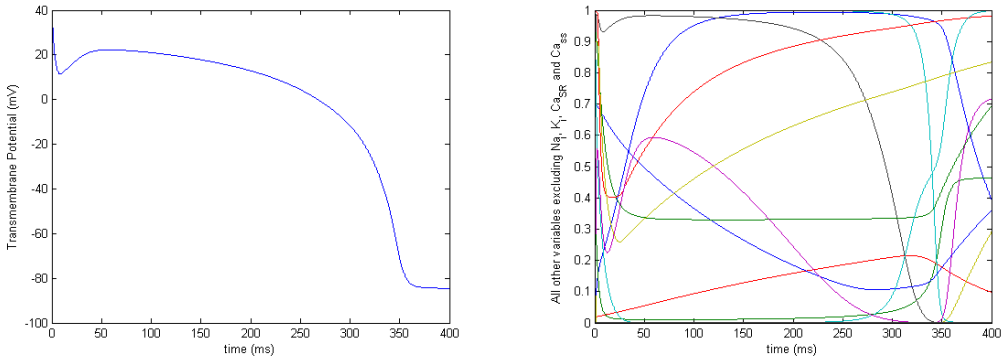
Figure 2.6: Waveforms for the Courtemanche Model

### 2.1.5 Tusscher model

Another model developed to study the effect of electrical alternans was that of Ten Tusscher and Alexander Panfilov in 2006 [11]. Unlike the Fox model, Tusscher used data from human mid-myocardial tissue to formulate the series of equations. This included a more complex handling of the calcium dynamics than their previous models. This includes 12 ionic currents as follows:

$$\begin{aligned} I_{ion} =& I_{K1} + I_{to} + I_{Kr} + I_{Ks} + I_{CaL} + I_{NaK} \\ &+ I_{Na} + I_{Nab} + I_{NaCa} + I_{Cab} + I_{Kp} + I_{pCa} \end{aligned} \tag{2.15}$$

These currents have the same definition as in the previous models discussed. This model is also considered more computationally stiff and therefore harder to solve than the previous models [29]. For the full discussion on stiffness please refer to Chapter 3. This model presents a system of 18 ODEs that must be solved to obtain the transmembrane potential. An example of the waveform of the potential and recovery variables can be seen in Figure 2.7. Furthermore, the full model can be found in Appendix A.




(a) Transmembrane potential

(b) All other variables excluding $Na_i$, $K_i$, $Ca_{SR}$ and $Ca_{ss}$

Figure 2.7: Waveforms for the Tusscher Model

### 2.1.6 Simplified Ionic Models

Depending on the intent of the study or the physiological phenomena that is to be modelled, it is not always necessary to model the entire ionic cellular function when simulating the tissue. Specifically, it can become nearly impossible to obtain numerical results in a reasonable time-frame when attempting to simulate a complex tissue. In these cases, it is advantageous to use simpler two variable models which attempt to simulate the behaviour of the cellular electric potential without requiring the solving of large systems of ODEs. Two popular examples of these are the FitzHugh-Nagumo and the Aliev-Panfilov models.

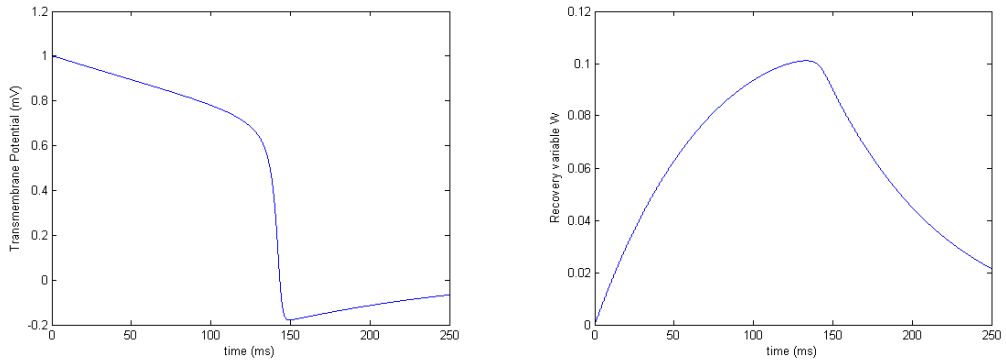
#### FitzHugh-Nagumo

The FitzHugh-Nagumo model first introduced in 1961 gives a fair representation of the waveform [12]. This model uses both an excitation and recovery variable to model the polarisation and repolarisation of the action potential. Although the form of the equations has remained quite consistent, the values of the constants have been modified multiple times in order to increase the accuracy of the waveform. Therefore, only two odes must be solved to obtain the action potential:

$$\frac{dV}{dt} = kV(V-a)(1.0-V) - W \tag{2.16}$$

$$\frac{dW}{dt} = \epsilon(\gamma U - \beta W) \tag{2.17}$$

where V is the membrane potential, W is the recovery variable and the constants are chosen to obtain the desired form of the potential. The values for the constants employed in this thesis are similar to the ones used in Belhamadia et al. [13]. An example of the shape of the transmembrane potential and recovery variable can be seen in Figure 2.8




(a) Transmembrane potential (b) Recovery variable W

Figure 2.8: Waveforms for the Fitz-Nagumo Model

### Aliev-Panfilov

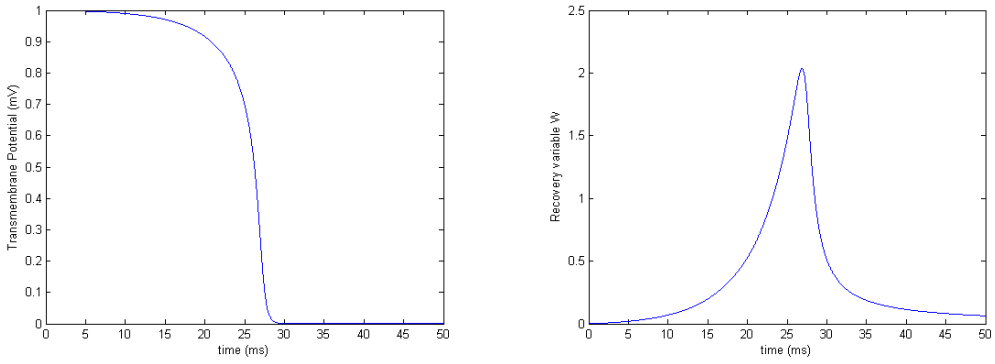
Building upon the FitzHugh -Nagumo model, Rubin Aliev and Alexander Panfilov developed the Aliev-Panfilov simplified model in 1996 in order to improve the shape of the action potential and address the restitution properties of cardiac tissue [17]. Thus, the model addresses the relation between cycle length, which is the time between heart beats, and the length of the duration of the action potential. Once again this is important when attempting to simulate arrhythmias in cardiac tissues. This simplified model is especially useful as modelling the effects of changes in action potential duration often requires long simulation times and therefore ionic models might be too computationally intensive to feasibly be taken. As with the FitzHugh-Nagumo model Aliev-Panfilov only requires the solving of 2 ODEs:

$$\frac{dV}{dt} = kV(V-a)(1-V) - (VW) \tag{2.18}$$

$$\frac{dW}{dt} = (\epsilon + \frac{u_1 W}{u_2 + V})(-W - (kV(V-a-1))) \tag{2.19}$$

Where V is the transmembrane potential and W is the recovery variable. Once again the values for the constants employed in this thesis are similar to

the ones used in Belhamadia et al. [13]. An example of the waveform of the transmembrane potential and recovery variable can be seen in Figure 2.9.




(a) Transmembrane potential (b) Recovery variable W

Figure 2.9: Waveforms for the Aliev-Panfilov Model

## 2.2 Cardiac Tissue Models

The human heart can contain hundreds of millions of cells. The orientation of these muscle cells in the heart is quite complex. Using very recent diffusion tensor magnetic resonance imaging results, Poveda et al. [50] have postulated that the myocardial fibers have a helical architecture as seen in figure 2.10. For further accuracy, it is possible to take this structure into account in tissue modelling as was performed by Marcé-Nogué et al. [51] in 2013. However, for simplicity and due to the fact that we will not be simulating 3-dimensional results in this thesis, we will be assuming a homogeneous arrangement of the cells in tissue.

A possible method for modelling the electrical activity at the organ level is to simply simulate each individual cell and apply coupling between them. However, due to the extremely computationally expensive nature of this method and the unnecessary level of accuracy it is generally preferable to employ a less

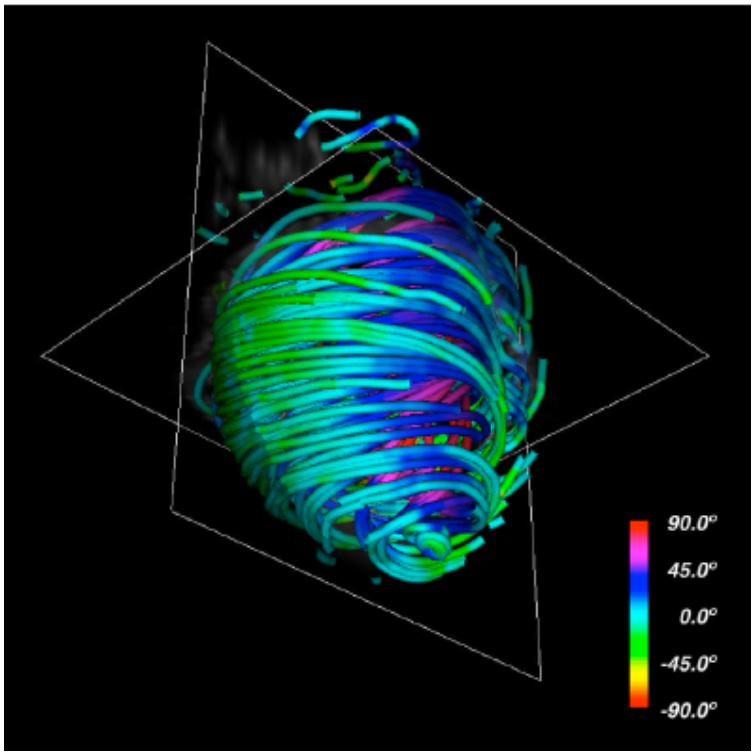



Figure 2.10: Diffusion tensor image showing the helical orientation of myocardial fibers as presented in [50]

brute force approach. In biomedical modelling we often group a large collection of cells as volume conductors which have the average values of the cells contained in the region. In this context, it is necessary to split the volume into sections which are small enough for the behavior we are trying to study but large in comparison to the actual cellular level. Two methods which utilize this approach at the tissue scale are the bidomain and monodomain models.

### 2.2.1 Bidomain Model

In 1978, Leslie Tung developed a volume conductors approach for simulating myocardial tissue [22]. This method splits the domain into two parts: the intracellular domain represents the potential inside the cell while the extracellular domain considers the electric potential outside of the cell. The following series of equations are the standard bidomain equations for cardiac tissue:

$$\nabla \cdot (M_i \nabla v) + \nabla \cdot (M_i \nabla u_e) = \chi C_m \frac{\partial v}{\partial t} + \chi I_{ion} \tag{2.20}$$

$$\nabla \cdot (M_i \nabla v) + \nabla \cdot ((M_i + M_e) \nabla u_e) = 0 \tag{2.21}$$

Although many complex boundary conditions have been proposed, for simplicity the following Neumann boundary conditions are used:

$$n \cdot (M_i \nabla v) = 0 \tag{2.22}$$

$$n \cdot (M_e \nabla u_e) = 0 \tag{2.23}$$

However, in order to avoid numerical problems when solving equation 2.21 it is, for instance, necessary to ground one of the sides with the following Dirichlet boundary condition

$$u_e(0) = 0 \tag{2.24}$$

For a more thorough exploration of boundary conditions in the bidomain model please refer to "Computing the Electrical Activity in the Heart" [15].

In the above equations $M_i$ and $M_e$ are the conductivities in the intracellular and extracellular domains respectively. The electric potentials $u_e$, $u_i$ and $v$ are for the extracellular, intracellular and transmembrane regions where $v = u_i - u_e$. Furthermore, $C_m$ is the capacitance of the cell membrane, $\chi$ is the membrane area to volume ratio and $I_{ion}$ is the ionic current which we obtain from the cellular models. Finally $n$ is simply the normal to the boundary. However, a system of partial differential equations is often very complex to solve and therefore a simpler tissue model is regularly desired to obtain more reasonable computational time. This simplified model is known as the monodomain tissue model.

### 2.2.2 Monodomain Model

In certain instances it is possible to assume that the conductivities $M_i$ and $M_e$ have equal anisotropy ratios. This means that they are equal to one proportional to the other and therefore $M_e = \lambda M_i$ where $\lambda$ is a scalar. If this assumption is made we can simplify the bidomain equations 2.20 to 2.23 to the following equation:

$$\frac{\lambda}{1+\lambda}\nabla \cdot (M_i \nabla v) = \chi C_m \frac{\partial v}{\partial t} + \chi I_{ion} \tag{2.25}$$

with boundary conditions:

$$n \cdot (M_i \nabla v) = 0 \tag{2.26}$$

where $\lambda$ is a scalar and all other variables have the same definition as described in the previous section. Although the monodomain equations are much less complex, they do have limitations. First off, it is very difficult to chose a proper value of $\lambda$ in order to obtain proper behavior in the myocardium. Furthermore, equal anisotropy is not the case in human tissue and many electrophysiological conditions will not manifest themselves in simulations using this simplified model.

Nevertheless, both the monodomain and bidomain models alone are not adequate in simulating the complex cardiac behavior for realistic heart geometries. As can be seen in equations 2.20 and 2.25 the tissue models contain a variable for the ionic current passing through the membrane. These currents are described by the ionic and simplified cell models that were discussed in section 2.1. Since these cell models are represented by a system of ODEs while the tissue models involve a series of PDEs, it is necessary to use a splitting method in order to enable the solving of both systems simultaneously. This splitting technique as well as the numerical methods used for solving the ODEs and PDEs will be discussed in chapter 3.

# Chapter 3

# Numerical Methods

In chapter 2, we presented a mechanism for representing the complex behavior of cardiac electrical activity in a usable mathematical form. However, the derived equations contain series of coupled ODEs and PDEs which cannot be solved analytically. Furthermore, the nature of these equations can be very computationally complex and therefore efficient and robust solvers must be used in order to obtain accurate solutions within a reasonable time frame. In this chapter, we will first be detailing multiple ODE solvers which have previously been used in literature for solving the cardiac cell models. This will be followed by a presentation of a PDE solver well suited for both the Monodomain and Bidomain tissue models. Finally we will offer a splitting method which will enable us to obtain results from the coupled tissue and cell models.

## 3.1 Solving Ordinary Differential Equations

When simulating the cardiac cell models previously discussed in chapter 2, it is necessary to solve a system of ordinary differential equations which have the following form:

$$\frac{dy}{dt} = f(y, t) \tag{3.1}$$

Since the cardiac cell ODEs model are non-linear, it is not possible to solve them analytically and therefore a robust solver much be developed. In this section, we will first be introducing basic ODE solving concepts such as the theory behind the Forward Euler and Runge-Kutta methods. This will then be followed by describing more complex efficient solvers better suited for cardiac cell models.

### 3.1.1 Forward Euler Method

The Forward Euler method is the simplest and most common numerical solver for ordinary differential equations. This method is based on considering the derivative of a function at a point as equal to its slope at that location. Therefore, we have the following equation:

$$\frac{dy}{dt} = f(t, y) \approx \frac{\Delta y}{\Delta t} \tag{3.2}$$

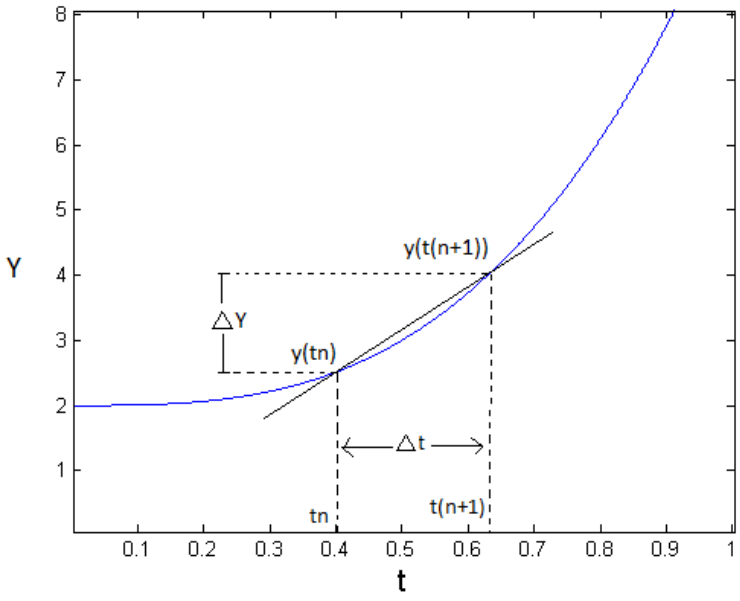



Figure 3.1: Example of a function that will be solved using the Euler Method

From Figure 3.1, we can see that the slope can be approximated as:

$$f(t_n, y_n) \approx \frac{y_{n+1} - y_n}{t_{n+1} - t_n} \tag{3.3}$$

By rearranging and using an iteration step size of $h = \Delta t$ we obtain the following form for the explicit Euler method:

$$y_{n+1} = y_n + hf(t_n, y_n) \tag{3.4}$$

Therefore, if we have an initial value we can obtain an approximate solution for the value $y$. However, this method often requires very small step sizes to obtain accurate results which is computationally expensive. Thus, higher order methods should be considered. These will be presented in the next section.

### 3.1.2 Runge-Kutta Methods

To increase the order of accuracy of the Euler method, Taylor series expansion can be used to create a wide range of ODE solvers with a varying degree of accuracy. When studying Runge-Kutta methods it is first necessary to familiarize ourselves with the Butcher tableau. This tableau is a popular, simple representation of the more traditional long formed equations of Runge-Kutta methods. They are presented in the following form:

$$\begin{array}{c|cccc} c_1 & a_{11} & a_{12} & \dots & a_{1m} \\ c_2 & a_{21} & a_{22} & & \vdots \\ \vdots & \vdots & & \ddots & \vdots \\ c_m & a_{m1} & \dots & \dots & a_{mm} \\ \hline & b_1 & b_2 & \dots & b_m \end{array} \tag{3.5}$$

The entries of a Butcher tableau is all that is needed to construct all the Runge-Kutta methods. The intention is to use these entries and apply them

to the following equations:

$$y_{n+1} = y_n + \sum_{i=1}^{m} b_i k_i \tag{3.6}$$

$$k_i = hf(t_i + c_i h, y_n + h \sum_{j=1}^{m} a_{ij} k_j) \tag{3.7}$$

For instance, the classic fourth order Runge-Kutta method (RK4) can be represented by using the following Butcher tableau:

$$\begin{array}{c|cccc} 0 & & & & \\ 1/2 & 1/2 & & & \\ 1/2 & 0 & 1/2 & & \\ 1 & 0 & 0 & 1 & \\ \hline & 1/6 & 1/3 & 1/3 & 1/6 \end{array} \tag{3.8}$$

Based on this tableau and by using equations 3.6 and 3.7 we get the following formula for the classic RK4:

$$y_{n+1} = y_n + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) \tag{3.9}$$

$$k_1 = hf(t_n, y_n) \tag{3.10}$$

$$k_2 = hf(t_n + \frac{h}{2}, y_n + \frac{h}{2}k_1) \tag{3.11}$$

$$k_3 = hf(t_n + \frac{h}{2}, y_n + \frac{h}{2}k_2) \tag{3.12}$$

$$k_4 = hf(t_n + h, y_n + hk_3) \tag{3.13}$$

From this we can see that the Forward Euler method can also be represented with the following Butcher tableau:

$$\begin{array}{c|c} 0 & 0 \\ \hline & 1 \end{array} \tag{3.14}$$

The above examples, RK4 and Euler, are considered explicit methods as the new solution $y_{n+1}$ is only dependent on the previous solutions $y_n$. However, cardiac cell models are often considered stiff and therefore the limited stability of explicit methods may not be sufficient. Stiffness will be further discussed in the following section.

### 3.1.3 Stiff ODEs

Although the strict definition of stiffness is still very controversial in numerical analysis, an ODE is generally considered stiff when the numerical solution is unstable and very small step-sizes are usually necessary. This is largely caused by a solution which rapidly changes values, as in the case of cardiac cell models.

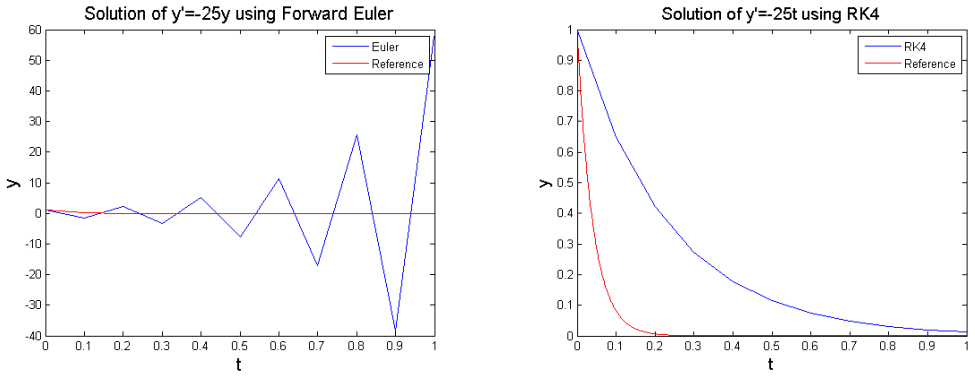



Figure 3.2: Illustration of the solution of a stiff ODE using forward Euler and RK4 with step-size=0.1 which depicts instability

From Figure 3.2 we can see that a simple and easy stiff ode, $y' = -25y$, solved with the Forward Euler method exhibits instability in the solution whereas the higher order RK4 method gives a relatively accurate result using the same step-size. Nevertheless, it is possible to further increase the stability of the solution by using a set of numerical solvers known as implicit methods. These techniques will be discussed in the following section.

### 3.1.4 Implicit Methods

If we look at the generalized butcher tableau of Equation 3.5 we can see that all the previous explicit methods have a value of zero for anything on and to the right of the diagonal of the $A$ matrix. If this was not the case, however, when substituting the values to the equations 3.6 and 3.7, we would obtain a series of equations which would depend on future values. These would therefore necessitate the usage of root solving techniques such as Newton's method. Although this is more computationally expensive compared to explicit methods, but it leads to very accurate results. Now to illustrate how a Butcher tableau can lead to implicit methods, we give the following example:

$$\begin{array}{c|c} 1 & 1 \\ \hline & 1 \end{array} \tag{3.15}$$

This corresponds to the following equation:

$$y_{n+1} = y_n + hf(t_n + h, y_{n+1}) \tag{3.16}$$

which is known as the implicit or backward Euler method. As we can see $y_{n+1}$ cannot be isolated and therefore it is necessary to use Newton's method as the equation is non-linear.

Newton's method attempts to numerically find the solution to the equation $F(x) = 0$ when given an initial estimate. In the case of Implicit Euler, $F(x)$ will be $y_n - x + hf(t_n + h, x)$. Newton's method is an iterative solver who's algorithm is as follows:

$$x_{k+1} = x_k - \frac{F(x_k)}{F'(x_k)} \tag{3.17}$$

However, equation 3.17 is only valid when $F$ is a single equation. If $F$ corresponds to a system of equations, as is the case in cardiac cell models, it is

necessary to adapt the method and transform it as follows:

$$J(x_k)(x_{k+1} - x_k) = -F(x_k) \tag{3.18}$$

In equation 3.18, $F$ now corresponds to an array of functions containing the series of equations which must be solved. Furthermore, $J$ is the jacobian of the function $F$, which consists of a matrix containing all the partial derivatives of the function $F$ with respect to the vector $x_k$ and is as follows:

$$J(x_k) = \begin{bmatrix} \frac{\partial F_1}{\partial x_k^1} & \cdots & \frac{\partial F_1}{\partial x_k^m} \\ \vdots & \ddots & \vdots \\ \frac{\partial F_m}{\partial x_k^1} & \cdots & \frac{\partial F_m}{\partial x_k^m} \end{bmatrix} \tag{3.19}$$

Since no analytical function for the jacobian can be obtained it is necessary to approximate its value. This is done by calculating the slope of $F(x_k)$ at a location very near to $x_k$. Therefore, each element in the jacobian matrix can be approximated as follows:

$$\frac{\partial F_m}{\partial x_k^m} \approx \frac{F_m(x_k^m + tol) - F_m(x_k^m - tol)}{2tol} \tag{3.20}$$

where *tol* is a very small number such as $\sim 10^{-10}$. At every time-step of the implicit Euler solver it is necessary to do multiple iterations of Newton's method before obtaining the solution. This greatly increases the computational time of each step of the ODE solver. However, implicit methods generally have much higher stability properties.

Using the previous example of a stiff ODE, as seen in figure 3.2, we can now compare results using this first order implicit Euler method with those of RK4. These two solutions are shown in figure 3.3. As we can see from this figure, the implicit Euler method is much more stable and provides a more accurate solution than that of RK4 despite being only of first order accuracy. Therefore,

since larger step-sizes could potentially be used when solving unstable solutions, implicit methods could save on the solving time of stiff ODEs. In the case of cardiac cell models, the stiff nature of the ODEs make the usage of higher ordered implicit methods very attractive. This can be seen with the application of ESDIRK3 and SDIRK4 in literature [16] [19]. These higher ordered implicit methods will be introduced in the following sections.

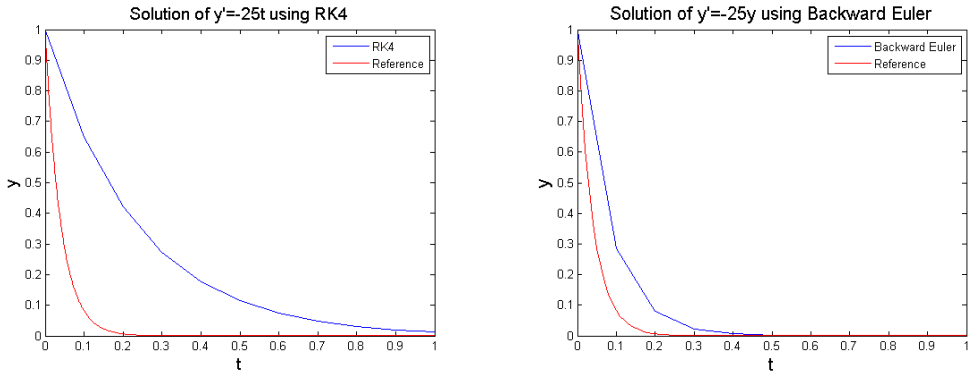



Figure 3.3: Illustration of the solution of a stiff ODE using RK4 and implicit Euler with step-size=0.1

### Higher Ordered Fully Implicit Methods

Fully Implicit methods are not limited to a first order of accuracy. Many higher ordered methods exist that are fully-implicit and are used when solving very stiff ODEs. For example the popular RADAU5 [18] method is of fifth order accuracy and very stable as it is fully implicit; it has the following Butcher tableau:

$$\begin{array}{c|ccc} \frac{4-\sqrt{6}}{10} & \frac{88-\sqrt{6}}{360} & \frac{296-169\sqrt{6}}{1800} & \frac{-2+3\sqrt{6}}{225} \\ \frac{4+\sqrt{6}}{10} & \frac{296+169\sqrt{6}}{1800} & \frac{88+7\sqrt{6}}{360} & \frac{-2-3\sqrt{6}}{225} \\ 1 & \frac{16-\sqrt{6}}{36} & \frac{16+\sqrt{6}}{36} & \frac{1}{9} \\ \hline y1 & \frac{16-\sqrt{6}}{36} & \frac{16+\sqrt{6}}{36} & \frac{1}{9} \end{array} \tag{3.21}$$

If we were to transform the first three rows of the Butcher tableau into it's

equations we would obtain:

$$k_1 = hf\left(t + h\frac{4-\sqrt{6}}{10}, y_n + h(k_1\frac{88-\sqrt{6}}{360} + k_2\frac{296-169\sqrt{6}}{1800} + k_3\frac{-2+3\sqrt{6}}{225})\right) \tag{3.22}$$

$$k_2 = hf\left(t + h\frac{4+\sqrt{6}}{10}, y_n + h(k_1\frac{296+169\sqrt{6}}{1800} + k_2\frac{88+7\sqrt{6}}{360} + k_3\frac{-2-3\sqrt{6}}{225})\right) \tag{3.23}$$

$$k_3 = hf\left(t + h, y_n + h(k_1\frac{16-\sqrt{6}}{36} + k_2\frac{16+\sqrt{6}}{36} + k_3\frac{1}{9})\right) \tag{3.24}$$

As we can see from equations 3.22-3.24, all three equations for the $k$s will be dependent on all three values of the $k$s. Therefore, we cannot use Newton's method directly and will instead have to solve the following system of equations:

$$(M)(k_l^{j+1} - k_l^j) = F_l(k_l^j) \tag{3.25}$$

$$F_l = -k_l^j + f(t_n + c_l h, y_n + h\sum_{i=1}^{n} a_{li}k_i^j) \tag{3.26}$$

where $k_l^{j+1}$ is the new solution to Newton's method, $a$, $b$ and $c$ are the elements of the Butcher tableau and M is as follows:

$$M = \begin{bmatrix} I - h \times a_{11} \times J(x_j) & -h \times a_{12} \times J(x_j) & -h \times a_{13} \times J(x_j) \\ -h \times a_{21} \times J(x_j) & I - h \times a_{22} \times J(x_j) & -h \times a_{23} \times J(x_j) \\ -h \times a_{31} \times J(x_j) & -h \times a_{32} \times J(x_j) & I - h \times a_{33} \times J(x_j) \end{bmatrix} \tag{3.27}$$

Here $J(x_j)$ is the jacobian of the function and I is the identity matrix. As we can see, this matrix can potentially get very large. For example, if we were to solve the Courtemanche cell model that contains 21 ODEs with the RADAU5 method, the M matrix would be $63 \times 63$. This means that a $63 \times 63$ linear system would need to be solved at every Newton's iteration at every time step in our ODE solver. Despite the high stability of this method, the added computational complexity of the large linear systems negates the potential

computational advantage of the larger step-sizes used [19]. Therefore, it would be advantageous to find a numerical method that has the stability properties of implicit methods without the need to solve large systems of equations.

### 3.1.5 Singly Diagonally Implicit Runge-Kutta Methods

A different class of implicit solvers is the singly diagonally implicit Runge-Kutta (SDIRK) methods. Although they are implicit methods and therefore require the usage of a root solving method, the size of the linear systems that must be solved is significantly reduced. All SDIRK methods have only null values to the right of the diagonal in the $A$ matrix of the Butcher tableau (eq 3.5). In this thesis, we will be presenting results obtained by the fourth order SDIRK4 [19] and third order ESDIRK3 [16] methods both of which have previously been used in solving cardiac cell models. The E in ESDIRK3 stands for explicit as the first stage is explicit. Both Butcher tableaus are presented below:

$$
\begin{array}{c|cccc}
0 & 0 & & & \\
c_2 & a_{21} & \gamma & & \\
c_3 & \hat{b_1} & \hat{b_2} & \gamma & \\
c_4 & b_1 & b_2 & b_3 & \gamma \\
\hline
y_1 & b_1 & b_2 & b_3 & \gamma
\end{array}
\qquad
\begin{array}{rc}
c_2= & 0.87173304301691799884 \\
c_3= & 1 \\
c_4= & 1 \\
\gamma= & 0.43586652150845899942 \\
a_{21}= & 0.43586652150845899942 \\
\hat{b_1}= & 0.49056338842178057060 \\
\hat{b_2}= & 0.073570090069760429950 \\
b_1= & 0.30880996997674652335 \\
b_2= & 1.4905633884217805707 \\
b_3= & -1.2352398799069860932
\end{array}
\tag{3.28}
$$

$$
\begin{array}{c|ccccc}
\frac{1}{4} & \frac{1}{4} & & & & \\
\frac{3}{4} & \frac{1}{2} & \frac{1}{4} & & & \\
\frac{11}{20} & \frac{17}{50} & \frac{-1}{25} & \frac{1}{4} & & \\
\frac{1}{2} & \frac{371}{1360} & \frac{-137}{2720} & \frac{15}{544} & \frac{1}{4} & \\
1 & \frac{25}{24} & \frac{-49}{48} & \frac{125}{16} & \frac{-85}{12} & \frac{1}{4} \\
\hline
y_1 & \frac{25}{24} & \frac{-49}{48} & \frac{125}{16} & \frac{-85}{12} & \frac{1}{4}
\end{array}
\tag{3.29}
$$

Now if we were to transform the Butcher tableau of ESDIRK3 into its equations we obtain:

$$k_1 = hf(t, y_n) \tag{3.30}$$

$$k_2 = hf(t + hc_2, y_n + h(a_{21}k_1 + \gamma k_2) \tag{3.31}$$

$$k_3 = hf(t + hc_3, y_n + h(\hat{b_1}k_1 + \hat{b_2}k_2 + \gamma k_3)) \tag{3.32}$$

$$k_4 = hf(t + hc_4, y_n + h(b_1k_1 + b_2k_2 + b_3k_3 + \gamma k_4)) \tag{3.33}$$

and for SDIRK4 we obtain:

$$k_1 = hf(t + h\frac{1}{4}, y_n + \frac{h}{4}k_1) \tag{3.34}$$

$$k_2 = hf(t + h\frac{3}{4}, y_n + h(\frac{1}{2}k_1 + \frac{1}{4}k_2)) \tag{3.35}$$

$$k_3 = hf(t + h\frac{11}{20}, y_n + h(\frac{17}{50}k_1 + \frac{-1}{25}k_2 + \frac{1}{4}k_3)) \tag{3.36}$$

$$k_4 = hf(t + h\frac{1}{2}, y_n + h(\frac{371}{1360}k_1 + \frac{-137}{2720}k_2 + \frac{15}{544}k_3 + \frac{1}{4}k_4)) \tag{3.37}$$

$$k_5 = hf(t + h, y_n + h(\frac{25}{24}k_1 + \frac{-49}{48}k_2 + \frac{125}{16}k_3 + \frac{-85}{12}k_4 + \frac{1}{4}k_5)) \tag{3.38}$$

From equations 3.30-3.38 we can see that as long as the $A$ matrix is empty to the right of the diagonal it is possible to use Newton's method directly to solve for the various $k$ variables individually. This property greatly reduces the size of the linear system that must be solved when compared to the fully-implicit methods. Nevertheless, all the $k$'s can be solved simultaneously using LU factorisation. As Newton's method is the most time-consuming step in the solver, some efforts have been made to decrease the computational time while keeping its efficiency. These modified root-finding algorithms will be explored in the following section.

### Simplified Newton's Method

One of the costliest steps in implicit methods is the usage of Newton's method. The constant recalculation of the jacobian and the potentially slow convergence can be too computationally expensive and cause the loss of any advantage gained by the increase in stability. However, it is possible to modify the classic Newton's equation 3.18 in order to increase both its rate of convergence and eliminate the necessity to recalculate the jacobian at each iteration. The formulation of these simplified Newton's methods are dependent on the numerical ODE solver used. For more information on modified Newton's method for ODE solvers, please refer to [18]. For ESDIRK3 it was found that the following Newton's formula was the most efficient [16]:

$$(I - h\gamma J)(k_i^{m+1} - k_i^m) = k^m + f(t + hc_i, y_n + h\sum_{j=1}^{i-1} a_{ij}k_j + \gamma k_i^m) \tag{3.39}$$

where the above values correspond to the Butcher tableau equations found in Equation 3.28. For SDIRK4 we use the following formula [18]:

$$(I - \frac{h}{4}J)(k_i^{m+1} - k_i^m) = k^m + f(t + hc_i, y_n + h\sum_{j=1}^{i-1} a_{ij}k_j + \frac{1}{2}k_i^m) \tag{3.40}$$

Furthermore, in order to avoid recalculating the jacobian at every iteration of Newton's method, we suppose that the value will remain relatively constant and approximate it as:

$$J \approx \frac{\partial f}{\partial y}(t_n, y_n) \tag{3.41}$$

for both ESDIRK3 and SDIRK4. Finally the jacobian of equation 3.41 will be decomposed using LU factorization and used for the implicit equations needed for all the $k$s. By using these simplified Newton's methods we have decreased the computational complexity of all the implicit solvers and have further increased their appeal in the application of cardiac cell models.

Despite the advantages of the SDIRK methods, some attempt to increase the stability of the solver while avoiding root-solving methods. These approaches are known as the Rush-Larsen methods which are specifically designed for usage in cardiac cell models and will be explored in the following two sections.

### 3.1.6 Rush-Larsen Method

In 1978, Stanley Rush and Hugh Larsen applied a first order solver to membrane models which take advantage of the structure of the ODEs found in cardiac cell models [21]. The Rush-Larsen (RL) method is still very widely used in cardiac cell modelling and actively researched [29]. The basis of their approach comes when considering the gating variables which have ODEs of the following form:

$$\frac{dy}{dt} = \frac{y_{\text{inf}} - y}{\tau_y} \tag{3.42}$$

where $y_{\text{inf}}$ and $\tau_y$ are dependent only on the transmembrane potential $V$. If we assume that in the interval between $t_{n+1}$ and $t_n$ the potential is constant, then equation 3.42 can be solved analytically in this interval as follows:

$$y_{n+1} = y_{\text{inf}} + (y_n - y_{\text{inf}}) \exp\left(\frac{-h}{\tau_y}\right) \tag{3.43}$$

For all other ODEs in the model that are not in the form of equation 3.42 we apply the forward Euler method as described in section 3.1.1. This method has been shown to be faster than the classical forward Euler method as well as allowing the usage of larger step-sizes [29].

### 3.1.7 Second Order Rush-Larsen Method

An extension of the Rush-Larsen method was presented by Sundnes et al. in 2009 which had an increased order of accuracy of 2 (RL2) [14]. This method

employs the exact same technique for the gating variables as seen in equation 3.43. However, unlike the classic method this equation is performed from $t_n$ to $t_{n+\frac{1}{2}}$ and then again from $t_{n+\frac{1}{2}}$ to $t_{n+1}$. Furthermore, the non-gating ODEs are not simply solved with the Euler method but a Taylor expansion is used. Thus, for the ODEs of the form:

$$\frac{dy_i}{dt} = f(y_i) \qquad i = 1, \cdots, k \tag{3.44}$$

We perform a Taylor expansion around a value $\eta$ and obtain:

$$\frac{dy_i}{dt} = f_i(\eta) + (y_i - \eta_i)\frac{\partial}{\partial y_i} f_i(\eta) \tag{3.45}$$

Which can then be solved analytically as:

$$y_i(t) = \eta_i + \frac{a}{b}(\exp(bh) - 1) \tag{3.46}$$

where $a = f_i(\eta)$ and $b = \partial f_i(\eta)/\partial y_i$. Furthermore, only the diagonal values of the jacobian must be calculated to obtain the values for $b$:

$$\frac{\partial f_i(\eta)}{\partial y_i} \approx \frac{f_i(\eta_1, \cdots, \eta_i + \delta, \cdots, \eta_k) - f_i(\eta)}{\delta} \tag{3.47}$$

where $\delta$ is very small such as $\sim 10^{-10}$. The algorithm for solving the ODEs of the form 3.44 is done in two steps:

1. Using $\eta = y_n$, we apply equation 3.46 from $t_n$ to $t_{n+1/2}$ and denote the solution as $y_{n+1/2}$

2. Now we can substitute the values of $\eta$ as follows

$$\eta = (y^1_{n+1/2}, \cdots, y^i_{n+1/2}, \cdots, y^k_{n+1/2})$$

   and apply it to equation 3.46 from $t_{n+1/2}$ to $t_n$ in order to obtain our final

solution of $y_{n+1}$.

This method retains the stability of the Rush-Larsen method while increasing the order of accuracy by one.

In this section, we presented various different methods that have been used in an attempt to increase the efficiency of solving the ODEs of the cardiac cell models. Since the solving of these ODEs are often the most time consuming part of cardiac tissue modeling, the search for increasingly accurate and efficient solvers is still an active area of research. Our contribution in this area will be presented in chapter 4 where a novel fourth order fully-implicit method will be introduced and studied in the context of cardiac cell modeling.

First, however, it is necessary to explore a technique which will enable us to couple the above ODE solvers with the PDEs of the tissue models. This will be presented in the following section.

## 3.2 Splitting Method for Solving Coupled Equations

When modeling the electrical activity of the heart, the information given by the cardiac cell models are necessary but not sufficient in providing a full description of the propagation of electric waves across the heart tissue. For this, it is necessary to incorporate the tissue models such as the bidomain and monodomain models. The Bidomain model contains two PDEs as follows:

$$\nabla \cdot (M_i \nabla v) + \nabla \cdot (M_i \nabla u_e) = \chi C_m \frac{\partial v}{\partial t} + \chi I_{ion} \tag{3.48}$$

$$\nabla \cdot (M_i \nabla v) + \nabla \cdot ((M_i + M_e) \nabla u_e) = 0 \tag{3.49}$$

Equation 3.48 has an parabolic form while equation 3.49 has an elliptic form. In contrast, the Monodomain model only has one PDE of the following form:

$$\frac{\lambda}{1+\lambda}\nabla \cdot (M_i \nabla v) = \chi C_m \frac{\partial v}{\partial t} + \chi I_{ion} \tag{3.50}$$

which is also of parabolic form. However, both the Monodomain and Bidomain models include a term for the ionic currents flowing through the cell membrane here written as $I_{ion}$. The value for $I_{ion}$, however, is given by the cell models and therefore consists of a series of ODEs that must be solved. For simplicity, it would be advantageous to be able to independently solve the ODEs and PDEs despite the fact that they are coupled. A first order splitting technique will be used in order to solve the tissue models as is described in detail in "Computing the Electrical Activity in the Heart" [15].

The first step in this splitting technique is to divide the simulation time into many much smaller time intervals which will be denoted as $\Delta t$. Therefore, each value of the potentials at a time $t_n$ is denoted as $v(t_n) = v_n$, $u_e(t_n) = u_e^n$ and $t_{n+1} = t_n + \Delta t$. The steps for solving the coupled model will be as follows:

1. Solve the cell model from $t = t_n$ to $t = t_n + \Delta t$ with initial conditions $v_n$ and $w_n^k$:

$$\chi C_m \frac{\partial v}{\partial t} = -I_{ion}(v, w^k) \tag{3.51}$$

$$\frac{\partial w^k}{\partial t} = f(v, w^k) \qquad k = 0 \cdots m \tag{3.52}$$

   Here $w^k$ represents all the other variables which are described by ODEs in the cardiac cell models of section 2.1. Therefore, for example, in the case of the Fox model this step would involve solving a system of 13 ODEs. In this step, the ODEs must be solved using any of the numerical techniques previously described in section 3.1. The solution of these equations will be saved as $v_{ODE} = v_{n+1}$ for the transmembrane potential and $w_{n+1}^k$ for all other ODEs.

2. Solve the system of PDEs from $t = t_n$ to $t = t_n + \Delta t$ with initial conditions $v_n = v_{ODE}$ and $u_e^n$

$$\nabla \cdot (M_i \nabla v) + \nabla \cdot (M_i \nabla u_e) = \chi C_m \frac{\partial v}{\partial t} \tag{3.53}$$

$$\nabla \cdot (M_i \nabla v) + \nabla \cdot ((M_i + M_e) \nabla u_e) = 0 \tag{3.54}$$

The solutions for this stage will be denoted as $v_{n+1}$ and $u_e^{n+1}$. Once this stage is completed the time is advanced, $t_{n+1} = t_n + \Delta t$, and we return to step 1. This is continued until the end of the time interval that is to be studied. If we wish to use the Monodomain equation instead of Bidomain, most of the steps are the same. The difference arises in step 2 where equation 3.54 is not used and equation 3.53 is replaced with equation 3.50 but without the $-I_{ion}$ term.

Although similar, both the elliptic and parabolic PDEs must be numerically solved using different techniques. Many methods exist for solving PDEs, however, we will be using an implicit finite difference method for the calculations. This will be discussed in detail in the following section.

## 3.3 Solving PDEs with Finite Difference

The basic premise of the finite difference method comes from approximating the derivatives in a similar way as was described in section 3.1. We first approximate the partial derivative by approximating the slope. This can be done using forward difference:

$$\frac{\partial f}{\partial t} \approx \frac{f(t+h) - f(t)}{h} \tag{3.55}$$

where $h$ is a small step-size used. This method is of first order accuracy. Another method of first order accuracy is that of backward difference:

$$\frac{\partial f}{\partial t} \approx \frac{f(t) - f(t-h)}{h} \tag{3.56}$$

Nevertheless a second order accurate method can also be constructed as with the central difference method:

$$\frac{\partial f}{\partial t} \approx \frac{f(t + (1/2)h) - f(t - (1/2)h)}{h} \tag{3.57}$$

However, in the tissue models some of partial derivatives are of second order. In order to solve these higher order derivatives we can perform the central difference a second time:

$$\frac{\partial^2 f}{\partial t^2} \approx \frac{f'(t + (1/2)h) - f'(t - (1/2)h)}{h} \tag{3.58}$$

$$\frac{f(t+h) - 2f(t) + f(t-h)}{h^2} \tag{3.59}$$

Using equations 3.55, 3.56 and 3.59 it is possible to construct appropriate solvers for both the elliptic and parabolic PDEs. These will be discussed in the following sections. Although the same technique can be used for higher dimensions, all derivations will be made assuming we will be simulating a one dimension fiber.

### 3.3.1 Solving Parabolic PDEs

In the case of the parabolic PDE we have the following equation in 1-D:

$$M_i \frac{\partial^2 v}{\partial x^2} + M_i \frac{\partial^2 u_e}{\partial x^2} = \chi C_m \frac{\partial v}{\partial t} \tag{3.60}$$

Now we can start replacing the partial derivatives with the finite difference equations. First begin by replacing $\frac{\partial v}{\partial t}$ with the forward difference equation

3.55 and then replace both second order partial differential equations with the second order central difference equation 3.59:

$$M_i \frac{v(x+\Delta x) - 2v(x) + v(x-\Delta x)}{\Delta x^2} + M_i \frac{u_e(x+\Delta x) - 2u_e(x) + u_e(x-\Delta x)}{\Delta x^2} = \chi C_m \frac{v(t+h) - v(t)}{h} \tag{3.61}$$

Where $\Delta x$ is the spatial step-size and $h$ is the temporal step-size. For simplicity, we will assume for the remainder of the derivation that $j$ represents the position and $n$ represents the time step; i.e $v_j = v(x)$, $v_{j+1} = v(x + \Delta_x)$, $v_n = v(t)$ and $v_{n+1} = v_n$. Therefore, equation 3.61 is identical to the following:

$$M_i \frac{v_{j+1}^n - 2v_j^n + v_{j-1}^n}{\Delta x^2} + M_i \frac{ue_{j+1}^n - 2ue_j^n + ue_{j-1}^n}{\Delta x^2} = \chi C_m \frac{v_j^{n+1} - v_j^n}{h} \tag{3.62}$$

Although this algorithm can be applied directly to solve for $v_j^{n+1}$, we must remember that stability is often the limiting factor for cardiac tissue. Therefore, by instead applying backward difference (equation 3.56) to $\frac{dv}{dt}$ and substituting $n-1$ with $n$ for the $v$ variables, we can obtain the following numerically stable solver:

$$M_i \frac{v_{j+1}^{n+1} - 2v_j^{n+1} + v_{j-1}^{n+1}}{\Delta x^2} + M_i \frac{ue_{j+1}^n - 2ue_j^n + ue_{j-1}^n}{\Delta x^2} = \chi C_m \frac{v_j^{n+1} - v_j^n}{h} \tag{3.63}$$

By rearranging this equation we obtain the following:

$$(1+2r)v_j^{n+1} - rv_{j+1}^{n+1} - rv_{j-1}^{n+1} = (r)ue_{j+1}^n - (2r)ue_j^n + (r)ue_{j-1}^n + v_j^n \tag{3.64}$$

where $r = \frac{M_i h}{\chi C_m \Delta x^2}$. From here we can see that equation 3.64 is a linear system

that can be solved with the usage of any solver of systems of linear equations. For our purpose we used LU factorization to solve this system. This solver has a second order of accuracy with respect to space and a first order accuracy with respect to time. If we wish to use the monodomain model instead of bidomain, equation 3.64 is simply modified as follows:

$$(1+2r)v_j^{n+1} - rv_{j+1}^{n+1} - rv_{j-1}^{n+1} = v_j^n \tag{3.65}$$

where $r = \frac{M_i h}{\chi C_m \Delta x^2}$. A similar approach can be taken in order to solve the elliptic equation. This will be explored in detail in the next section.

### 3.3.2 Solving Elliptic PDEs

In the case of the elliptic PDE, we have the following equation in 1-D:

$$(M_i + M_e)\frac{\partial u_e}{\partial x^2} = -M_i \frac{\partial^2 v}{\partial x^2} \tag{3.66}$$

Now using central difference (eq. 3.59) on both PDEs and assuming we'll be solving for a future value, we obtain:

$$\frac{ue_{j+1}^{n+1} - 2ue_j^{n+1} + ue_{j-1}^{n+1}}{\Delta x^2} = \frac{-M_i}{M_i + M_e}\frac{v_{j+1}^{n+1} - 2v_j^{n+1} + v_{j-1}^{n+1}}{\Delta x^2} \tag{3.67}$$

Once again this is a system of linear equations which can be solved using any linear system solver. As with the parabolic PDEs we decided to use LU factorization.

### 3.3.3 Boundary Conditions

From the previous sections, it can be noticed that all of our numerical solutions require values at the boundaries in order to obtain a solution. For the Bidomain

model we have 3 sets of boundary conditions which are as follows:

$$n \cdot (M_i \nabla v) = 0 \tag{3.68}$$

$$n \cdot (M_e \nabla u_e) = 0 \tag{3.69}$$

$$u_e(0) = 0 \tag{3.70}$$

Equations 3.68 and 3.69 are considered Neumann boundary conditions whereas 3.70 is a Dirichlet boundary condition. For the Dirichlet boundary we may simply substitute the value directly into our solutions. However, the Neumann boundaries require a bit more effort. Since we wish to preserve the second order of accuracy it is necessary to use a second order method for approximating the boundary condition. This method, similar to our derivations of the finite difference methods for PDEs, can be found in "Mathematically Modelling the Electrical Activity of the Heart"[28] and is presented as follows in 1-D:

$$\frac{-3v_0^{n+1} + 4v_1^{n+1} - v_2^{n+1}}{2\Delta x} = 0 \tag{3.71}$$

$$\frac{v_{l-2}^{n+1} - 4v_{l-1}^{n+1} + 3v_l^{n+1}}{2\Delta x} = 0 \tag{3.72}$$

$$\frac{-3ue_0^{n+1} + 4ue_1^{n+1} - ue_2^{n+1}}{2\Delta x} = 0 \tag{3.73}$$

$$\frac{ue_{l-2}^{n+1} - 4ue_{l-1}^{n+1} + 3ue_l^{n+1}}{2\Delta x} = 0 \tag{3.74}$$

Here $l$ represents the size of the fiber, i.e $l = x_{max}/\Delta x$, and $n$ represents the current time step of the solution. In the case of the Monodomain model, we have to use only equations 3.71 and 3.72. As we can see, the equations are implicit in order to preserve the stability of our implicit PDE solvers. Therefore, the boundaries cannot be solved separately from the main solution; they must be included directly in the matrix of the PDE solver at each time step and solved concurrently with the values at all other spatial locations. With these

boundary conditions it is now possible to use the splitting method to completely solve the Bidomain model numerically. The final algorithm will be given in the next section.

## 3.4 Combined Algorithm for Solving the Bidomain Model

By combining all the information provided in the previous sections it is possible to construct a complete algorithm for solving the Bidomain model. Here we'll assume that we're solving a one-dimensional fiber with Neumann boundary conditions using a first order splitting method with the Aliev-Panfilov cell model. This can be solved using the following series of instructions:

1. Solve the cell model from $t = t_n$ to $t = t_n + \Delta t$ with initial conditions $v_n$ and $w_n$ using any technique described in section 3.1. The ODE solver will have to be applied at every location $j$:

$$\frac{\partial v_j}{\partial t} = kv_j(v_j - a)(1 - v_j) - (v_j w_j) \tag{3.75}$$

$$\frac{dw_j}{dt} = (eps + \frac{u_1 w_j}{u_2 + v_j})(-w_j - (kv_j(v_j - a - 1))) \tag{3.76}$$

   The values will be saved as $v_{ODE} = v_{n+1}$ and $w_{n+1}$ and will contain elements for every location from $j = 0$ to $j = l$.

2. Solve the system of PDEs from $t = t_n$ to $t = t_n + \Delta t$:

   (a) Solve the parabolic equation by applying LU decomposition to the

following system with initial conditions $v_n = v_{ODE}$ and $u_e^n$:

$$\begin{bmatrix} -3 & 4 & -1 & 0 & \cdots & 0 \\ -r & (1+2r) & -r & 0 & \cdots & \vdots \\ 0 & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & -r & (1+2r) & -r \\ 0 & \cdots & 0 & 1 & -4 & 3 \end{bmatrix} \begin{bmatrix} v_0^{n+1} \\ \vdots \\ \vdots \\ \vdots \\ v_l^{n+1} \end{bmatrix} = \begin{bmatrix} 0 \\ (r)ue_2^n - (2r)ue_1^n + (r)ue_0^n + v_1^n \\ \vdots \\ (r)ue_l^n - (2r)ue_{l-1}^n + (r)ue_{l-2}^n + v_{l-1}^n \\ 0 \end{bmatrix} \tag{3.77}$$

Where $r = \frac{M_i \Delta t}{\chi C_m \Delta x^2}$. The results of this step gives the values for $v_{n+1}$.

(b) Solve the elliptic equation by applying LU decomposition to the following system with initial conditions $v_{n+1}$ and $u_e^n$:

$$\begin{bmatrix} -3 & 4 & -1 & 0 & \cdots & 0 \\ r_1 & -2r_1 & r_1 & 0 & \cdots & \vdots \\ 0 & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & r_1 & -2r_1 & r_1 \\ 0 & \cdots & 0 & 1 & -4 & 3 \end{bmatrix} \begin{bmatrix} ue_0^{n+1} \\ \vdots \\ \vdots \\ \vdots \\ ue_l^{n+1} \end{bmatrix} = \begin{bmatrix} 0 \\ (r_2)v_2^{n+1} - (2r_2)v_1^{n+1} + (r_2)v_0^{n+1} \\ \vdots \\ (r_2)v_l^{n+1} - (2r_2)v_{l-1}^{n+1} + (r_2)v_{l-2}^{n+1} \\ 0 \end{bmatrix} \tag{3.78}$$

Where $r_1 = \frac{1}{\Delta x^2}$ and $r_2 = \frac{M_i}{M_i + M_e} \frac{1}{\Delta x^2}$. This step gives us the results for $u_e^{n+1}$.

Once these steps are completed, the values for $u_e^{n+1}$, $v^{n+1}$ and $w^{n+1}$ are saved then the solution is advanced to $t_{n+1} = t_n + \Delta t$ and we return to step 1. This is repeated until we reach the end of the time interval that is to be studied.

As we can see from this algorithm, the ODE solver will be applied extensively as it is required for every temporal and spatial position. Therefore, any increase in efficiency in the ODE solver can significantly decrease the computational time required to simulate cardiac tissue. In the following chapter, we will be introducing an implicit Runge-Kutta method that has yet to be applied

in the context of electrophysiology. The unique properties of this solver seems to make it ideally suited for cardiac cell models and, therefore, its performance will be compared with that of multiple other popular methods.

# Chapter 4

# Nested Implicit Runge-Kutta Method of Order 4

When modelling the electrical activity in the heart, solving the stiff ODEs in the cardiac cell models is often the most time consuming portion of the simulation by a significant factor [16]. Therefore, there is much incentive to improve the efficiency of solving these systems of ODEs in order to greatly decrease the extensive computational time of tissue simulations. For this reason, the application of new ODE solvers to cardiac cell models is still an important active area of research.

In 2001, Sundnes et al used a $3^{rd}$ order implicit Runge-Kutta method in solving the ODEs of a stiff human cardiac model when coupling with the bidomain equation [16]. It was found that this method saved significant computational time when compared to a similarly ordered explicit solver. In contrast, Spiteri and Dean used an explicit-implicit Runge-Kutta solver which employed a splitting method in an attempt to only use the implicit solver when stability was necessary while defaulting to the more rapid explicit solver [19]. This method's performance was compared with a collection of other popular implicit and explicit solvers. However, Runge-Kutta methods are not the only family of solvers that have been explored for this application. In 2009, Sundnes et al. proposed

a second order Rush-Larsen method which was found to be much more efficient than the classical first order Rush-Larsen [14]. However, in 2012 Marsh et al. compared the performance of the classical Rush-Larsen method with that of their own generalized Rush-Larsen method and found that RL outperformed their own method for all but the most stiff cell models [29]. Nevertheless, a search for increasingly efficient solvers of cardiac cell models is ever-present.

In this chapter, we will be presenting a Nested Implicit Runge-Kutta method of order four (NIRK4) and comparing its performance to that of previously studied solvers of cardiac cell models. This method, first introduced in 2006 by Gennady Kulikov and Sergey Shindin, is fully implicit and A-stable [20]. As previously discussed in chapter 3, fully-implicit methods are rarely used due to the computational complexity caused by the large linear systems that must be solved. However, one of the unique characteristics of NIRK4 is its explicit internal stages which reduces the solver to one non-linear equation. These characteristics will be explored in more detail in the following section. Furthermore, its performance will be compared with multiple ODE solvers for various cell models of increasing complexity.

## 4.1 Numerical Method

First, let us examine the Butcher tableau of NIRK4:

$$\begin{array}{c|cccc} 0 & 0 & 0 & 0 & 0 \\ c_1^2 & \frac{6(c_1^2+\theta)-5}{12} & \frac{1-\theta}{2} & \frac{1-\theta}{2} & \frac{6(c_1^2+\theta)-7}{12} \\ 1-c_1^2 & \frac{7-6(c_1^2+\theta)}{12} & \frac{\theta}{2} & \frac{\theta}{2} & \frac{5-6(c_1^2+\theta)}{12} \\ 1 & 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ \hline & 0 & \frac{1}{2} & \frac{1}{2} & 0 \end{array} \tag{4.1}$$

where $c_1^2 = (3-\sqrt{3})/6$ and $\theta = 1/2 + 2\sqrt{3}/9$. Now the explicit internal stages of this method allows us to simplify all the equations corresponding to the

Butcher tableau into the following:

$$\mathbf{y}_{k+1} = \mathbf{y}_k + h(b_1 f(t_k + c_1 h, \mathbf{x}_1) + b_2 f(t_k + c_2 h, \mathbf{x}_2)) \tag{4.2}$$

$$\mathbf{x}_1 = a_{11}\mathbf{y}_k + a_{12}\mathbf{y}_{k+1} + h(d_{11} f(t_k, \mathbf{y}_k) + d_{12} f(t_{k+1}, \mathbf{y}_{k+1})) \tag{4.3}$$

$$\mathbf{x}_2 = a_{21}\mathbf{y}_k + a_{22}\mathbf{y}_{k+1} + h(d_{21} f(t_k, \mathbf{y}_k) + d_{22} f(t_{k+1}, \mathbf{y}_{k+1})) \tag{4.4}$$

$$\begin{array}{llllll} a_{11} = & \theta & a_{12} = & 1-\theta & a_{21} = & \theta \\ a_{22} = & \theta & b_1 = & \frac{1}{2} & b_2 = & \frac{1}{2} \\ c_1 = & \frac{3-\sqrt{3}}{6} & c_2 = & \frac{3+\sqrt{3}}{6} & d_{11} = & \frac{6\theta-2-\sqrt{3}}{12} \\ d_{12} = & \frac{6\theta-4-\sqrt{3}}{12} & d_{21} = & \frac{-6\theta+4+\sqrt{3}}{12} & d_{22} = & \frac{-6\theta+2+\sqrt{3}}{12} \end{array}$$

As you can see, equations 4.2, 4.3 and 4.4 can all be combined into one large formula which has only $\mathbf{y}_{k+1}$ as its unknown. Therefore, Newton's method can be applied directly to this one equation while keeping the high order of 4. In contrast, Newton's method must be applied three and four distinct times in ESDIRK3 and SDIRK4 respectively. This property makes NIRK4 especially attractive to the large systems of stiff ODES found in cardiac cell models.

Furthermore, as with SDIRK4 and ESDIRK3, it is possible to use a simplified Newton's method for NIRK4 to further increase its efficiency. We used the following simplified formula [20]:

$$(I - \frac{h}{4}J)^2(\mathbf{y}_k^{m+1} - \mathbf{y}_k^m) = \mathbf{y}_k + h(b_1 f(t_k + c_1 h, \mathbf{x}_1^m) + b_2 f(t_k + c_2 h, \mathbf{x}_2^m)) \tag{4.5}$$

where the above values can be found in Section 4.1. Furthermore, in order to avoid recalculating the jacobian at every iteration of Newton's method we suppose that the value will remain relatively constant and approximate it as:

$$J \approx \frac{\partial f}{\partial y}(t_{n+1} + h, y_{n+1}) \tag{4.6}$$

By using this simplified Newton's method, we have further decreased the computational complexity of NIRK4.

## 4.2 Results for Solving Cardiac Cell Models

### 4.2.1 Calculating the Norm

In order to properly compare the performance of multiple ODE solvers, it is imperative to calculate the error of the numerical solution when compared to the actual results. When an analytical solution is possible this is rendered quite easy as the result can be compared directly to its theoretical values. However, for cardiac cell models this is not possible as the ODEs cannot be solved analytically. Nevertheless, we must still be able to compare the accuracy of the numerical solvers. Therefore, in these cases it is necessary to create a reference solution which would have a negligible error when compared to the solutions calculated in our tests. In the case of the cell models, we solved the reference solution using a variable step solver than is included in the MATLAB software called ODE45. This solver uses a popular variable step-size 5th order Runge-Kutta method called the Dormand-Prince technique. For our reference solution we specified a very small error tolerance of $\sim 10^{-15}$ for the relative and absolute tolerances. Appropriate reference solutions take very long to calculate and require much memory to store.

After obtaining a reference solution, we would like to estimate the error of our solution in order to compare the accuracy of our many numerical solvers. Calculating the norm between these two solutions is well suited for this purpose. In theory, the p-norm of a solution is as follows:

$$||x||_p = (\sum_{i=0}^{N} |x_i|^p)^{1/p} \tag{4.7}$$

Furthermore, if we solve for when $p$ approaches infinity we obtain:

$$||x||_\infty = max(|x_i|) \qquad 0 \leq i \leq N \tag{4.8}$$

which is called the infinity norm. In our case, we define the error as:

$$e_i = V_{ref}^i - V^i \tag{4.9}$$

where $V_{ref}$ and $V$ are the values of the transmembrane potential for the reference and calculated solutions respectively. Therefore, we are generally trying to calculate:

$$||e||_\infty = max(|V_{ref}^i - V^i|) \qquad 0 \leq i \leq N \tag{4.10}$$

Another popular method when comparing numerical methods in biomedical modelling is the Relative Root-Mean Squared error (RRMS). In the context of cardiac cell models, a solution is considered of adequate accuracy when it has an RRMS of maximum 5% [19][14]. For our application, the RRMS is calculated as follows:

$$||e||_{RRMS} = \sqrt{\frac{\sum_{i=0}^{N}(V^i - V_{ref}^i)^2}{\sum_{i=0}^{N}(V_{ref}^i)^2}} \tag{4.11}$$

### 4.2.2 Comparing Implicit Methods with NIRK4

In this section, the performance and the robustness of NIRK4 will be investigated. We therefore consider the two complex cardiac cell models of Fox and Luo-Rudy. These models were solved using three implicit methods: ESDIRK3, SDIRK4 and NIRK4. This enabled us to compare the NIRK method with implicit solvers previously used in electrocardiology. For both these models, an elevated initial condition for the transmembrane potential was used in order to stimulate the membrane instead of using a stimulus current. This value was $V = -40mV$ for Fox and $V = -35mV$ for Luo-Rudy. Furthermore, the length of simulation was $300ms$ for Fox and $350ms$ for Luo-Rudy which is adequate

time for the cell to return to a rest state. Finally, the infinite norm was calculated using 100 points evenly spaced throughout the simulation as similarly used in Dean et al. [19]. The results of the infinite norms and the simulation times for varying step-sizes are presented in Tables 4.2 and 4.1.

| LuoRudy I: Step-Size $h = 0.1$ | | |
|---|---|---|
| Methods | $\|\|e\|\|_\infty$ | CPU(s) |
| ESDIRK3 | 0.23 | 0.69 |
| Sdirk4 | 4.49e-2 | 0.77 |
| **NIRK4** | **2.58e-2** | **0.61** |

| LuoRudy I: Step-Size $h = 0.05$ | | |
|---|---|---|
| Methods | $\|\|e\|\|_\infty$ | CPU(s) |
| ESDIRK3 | 4.15e-3 | 1.28 |
| Sdirk4 | 2.35e-3 | 1.52 |
| **NIRK4** | **4.98e-4** | **1.21** |

| LuoRudy I: Step-Size $h = 0.025$ | | |
|---|---|---|
| Methods | $\|\|e\|\|_\infty$ | CPU(s) |
| ESDIRK3 | 4.19e-4 | 2.52 |
| Sdirk4 | 1.42e-4 | 3.08 |
| **NIRK4** | **1.73e-5** | **2.34** |

Table 4.1: Luo-Rudy results for implicit methods

As can be seen in Table 4.1, for the Luo-Rudy model NIRK outperforms the more popular singly diagonally implicit methods. For every step-size tested, NIRK4 exhibits both a smaller error while decreasing the total computational time necessary when compared to ESDIRK3 and SDIRK4. Furthermore, we can see that the fourth order of accuracy is maintained for SDIRK4 and NIRK4 while the third order is maintained for ESDIRK3 despite using the more efficient simplified Newton's method. In order to confirm the superior performance of NIRK4 we will perform the same test for the more complex Fox model whose results are presented in table 4.2.

As presented in Table 4.2, NIRK4 again clearly outperforms the similarly

| Fox: Step-Size $h = 0.1$ | | |
|---|---|---|
| Methods | $\|\|e\|\|_\infty$ | CPU |
| **ESDIRK3** | **0.158** | **0.589** |
| Sdirk4 | 2.94e-2 | 0.647 |
| **NIRK4** | **2.2e-2** | **0.622** |

| Fox: Step-Size $h = 0.05$ | | |
|---|---|---|
| Methods | $\|\|e\|\|_\infty$ | CPU |
| **ESDIRK3** | **1.99e-3** | **1.168** |
| Sdirk4 | 1.25e-3 | 1.283 |
| **NIRK4** | **2.46e-4** | **1.248** |

| Fox: Step-Size $h = 0.025$ | | |
|---|---|---|
| Methods | $\|\|e\|\|_\infty$ | CPU(s) |
| **ESDIRK3** | **5.71e-4** | **2.332** |
| Sdirk4 | 1.10e-4 | 2.577 |
| **NIRK4** | **2.83e-5** | **2.464** |

Table 4.2: Fox results for implicit methods

ordered SDIRK4 in every step-size used as the error and computational time is significantly decreased when using NIRK4. However, when compared with ESDIRK3, we can see that the computational time is slightly lower for this lower ordered method than that of NIRK4. Nevertheless, this comes at the cost of an increase of an order of magnitude in error when compared with the more accurate NIRK4 method.

From these results, we believe that the properties of the NIRK4 method allows for a superior performance when compared to those of the more popular implicit methods ESDIRK3 and SDIRK4. Now, in the following section, we will compare NIRK4's performance to that of the first and second order Rush-Larsen methods. The Rush-Larsen methods, specifically designed to take advantage of the unique properties of cardiac cell models, are the most widely used methods of solving these models [30].

### 4.2.3 Comparing Rush-Larsen and NIRK4

In much of literature, the stated advantage of the Rush-Larsen method is that it is very efficient when wanting to obtain a result that is within 5% RRMS of a reference solution [14][29]. Therefore, in order to further explore the advantages of NIRK4 it is imperative to compare its performance to that of Rush-Larsen in obtaining the stated error value. In order to accomplish this, we performed tests on the Courtemanche, Luo-Rudy, Fox and Tusscher cell models using the Rush-Larsen of order 1(RL), the Rush-Larsen of order 2(RL2) and the NIRK4 methods. The performance of Rush-Larsen methods using cell models has been investigated in Sundnes et al. [14]. In this section, we use the same conditions presented in [14] to compare the results of RL and RL2 with those of the NIRK4 method. Therefore, all simulations were performed from 0 to 10 ms using an initial potential difference of $V = -40mV$ to stimulate the solution. The step-sizes of the numerical solvers were gradually adjusted until we obtained a value of 5%, 1% and 0.2% error RRMS. The results of the simulation times and step-sizes used for the numerical methods are presented in Tables 4.3, 4.4, 4.5 and 4.6. The Rush-Larsen results of Tables 4.3 and 4.4 are very similar to those found in the 2009 paper by Sundnes et al.[14].

In table 4.3, we can see that NIRK4 outperforms both the second and first order Rush-Larsen methods at every error tolerance level tested using the Luo-Rudy model. Due to its ability to use larger step-sizes, the computational time was significantly lowered when compared to RL and RL2 especially as we increased the desired accuracy.

In the case of the more complex Courtemanche model, table 4.4 shows that once again the NIRK4 results exhibit a decrease in computational time for all error tolerances tested when compared to RL and RL2.

For the Fox model, the results of NIRK4 and RL2 are identical at the 5%

| LuoRudy I: RRMS=5% | | |
|---|---|---|
| Methods | step-size | CPU (s) |
| RL | 0.014 | 0.056 |
| RL2 | 0.11 | 0.012 |
| **NIRK4** | 0.152 | **0.009** |

| LuoRudy I: RRMS=1% | | |
|---|---|---|
| Methods | step-size | CPU (s) |
| RL | 0.0028 | 0.242 |
| RL2 | 0.038 | 0.033 |
| **NIRK4** | 0.152 | **0.009** |

| LuoRudy I: RRMS=0.2% | | |
|---|---|---|
| Methods | step-size | CPU(s) |
| RL | 0.00055 | 1.206 |
| RL2 | 0.016 | 0.067 |
| **NIRK4** | 0.095 | **0.019** |

Table 4.3: Results of Rush-Larsen and NIRK4 methods for Luo-Rudy

| Courtemanche: RRMS=5% | | |
|---|---|---|
| Methods | step-size | CPU (s) |
| RL | 0.011 | 0.136 |
| RL2 | 0.1 | 0.031 |
| **NIRK4** | 0.21 | **0.025** |

| Courtemanche: RRMS=1% | | |
|---|---|---|
| Methods | step-size | CPU (s) |
| RL | 0.0022 | 0.62 |
| RL2 | 0.04 | 0.112 |
| **NIRK4** | 0.18 | **0.029** |

| Courtemanche: RRMS=0.2% | | |
|---|---|---|
| Methods | step-size | CPU(s) |
| RL | 0.00045 | 3.088 |
| RL2 | 0.017 | 0.184 |
| **NIRK4** | 0.08 | **0.069** |

Table 4.4: Results of Rush-Larsen and NIRK4 methods for Courtemanche

| Fox: RRMS=5% | | |
|---|---|---|
| Methods | step-size | CPU (s) |
| RL | 0.01 | 0.093 |
| **RL2** | 0.1 | **0.017** |
| **NIRK4** | 0.15 | **0.017** |

| Fox: RRMS=1% | | |
|---|---|---|
| Methods | step-size | CPU (s) |
| RL | 0.0025 | 0.356 |
| RL2 | 0.035 | 0.041 |
| **NIRK4** | 0.15 | **0.017** |

| Fox: RRMS=0.2% | | |
|---|---|---|
| Methods | step-size | CPU(s) |
| RL | 0.0005 | 1.728 |
| RL2 | 0.15 | 0.096 |
| **NIRK4** | 0.11 | **0.021** |

Table 4.5: Results of Rush-Larsen and NIRK4 methods for Fox

error tolerance as seen in table 4.5. However, once the tolerance is lowered, the NIRK4 method once again exhibits a significant advantage when comparing its computational time to that of RL and RL2.

In the case of the Tusscher model, Rush-Larsen of order 2 gives the quickest result for the largest error tolerance. However, once a more accurate solution is desired, NIRK4 gives a significantly more computationally efficient result.

As can be deduced from the results obtained, when simulating the Luo-Rudy, Courtemanche, Fox and Tusscher models, NIRK4 has clear advantages to the very popular Rush-Larsen methods. When compared to the far more common classic RL method, NIRK4 outperformed at every tolerance for every model tested. However, RL2 outperformed NIRK4 once in the Tusscher model and performs equally once in the Fox model at $5\%RRMS$. Nevertheless, at all other tolerances and models NIRK4 gives significantly faster results than RL2.

By considering the results presented in both this and the previous sections,

| Tusscher: RRMS=5% | | |
|---|---|---|
| Methods | step-size | CPU (s) |
| RL | 0.0175 | 0.076 |
| **RL2** | 0.12 | **0.022** |
| NIRK4 | 0.14 | 0.03 |

| Tusscher: RRMS=1% | | |
|---|---|---|
| Methods | step-size | CPU (s) |
| RL | 0.0035 | 0.365 |
| RL2 | 0.035 | 0.071 |
| **NIRK4** | 0.14 | **0.03** |

| Tusscher: RRMS=0.2% | | |
|---|---|---|
| Methods | step-size | CPU(s) |
| RL | 0.00075 | 1.65 |
| RL2 | 0.016 | 0.154 |
| **NIRK4** | 0.13 | **0.03** |

Table 4.6: Results of Rush-Larsen and NIRK4 methods for Tusscher

we deduce that NIRK4 has significant advantages to the other more popular methods that have previously been studied in literature. Due to the increase in accuracy compared to the ESDIRK3 and SDIRK4 methods, combined with a usual decrease in computational time, we deem NIRK4 to have an increased performance to the singly diagonally implicit methods when solving the Fox and Luo-Rudy cell models. For further information on the performance of NIRK4 in comparison to other implicit methods, please refer to our previous work detailed in [31]. Furthermore, with its decrease in solving time for same error tolerances, we regard the NIRK4 method as more efficient when solving the four cell models as compared to classic Rush-Larsen as well as being frequently superior to the Rush-Larsen method of order 2 especially at lower error tolerances. All of the numerical techniques necessary to obtain the results in this chapter were programmed with C++. The analysis was then performed using MATLAB.

In the following chapter, we will be concentrating on the behavior of the propagation of the electrical signal in cardiac tissue. Based on this chapter, the

NIRK4 solver will always be used whenever we include complex cardiac cells to the tissue models. Experiment has shown that changes in temperature have significant impact on the conduct of the transmembrane potential. Therefore, in order to study multiple phenomena that could be affected by the temperature, it would be advantageous to add temperature effects to cardiac tissue and cell models. An overview of this new model, along with simulations involving special consideration for the effect of temperature on the action potential duration will be the concentration of chapter 5.

# Chapter 5

# Including Temperature Dependence to Tissue and Cell Models

Electrophysiological models can be used to accurately describe the propagation of the transmembrane potential in both the tissue and cellular domains. This is done using the Bidomain model, as described in section 2.2.1, and coupling it with the wide array of available cell models as introduced in section 2.1. However, the models detailed so far do not take into consideration any possible effects caused by changes of temperature in the tissue.

Motivation for including temperature effects in these simulations arise from experimental evidence which currently suggests that changes in temperature can have significant influence on the action potential duration (APD) [26] [32]. APDs are of especial interest in cardiac modeling as, for instance, alteration in these durations can lead to the formation of electrical alternans. These alternans, which are characterized by periodic changes in APDs, are of preeminent interest to researchers as they've been linked to causing spiral waves which can lead to arrhythmias such as ventricular fibrillation, a phenomena which has proved difficult to study in vivo [24]. Furthermore, Yamazaki et al. have

shown that regional cooling prolonged the APD and promoted termination of ventricular tachycardias in rabbit hearts [33].

In 2006, Bini et al [23] proposed a modification of the FitzHugh-Nagumo model which includes the addition of temperature influence on the behavior of a simulated nerve cell. This was followed in 2009 by the inclusion of temperature in the Hodgkin-Huxley nerve cell model to further increase the accuracy of this behavior. Similarly, Gizzi et al. [47] added temperature dependence to ionic intestine models in order to assess it's possible effects during surgery. Finally, very recent work by Fenton et al. [26] and Filippi et al. [27] have developed a cell model that includes temperature influence and applied it to a monodomain model for studying spiral wave effects in canine heart tissue.

Temperature effects on the electrical wave have two sources. First off, propagation of the electric wave itself induces changes in temperature caused by the joule effect. This effect causes changes in temperature of $\sim \mu^o C$ magnitude in the myocardium and therefore has very little effect on APDs. However, it has been proposed that the behavior of this localized heat source could be used as a potential method for visualizing the tip of spiral waves using thermal detectors [23]. This would have significant importance in multiple applications. Secondly, from the original work by Hodgkin and Huxley, temperature is shown to have a significant effect on the rate of change of cell conduction variables [2]. Subsequent modifications by FitzHugh [35] and Moore [36] also show linear changes of ionic conductances with respect to temperature. These alterations in the cellular responses to changes in temperature can have profound effects on the behavior of the transmembrane potential including APDs.

In this chapter, we will introduce the two above mentioned temperature effects on the bidomain model, which is considered the most realistic mathematical model in electrocardiology. Therefore, we will begin by introducing Pennes' bioheat equation and its coupling with the bidomain tissue model. This will be followed by suggesting changes to the simple 2 ODE Aliev-Panfilov and

the more complex 8 ODE Luo-Rudy models which allow for the inclusion of temperature effects at the cellular level. Finally, we will provide multiple simulations in order to study the consequences of temperature changes on the action potential in a one and two dimensional tissue.

## 5.1 Mathematical Model

As was previously discussed in section 2.1, the Aliev-Panfilov model can reproduce more realistic shapes of the cardiac action potential when compared to other simplified models such as FitzHugh-Nagumo. Furthermore, it can reproduce the APD restitution characteristic observed in the experiments [17]. Although this model uses dimensionless units, the simulation results could be compared to experimental studies. For instance, according to experimental observations presented in Elharrar and Surawicz [37] each non-dimensional time unit corresponds to 12.9ms in the Aliev-Panfilov model, and the actual transmembrane potential is recovered by $V_m[mV] = -80 + 100U$. For more details the reader is refereed to Nash and Panfilov [38]. In this work, we have scaled the Aliev-Panfilov model to obtain physiologically interpretable values and the modified equation are given by:

$$I_{ion}(V_m, W) = -((\frac{k}{v_{amp}^2})(V_m - v_{rest})(V_m - v_a)(V_m - v_{peak})) - (V_m - v_{rest})W \quad (5.1)$$

$$\begin{aligned} G(V_m, W) = & \frac{1}{12.9}(\epsilon_0 + \frac{\mu_1 W}{(V_m - v_{rest})/v_{amp} + \mu_2}) \times \\ & (-W - (k(\frac{V_m - v_{rest}}{v_{amp}})((\frac{(V_m - v_{rest})}{v_{amp}}) - a - 1))) \end{aligned} \quad (5.2)$$

This can then be included in the bidomain tissue model which was first presented in section 2.2:

$$\begin{cases} \chi C_m \dfrac{\partial V_m}{\partial t} - \nabla \cdot (M_i \nabla V_m) = \nabla \cdot (M_i \nabla u_e) - \chi I_{ion}(V_m, W) \\ \nabla \cdot ((M_i + M_e)\nabla u_e) = -\nabla \cdot (M_i \nabla V_m) \\ \dfrac{\partial W}{\partial t} = G(V_m, W), \end{cases} \tag{5.3}$$

However, despite now having a dimensional model with more realistic cardiac behavior, we still lack the ability to study any temperature effects. In order to accomplish this, we must first consider Pennes' Bioheat equation [34]. This equation, first introduced in 1948, seeks to describe the transfer of heat in biological tissue and is as follows:

$$\rho c_p \frac{\partial T}{\partial t} = \nabla \cdot (k \nabla T) + b(T_a - T) + A_0 \tag{5.4}$$

Where $k$ is the thermal conductivity, $T$ is the temperature, $T_a$ is the arterial temperature, $b$ is heat sink strength due to blood perfusion, $A_0$ is the metabolic heat generation, $\rho$ is the tissue density and $c_p$ is the tissue heat capacity. However, this equation does not take into consideration the effect of the propagation of the electric wave on the temperature. This response can be included simply by considering the joule effect, the heat generation rate per unit volume from an electric field $p$ (see [39]and [23]):

$$p = j \cdot (-\nabla V_m) \tag{5.5}$$

where $j$ is the current density as follows:

$$j = -\sigma \nabla V_m \tag{5.6}$$

and where $\sigma$ is the electrical conductivity tensor. This equation is incorporated as an internal source term in equation (5.4) which leads to the coupling

between the heat transfer in the tissue and the transmembrane potential in the myocardium. Furthermore, the terms for the metabolic heat and blood perfusion can be combined[23] as the generated heat is rapidly redistributed throughout the tissue [40] [41]:

$$\rho c_p \frac{\partial T}{\partial t} = \nabla \cdot (k \nabla T) + b_c(T^* - T) + \sigma \nabla V_m \cdot \nabla V_m \tag{5.7}$$

where $b_c$ is the combined term for any metabolic or blood perfusion effects and $T^*$ is the temperature of the environment. We can see from the structure of the equation 5.7 that this PDE is parabolic with respect to $T$. Therefore, by using the same approach as detailed in section 3.3.1 and applying central difference to approximate $\nabla V_m$, we obtain the following numerical equation:

$$k\frac{T_{j+1}^{n+1} - 2T_j^{n+1} + T_{j-1}^{n+1}}{\Delta x^2} + b_c(T^* - T_j^{n+1}) + \sigma \left( \frac{Vm_{j+1}^n - Vm_{j-1}^n}{2\Delta x} \right)^2 = \rho c_p \frac{T_j^{n+1} - T_j^n}{\Delta t} \tag{5.8}$$

where $j$ and $n$ represent the position in the space and time domains respectively. Furthermore, $\Delta x$ and $\Delta t$ are the spatial and temporal step-sizes.

Nevertheless, the temperature also has a significant influence on the ionic behavior in the cells. This must therefore be included in the Aliev-Panfilov model in order to accurately represent the consequences of temperature changes on the action potential. First, modifications by FitzHugh [35] and Moore [36] include the linear changes of ionic conductances with respect to temperature as follows:

$$I_{ion}(V_m, W, T) = \eta(T) I_{ion}(V_m, W) \tag{5.9}$$

Where

$$\eta(T) = A(1 + B(T - T_0)) \tag{5.10}$$

where A and B are constants and $T_0$ is a reference temperature which, in the case of cardiac cells, will be $37^oC$. Furthermore, Hodgkin and Huxley studied

the effects of temperature on the rate of change of the conductance variables where it was deemed appropriate to use $Q_{10}$ factors to modify these variables:

$$G(V_m, W, T) = \phi(T)G(V_m, W) \tag{5.11}$$

where

$$\phi(T) = Q_{10}^{(T-T_0)/10} \tag{5.12}$$

where $Q_{10}$ is the said temperature coefficient. By coupling Pennes' equation, the bidomain tissue model and the dimensional Aliev-Panfilov cell model with included temperature coefficients, we obtain the following system of equations:

$$\begin{cases} \chi C_m \dfrac{\partial V_m}{\partial t} - \nabla \cdot (M_i \nabla V_m) = \nabla \cdot (M_i \nabla u_e) - \chi I_{ion}(V_m, W, T), \\ \nabla \cdot ((M_i + M_e)\nabla u_e) = -\nabla \cdot (M_i \nabla V_m), \\ \dfrac{\partial W}{\partial t} = G(V_m, W, T), \\ \rho c_p \dfrac{\partial T}{\partial t} = \nabla \cdot (k \nabla T) + b_c(T^* - T) + \sigma \nabla V_m \cdot \nabla V_m. \end{cases} \tag{5.13}$$

Therefore, in order to apply the splitting technique to solve equations 5.13, we can perform steps 1 and 2 as detailed in section 3.4. This is followed by one final step where the parabolic PDE of Pennes' bioheat equation is solved

by applying LU factorization to the following system:

$$\begin{bmatrix} -3 & 4 & -1 & 0 & \cdots & 0 \\ -r & (1+2r+\frac{b_c\Delta t}{\rho c_p}) & -r & 0 & \cdots & \vdots \\ 0 & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & -r & (1+2r+\frac{b_c\Delta t}{\rho c_p}) & -r \\ 0 & \cdots & 0 & 1 & -4 & 3 \end{bmatrix} \begin{bmatrix} T_0^{n+1} \\ \vdots \\ \vdots \\ \vdots \\ T_l^{n+1} \end{bmatrix} = \begin{bmatrix} 0 \\ T_j^n + \frac{b_c\Delta t T^*}{\rho c_p} + \frac{\sigma\Delta t}{\rho c_p}\left(\frac{Vm_{j+1}^n - Vm_{j-1}^n}{2\Delta x}\right)^2 \\ \vdots \\ T_j^n + \frac{b_c\Delta t T^*}{\rho c_p} + \frac{\sigma\Delta t}{\rho c_p}\left(\frac{Vm_{j+1}^n - Vm_{j-1}^n}{2\Delta x}\right)^2 \\ 0 \end{bmatrix} \tag{5.14}$$

where $r = \Delta t k/(\rho c_p)$. In this equation, $Vm$ are the results of the first parabolic equation solver in step $2a$) of section 3.4. Furthermore, Neumann boundary conditions are used for the temperature and we applied initial conditions $T^0 = T^*$ when needed. This step is employed from $t = t_n$ to $t = t_n + \Delta t$ and the results for $T^{n+1}$ are saved before advancing the solution to $t_{n+1} = t_n + \Delta t$ and returning to step 1 of section 3.4.

However, note that the inclusion of the joule effect causes numerical challenges as the action potential of the heart is a stiff function. As the joule effect includes the gradient of this function, this value tends to go to infinity. This numerical challenge has also been mentioned in Bini et al. [25]. Therefore, small spatial step-sizes and more stable implicit methods as described in section 3.3 will be necessary for solving these PDEs. We now have the necessary tools to simulate the influence of temperature on the propagation of the cardiac action potential as will be studied in the following sections.

## 5.2 Effect of Temperature Changes on the APD

In order to study the realistic effects of temperature on the cardiac action potential, we must first assign reasonable values to all the necessary constants in equations 5.13, 5.1, 5.2, 5.10 and 5.11. For the Aliev-Panfilov model, we used the values found in Belhamadia et al. [13]. However, to obtain a dimensional version of the wave it was necessary to modify the model in order to obtain a potential with a resting voltage of $-85mV$ and peak value of $40mV$ as described in the previous section. The values of the constants used in the Aliev-Panfilov model are listed in table 5.1.

| constant | value |
|---|---|
| $k$ | 8 |
| $a$ | 0.15 |
| $u_1$ | 0.2 |
| $u_2$ | 0.3 |
| $\epsilon_0$ | 0.002 |
| $V_{rest}$ | $-85$ |
| $V_{peak}$ | 40 |
| $V_{amp}$ | $V_{peak} - V_{rest}$ |
| $V_a$ | $V_{amp}a + V_{rest}$ |

Table 5.1: Constants used in dimensional Aliev-Panfilov

In the case of the bidomain tissue models, many values for the constants are available for equation 5.3. For the surface to volume ratio $\chi$ and the capacitance $C_m$, we used the calibrated parameters obtained by Franzone et al. [42]. In the case of the conductances $M_i$ and $M_e$ we used constant values in the longitudinal directions. These values were taken from LeGuyader et al. [43] which were experimentally measured from canine myocardial tissue. The above constants are given in table 5.2.

For Pennes' Bioheat equation 5.7, there is limited experimental data available for the heat properties of cardiac tissue [25]. However, for the values of the

| constants | values | units |
|---|---|---|
| $\chi$ | 1000 | $cm^{-1}$ |
| $C_m$ | 1 | $\mu F/cm^2$ |
| $M_i$ | 2 | $\mu A/mV/cm$ |
| $M_e$ | 3 | $\mu A/mV/cm$ |

Table 5.2: Constants used in bidomain tissue model

thermal conductivity $k$, the density $\rho$ and the heat capacity $c_p$, experimentally determined values for cardiac muscle were presented in Mcintosh and Anderson [44]. For $\sigma$, it was deemed appropriate to simply use the value for the conductivity of the extracellular tissue. Finally, for the metabolic and blood perfusion term $b_c$ we were unable to find an experimental equivalent for cardiac tissue. However, in Tasaki et al. [40] [41] the heat caused by the propagation of the electric wave in both an olfactory and myellinated nerve fiber was rapidly reabsorbed by the medium ($\sim 30ms$). Therefore, despite possible differences in behavior of myocytes to nerve cells we performed a parametric study in order to reproduce this re-absorption as performed in Bini et al. [25]. For the constants used in Pennes' Bioheat equation please refer to table 5.3

| constants | values | units |
|---|---|---|
| $\rho$ | $1.084 \times 10^{-3}$ | $kg/cm^3$ |
| $c_p$ | 3676 | $J/kg/^oC$ |
| $k$ | $5.6 \times 10^{-6}$ | $J/ms/cm/^oC$ |
| $\sigma$ | 3 | $\mu A/mV/cm$ |
| $b_c$ | $8 \times 10^{-5}$ | $J/ms/cm^3/^oC$ |

Table 5.3: Constants used in Pennes' bioheat equation

As the Aliev-Panfilov model has not previously been used to study temperature effects, constants for equations 5.11 and 5.12 are not available in literature in the context of this cell model. However, in Ashihara et al. [45], they provided calculated values for the $Q_{10}$ of both the APD and the conduction velocity (CV)

of simulated cardiac tissue. These $Q_{10}$ values can be calculated as follows:

$$Q_{10}(APD) = \left(\frac{APD_{27^oC}}{APD_{37^oC}}\right)^{10/(37^oC-27^oC)} \tag{5.15}$$

$$Q_{10}(CV) = \left(\frac{CV_{27^oC}}{CV_{37^oC}}\right)^{10/(37^oC-27^oC)} \tag{5.16}$$

As we've determined all of the physical parameters necessary, we can begin by numerically simulating a 1-D tissue of $10cm$ length. The initial condition of $V_m$ is $40mV$ when $x \geq 7cm$ and $-85mV$ otherwise which provides adequate stimulus for the electric wave. The other two initial conditions are $u_e = 0$ and $W = 0$ for the entire length of the fiber. These calculations also used $\Delta t = 0.1$ and $\Delta x = 0.1$ as step-sizes with a simulation time of 1200ms. Furthermore, the APDs were calculated as the time between the two points where the potential is 95% of its resting value of -85mV at the middle of the fiber ($x = 5cm$). Additionally, a constant conduction velocity was assumed and calculated using the time difference between the first crossing of $0.95(V_{rest})$ at locations $x = 0cm$ and $x = 5cm$.

By using a parametric analysis we found that the constants of table 5.4 enabled us to obtain the Q10 values of table 5.5 which are near identical to those stated in Ashihara et al. [45]. The calculated values obtained are also very similar to those experimentally measured in Kiyosue et al. [46]. The waves used to calculate the $Q_{10}(APD)$ can be seen in figure 5.1 and those used to calculate $Q_{10}(CV)$ can be seen in figure 5.2.

| constants | values |
|---|---|
| $A$ | 1 |
| $B$ | 0.081 |
| $Q_{10}$ | 1.9 |

Table 5.4: Temperature constants used in Aliev-Panfilov equations

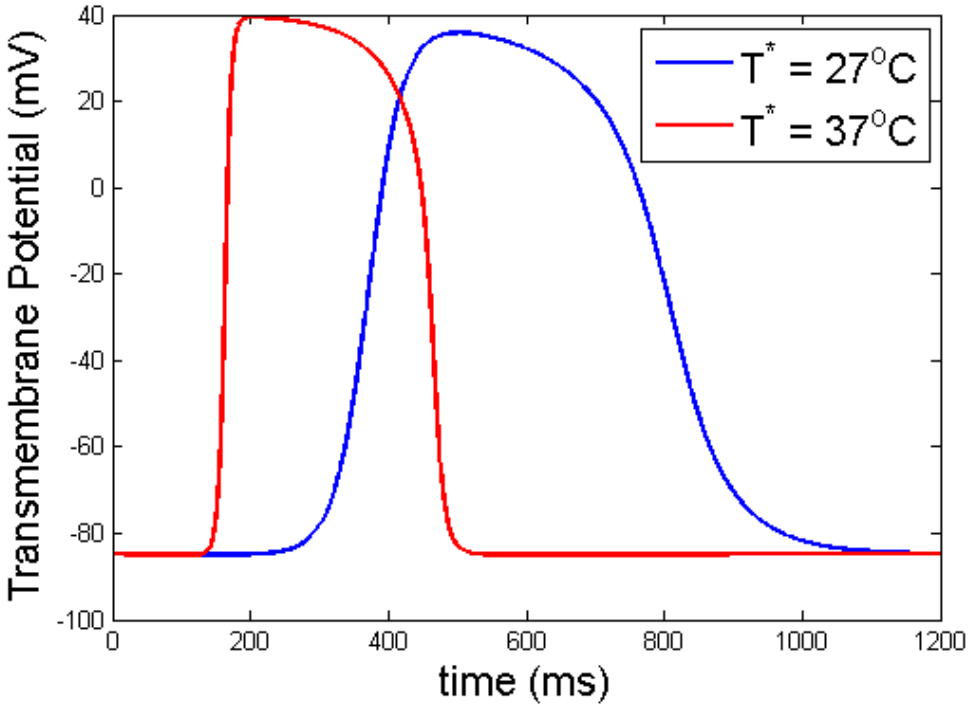
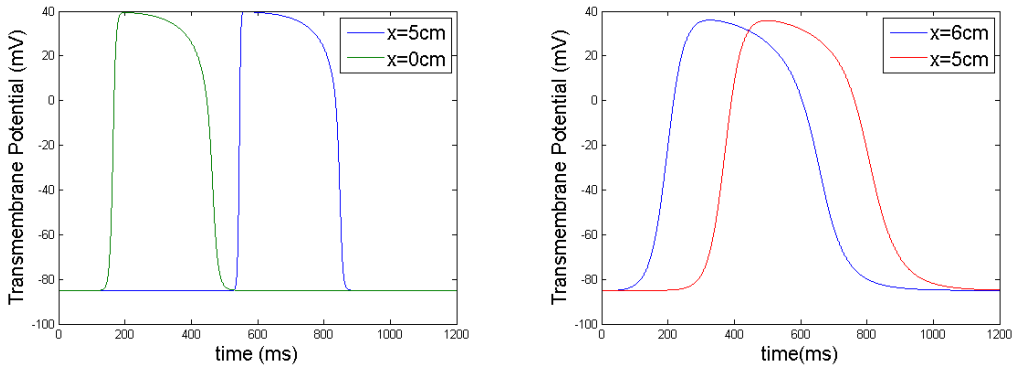



Figure 5.1: Changes in APD used to measure $Q_{10}(APD)$

| | $T^* = 37^oC$ | $T^* = 27^oC$ | $Q_{10}$ measured |
|---|---|---|---|
| $APD(ms)$ | 346 | 694 | 2 |
| $CV(cm/ms)$ | $1.28 \times 10^{-2}$ | $5.85 \times 10^{-3}$ | 2.19 |

Table 5.5: Measured APD, CV and $Q_{10}$ values from simulation

Using these values, we could then investigate the effects of relatively small changes in temperature on the action potential durations. From figure 5.3 we can see that small increases in temperature significantly decreased the APD while a decrease in temperature produced an APD increase. This behavior is as expected when compared to experimental results [32].




(a) $V_m$ used to measure CV at $T^* = 37^oC$

(b) $V_m$ used to measure CV at $T^* = 27^oC$

Figure 5.2: Changes in CV used to measure $Q_{10}(CV)$

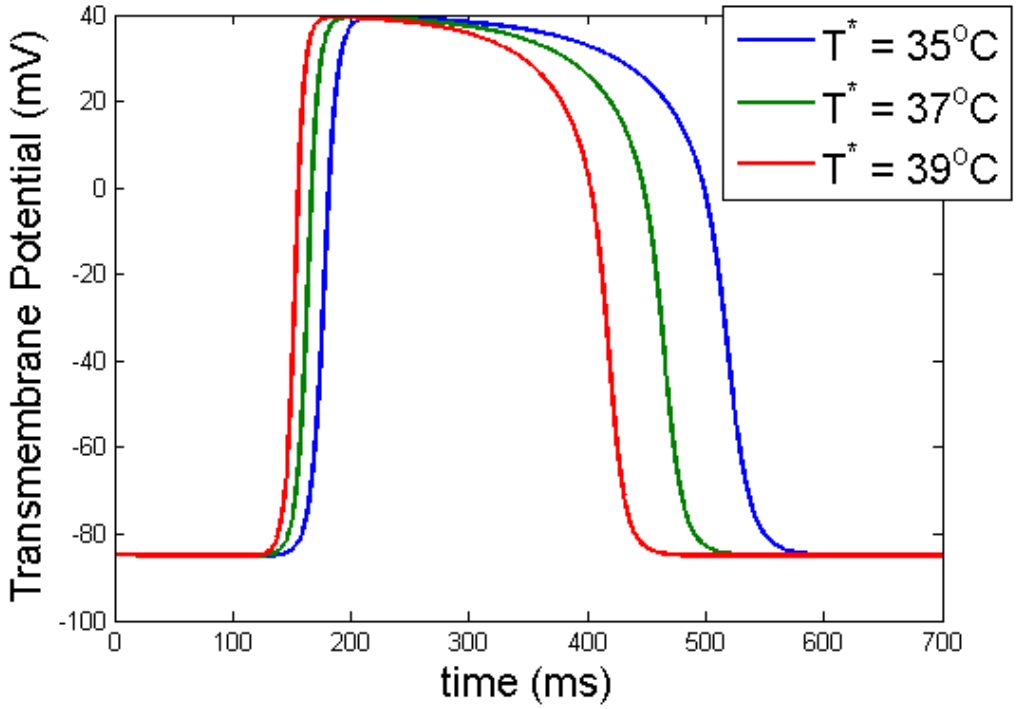
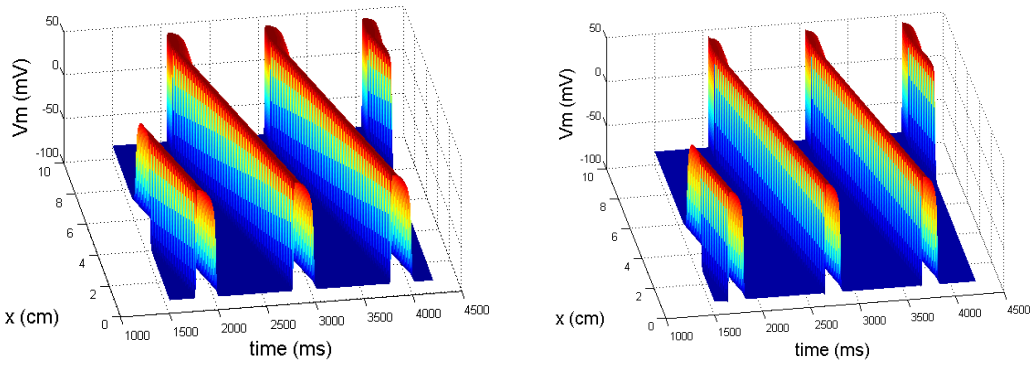



Figure 5.3: Changes in APD due to small modification of tissue temperature

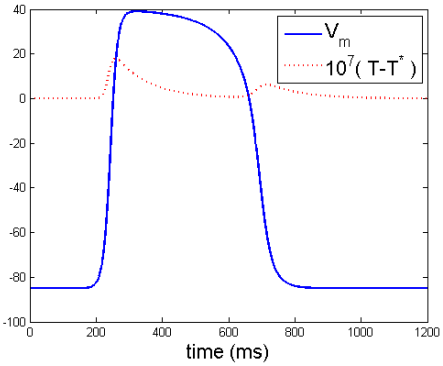
By applying a stimulus current of $I_{stim} = 500$ when $9 \leq x \leq 10$ every $1000ms$ with resting initial conditions for the transmembrane potential, we can further observe this conduct for multiple waves in the surface plots presented in figure 5.4.



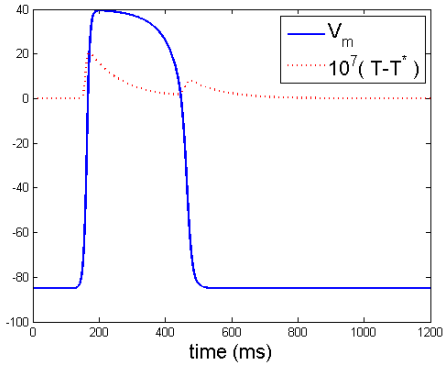

(a) $V_m$ at $T^* = 37^oC$ (b) $V_m$ at $T^* = 43^oC$

Figure 5.4: Surface plots of multiple transmembrane potential waves at different temperatures

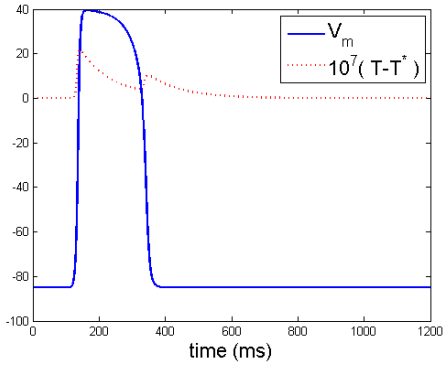
Furthermore, by virtue of the inclusion of Pennes' bioheat equation to the bidomain model it is possible to study the effect of the electrical wave propagation. In figure 5.5 we present the temperature and electric waves at the point $x = 5cm$. From these images, we can see that the temperature increases by $\sim \mu^oC$ whenever the transmembrane potential experiences rapid changes in magnitude. This increase in temperature is then rapidly reabsorbed by the medium. This behavior is similar to those found in the simulations of nerve and intestine cells [23] [25] [47] as well as experimentally observed in myellinated and non-myellinated nerve fibers [40] [41]. This effect has been proposed as a potential method for detecting the tips of spiral waves in tissue [23] [25].




(a) $V_m$ and temperature at $T^* = 30^oC$




(b) $V_m$ and temperature at $T^* = 37^oC$




(c) $V_m$ and temperature at $T^* = 43^oC$

Figure 5.5: Temperature and potential waves at multiple environmental temperatures

The simple Aliev-Panfilov model allows for simulations to be made in a reasonable time frame by avoiding the necessity of solving large systems of stiff ODEs. Therefore, the Aliev-Panfilov model which includes temperature provided an efficient way to ensure that experimental behavior could be replicated in a simplified model. However, many more physiologically relevant and complex models have previously been solved in this thesis. Therefore, in the following section we will attempt to use a similar approach of including temperature effects in the Luo-Rudy cell model in order to increase the realism of the results.

## 5.3 Including Temperature Dependence to the Luo-Rudy Ionic Model

The Luo-Rudy cardiac cell model provides advantages to the simplified Aliev-Panfilov model as it attempts to accurately simulate the underlying ionic activity which influences the transmembrane potential. This allows for more options in studying cardiac dynamics which arise from changes in ion concentrations such as treatments with drugs [17]. Therefore, it was deemed beneficial to attempt to include temperature effects to an ionic cardiac cell model. For this purpose the Luo-Rudy model was used as it is the simplest ionic model previously studied in this thesis.

When adding temperature dependence to the model, we applied the same technique as with Aliev-Panfilov. The linear changes of ionic conductances as represented by equation 5.10 was applied to the transmembrane term. Furthermore, the variation in the rate of change of conductances due to temperature can be included using $Q_{10}$ values as described in equation 5.12. Therefore, these equations can be included in the Luo-Rudy model as follows:

$$I_{ion}(T) = -\eta(T)(I_{ion}) \tag{5.17}$$

$$\frac{dm}{dt} = \phi(T)(\alpha_m(1.0 - m) - \beta_m m) \tag{5.18}$$

$$\frac{dh}{dt} = \phi(T)(\alpha_h(1.0 - h) - \beta_h h) \tag{5.19}$$

$$\frac{dj}{dt} = \phi(T)(\alpha_j(1.0 - j) - \beta_j j) \tag{5.20}$$

$$\frac{dd}{dt} = \phi(T)(\alpha_d(1.0 - d) - \beta_d d) \tag{5.21}$$

$$\frac{df}{dt} = \phi(T)(\alpha_f(1.0 - f) - \beta_f f) \tag{5.22}$$

$$\frac{dx}{dt} = \phi(T)(\alpha_x(1.0 - x) - \beta_x x) \tag{5.23}$$

$$\frac{dC_{ai}}{dt} = \phi(T)(-0.0001I_{si} + 0.07(0.0001 - C_{ai})) \tag{5.24}$$

Where the remaining variables can be found in appendix A.2 and $\eta(T)$ and $\phi(T)$ corresponds to equations 5.10 and 5.12 respectively.

Unlike the simple Aliev-Panfilov model, solving the Luo-Rudy cell model when coupled with the bidomain model offers more numerical complexity. First, the stiff nature of the ionic model necessitates an efficient ODE solver. This condition was satisfied by using the NIRK4 method whose properties are uniquely suited for the ODEs of cardiac cell models as was discussed in chapter 4. Furthermore, the increase in computational time due to this more complex ionic model necessitated the usage of the largest step-sizes possible while retaining accuracy. For our purposes, it was deemed that $\Delta t = 0.2$ and $\Delta x = 0.05$ satisfied these conditions. As with Aliev-Panfilov, the simulations were made of a $10cm$ 1-D fiber for during a time interval from $t = 0$ to $t = 1200ms$. The values of the constants used in both Pennes' bioheat equation and the tissue model are the same as those employed when solving Aliev-Panfilov and can be obtained in tables 5.2 and 5.3. The initial condition used to stimulate the fiber was $V_m = 20mV$ when $x \geq 9cm$ and $V_m = -85mV$ otherwise. For all other initial conditions please refer to table 5.6.

| Variable | Initial Condition |
|---|---|
| $T$ | $T^*$ |
| $u_e$ | 0 |
| $m$ | 0.00167 |
| $h$ | 0.928 |
| $j$ | 1 |
| $d$ | 0.00298 |
| $f$ | 0 |
| $x$ | 0.00602 |
| $Ca_i$ | 0.000178 |

Table 5.6: Initial conditions for simulating bidomain with Luo-Rudy

However, it is still necessary to determine the values for the constants $A$,

$B$ and $Q_{10}$ that will be used in the simulations. Previous work by Kiyosue et al. [46] and Klöckner et al. [48] has experimentally shown that the $Q_{10}$ values are different for the various channel conductances. However, these $Q_{10}$s are of the same magnitude and often have very similar values. Therefore, for the sake of simplicity our model will use only one constant value of $Q_{10}$ for all the conductances. As with Aliev-Panfilov, we manipulated the parameters $Q_{10}$, $A$ and $B$ in an attempt to obtain a final measured value of $Q_{10}(APD) = 2$ and $Q_{10}(CV) = 2.2$ as was obtained in Ashihara et al. [45]. The values for the constants are given in table 5.7 while the measured $APD$, $CV$ and respective $Q_{10}$ values taken where $x = 5cm$ are presented in table 5.8. The waves used to calculate these values are also presented in figures 5.6 and 5.7.

| constants | values |
|---|---|
| $A$ | 1 |
| $B$ | 0.086 |
| $Q_{10}$ | 1.22 |

Table 5.7: Temperature constants used in Luo-Rudy equations

| | $T^* = 37^oC$ | $T^* = 27^oC$ | $Q_{10}$ measured |
|---|---|---|---|
| $APD(ms)$ | 380.4 | 776 | 2.04 |
| $CV(cm/ms)$ | $5.3 \times 10^{-2}$ | $2.45 \times 10^{-2}$ | 2.17 |

Table 5.8: Measured APD, CV and $Q_{10}$ values from Luo-Rudy simulation

In figure 5.6, we can see that there is also a significant decrease in action potential amplitude between the wave where $T^* = 37^oC$ and $T^* = 27^oC$. This was also observed to a lesser degree in the Aliev-Panfilov model as can be seen in figure 5.1. This is consistent with behavior reported in Fenton et al. [26] which also observed a decrease of action potential amplitude with lowering temperature in both experiment and simulation.

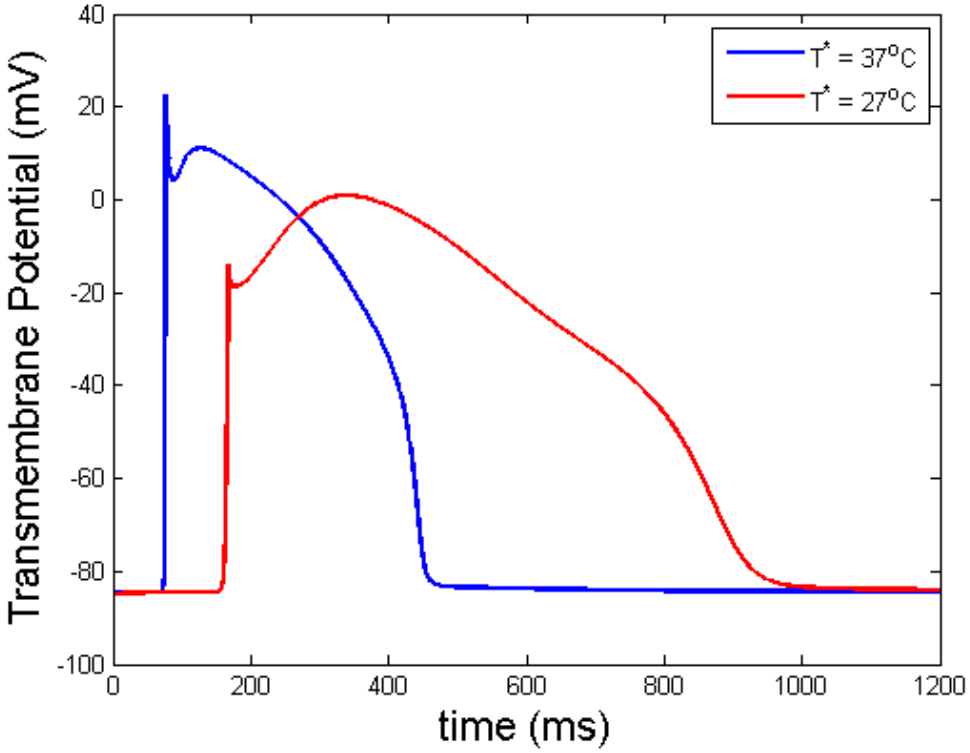
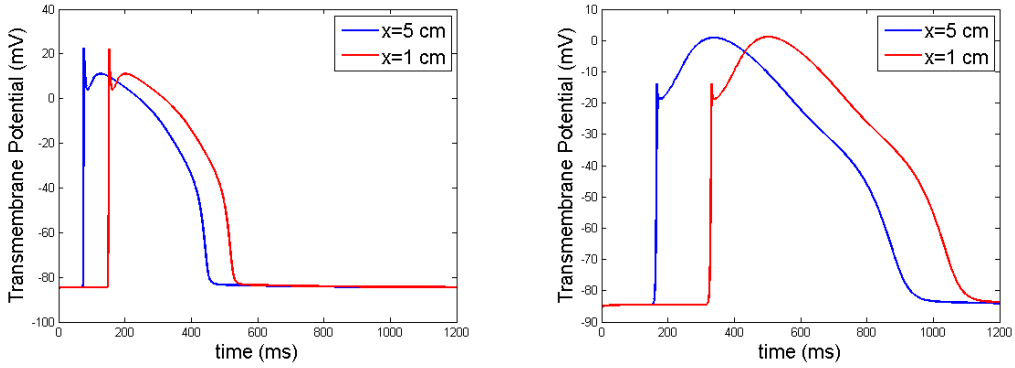



Figure 5.6: Changes in APD used to measure $Q_{10}(APD)$ in Luo-Rudy

Using the constants stated in table 5.7, we can once again study the effect of small temperature changes on the electric potential. In figure 5.8, we can once again see that a small decrease in temperature causes a significant increase in the APD while a small increase in temperature induces the opposite when compared to the reference temperature of $T^* = 37^oC$. This is as expected from our previous results from Aliev-Panfilov as well as experimental results [32].

Furthermore, we can once again investigate the slight heat induced by the transmembrane potential as described by Pennes' bioheat equation. In figure 5.9, we see that a small variation of temperature of $\sim \mu^oC$ is obtained when the potential is rapidly polarised. This behavior is similar to that presented with Aliev-Panfilov as well as previous modelling work and experimentally detected results (see section 5.2). However, unlike in the results of Aliev-Panfilov, there was very little detectable changes in temperature during the repolarization




(a) $V_m$ used to measure CV at $T^* = 37^oC$

(b) $V_m$ used to measure CV at $T^* = 27^oC$

Figure 5.7: Changes in CV used to measure $Q_{10}(CV)$ in Luo-Rudy

phase of transmembrane potential. This can be attributed to the much more gradual return to equilibrium of the wave when compared to Aliev-Panfilov.

Once again, all simulations performed in the current and in the previous section were generated using numerical techniques programmed in C++. These results were then analyzed in MATLAB.

By virtue of the stable and efficient NIRK4 ODE solver, it was possible to simulate the bidomain model coupled with Luo-Rudy in order to investigate the effects of temperature on the transmembrane potential described by this more complex cardiac cell model. The usage of cardiac cell models in comparison to simplified models was preferred as it allows for the investigation of the effect of underlying physiological currents which is of especial interest in drug treatment research. Using only one value for the constants $Q_{10}$, $A$ and $B$ it was possible to obtain results exhibiting behavior similar to both experiment and the previous Aliev-Panfilov simulations. Furthermore, with Luo-Rudy we were able to obtain small changes in temperature caused by the transmembrane potential although the form of this wave was slightly different as previously seen in Aliev-Panfilov.

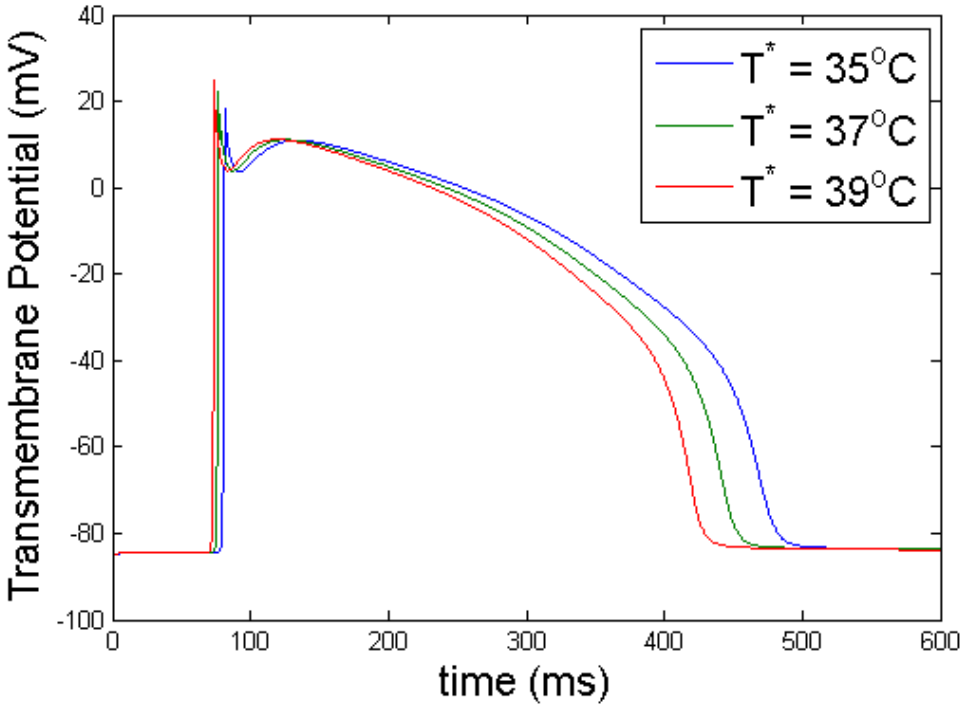
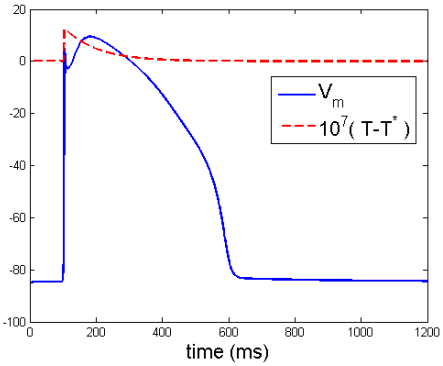



Figure 5.8: Effect of small temperature changes on APD in Luo-Rudy

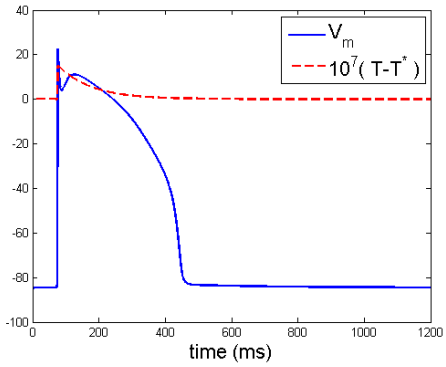
## 5.4 Spiral Wave Detection with Action Potential Induced Temperature Changes

In the previous sections, we mentioned that the heat produced by the action potential has been proposed as a possible method for detecting spiral waves using thermal detectors. This was advocated due to the modelling work by Bini et al. [25] [23] which found that this temperature peaks near the tip of the spiral wave when simulating nerve cells coupled with the monodomain model. Therefore, in this section we will investigate whether this behavior will also be present when simulating cardiac action potentials using the bidomain model.

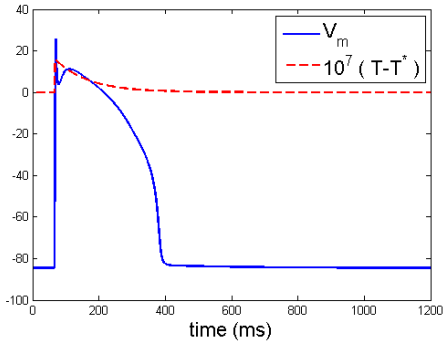
The detection of spiral-waves are of considerable interest to cardiac researchers as the phenomena has been linked with tachycardia and fibrillation. However, it is not possible to study spiral waves in one dimension. Therefore, in this section it will be necessary to extend our model to a two dimensional




(a) $V_m$ and temperature at $T^* = 30^oC$




(b) $V_m$ and temperature at $T^* = 37^oC$




(c) $V_m$ and temperature at $T^* = 43^oC$

Figure 5.9: Temperature and potential waves at differing temperatures using Luo-Rudy

surface. For this purpose, we will use the MEF++ numerical software provided by the GIREF group at Laval University[1]. Our model has been simulated using a quadratic finite element method for spatial discretization along with a fully implicit second order scheme for the temporal derivatives. Further information on the usage of this method for solving cardiac tissue using the Aliev-Panfilov model can be found in Belhamadia et al. [49].

Using MEF++, we will first attempt to simulate the effect of temperature on the regular action potential of 2-D tissue with the Aliev-Panfilov model and then extend our work to the spiral wave.

[1]For more information on GIREF's work and the MEF++ software please visit: `http://www.giref.ulaval.ca/mef.html`

### 5.4.1 Regular 2-D Wave Simulations

In these simulations, we will be using identical constants as those previously presented in the 1-D case of section 5.2. Therefore, the values of the constants for Aliev-Panfilov, the bidomain model, Pennes' equation and the temperature constants can be viewed in tables 5.1, 5.2, 5.3 and 5.4 respectively. Additionally, the values for the conductivities used are the same irrespective of direction. Furthermore, the initial conditions applied are as follows:

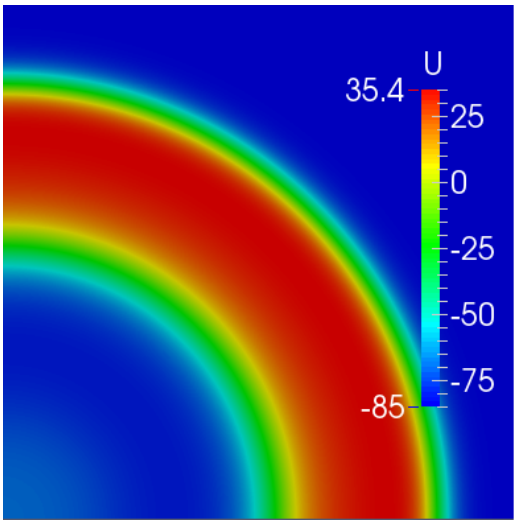
$$V_m = \begin{cases} 40 & \sqrt{x^2+y^2} < 3 \\ -85 & otherwise \end{cases}$$

$$u_e = 0 \qquad W = 0 \qquad T = T^*$$

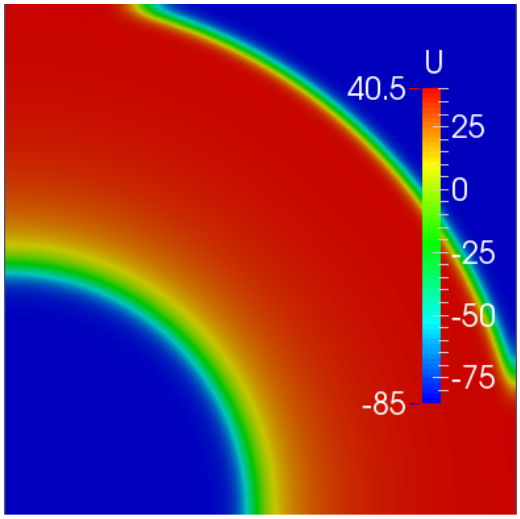
Where $x$ and $y$ are the spatial values. Finally, the $10cm \times 10cm$ sheet was simulated for $1300ms$ and the results can be seen in figure 5.10.

In figures 5.10a and 5.10b, we can see the images for the transmembrane potentials at $T^* = 27^oC$ and $T^* = 37^oC$ obtained at a single point in time. Here we observe that the area of the excited potential is larger for the warmer tissue. This is due to the faster speed of propagation at higher temperatures. Furthermore, in figures 5.10c and 5.10d we can see that a region of elevated temperature is formed at the front of the propagating transmembrane wave. This is as expected from the 1-D results of figure 5.5. Finally, by plotting the values for both the action potential and the temperature at the central point $(5, 5)$ for the entire interval of simulation, we obtained figures 5.10e and 5.10e. From these figures we observe that the decrease in temperature produces an increase in action potential duration as was the case in all previous 1-D simulations. All 2-D images were produced using paraview while the plots were generated in MATLAB.

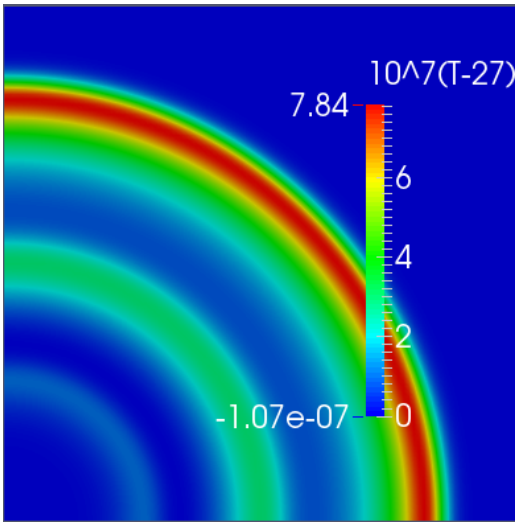
$T^* = 27^oC$ | $T^* = 37^oC$



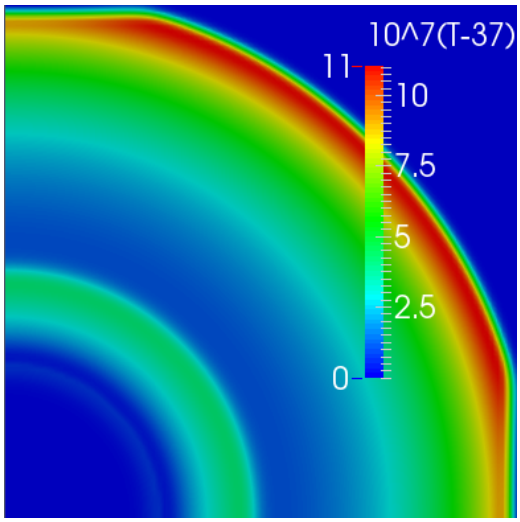

(a) $V_m$ at t=747ms



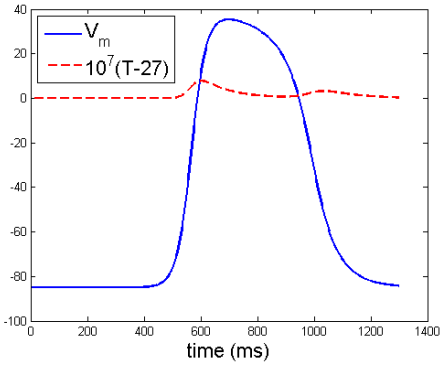

(b) $V_m$ at t=408ms



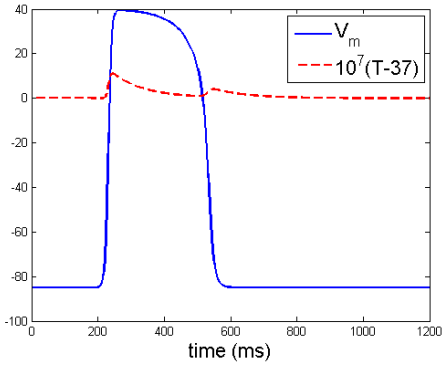

(c) $10^7(T - T^*)$ at t=747ms




(d) $10^7(T - T^*)$ at t=408ms




(e) $V_m$ and $10^7(T - T^*)$ at the center point (5,5)




(f) $V_m$ and $10^7(T - T^*)$ at the center point (5,5)

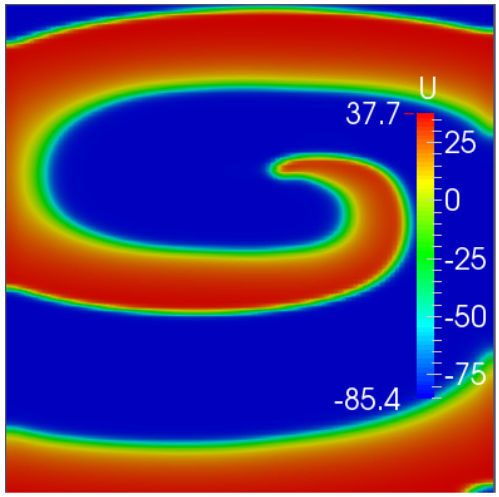
Figure 5.10: 2D simulations of cardiac tissue at $T^* = 27^oC$ and $T^* = 37^oC$

### 5.4.2 2-D Spiral Wave Simulations

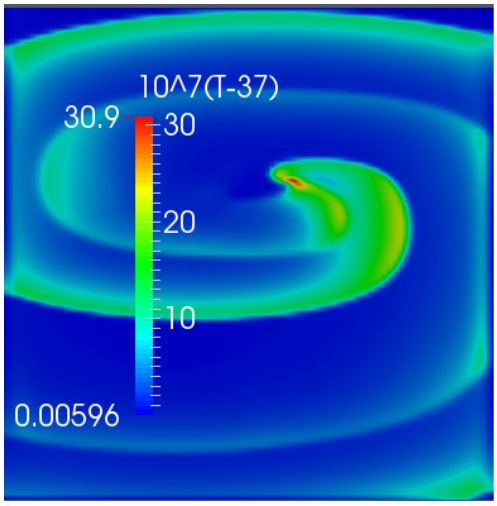
Now that we are confident of the accuracy of the 2-D model, we can investigate the behavior of spiral waves. In order to produce spiral waves it is necessary to use more realistic tissue constants. Therefore, we will employ conductivities which differ based on the direction of the myocardial fibers. This is achieved by employing a tensor instead of a constant for the $M_i$, $M_e$ and $\sigma$ values in the bidomain model. The following tensor values were obtained from Belhamadia et al [49] and applied to our model:

$$\mathbf{M}_i = \begin{pmatrix} 3 & 0 \\ 0 & 0.32 \end{pmatrix} \qquad \mathbf{M}_e = \begin{pmatrix} 2 & 0 \\ 0 & 1.24 \end{pmatrix} \qquad \sigma = \begin{pmatrix} 2 & 0 \\ 0 & 1.24 \end{pmatrix}$$

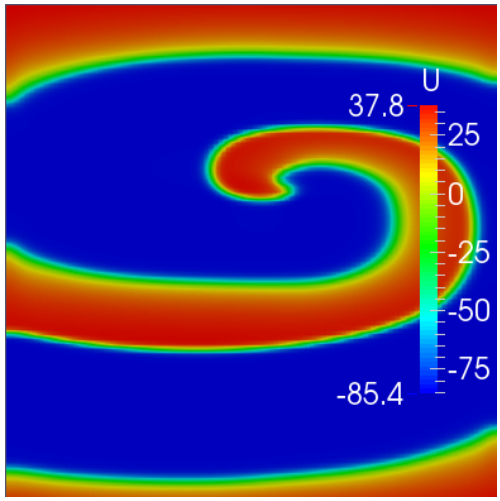
In order to obtain the desired shape of the wave, we simply used a spiral wave obtained on a fine mesh as the initial conditions. All other constants used are identical to those provided in the previous 2-D example. Furthermore, in these simulations a constant environmental temperature of $T^* = 37^oC$ was maintained. A time evolution for the 2-D spiral wave action potential and its induced temperature increases are presented in figure 5.11. Using figure 5.11, we can deduce that as the spiral wave evolves, an accumulation of heat develops near the tip of the spiral wave. Thus, we have confirmed that this property could potentially be used for detecting the location of spiral tips. All 2-D simulations were visualized using Paraview software.



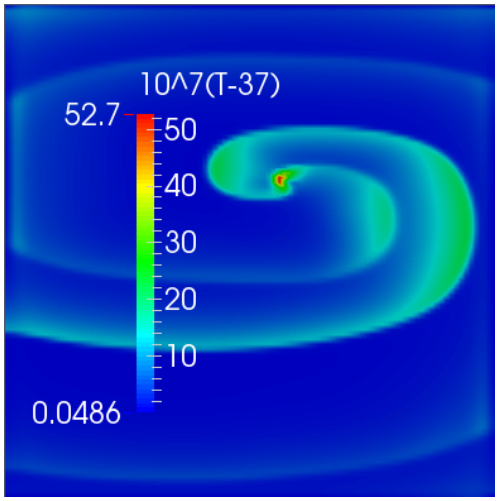

(a) $V_m$ at t=600ms



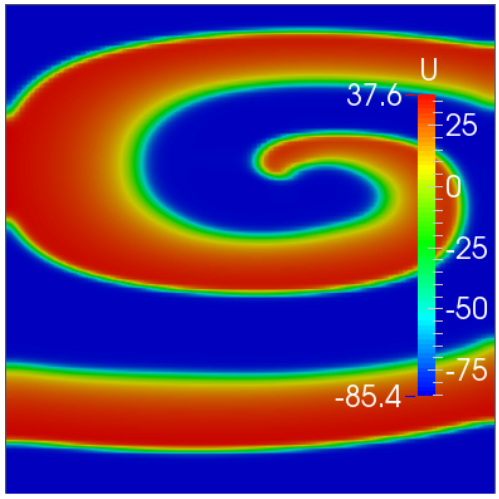

(b) $10^7(T - T^*)$ at t=600ms



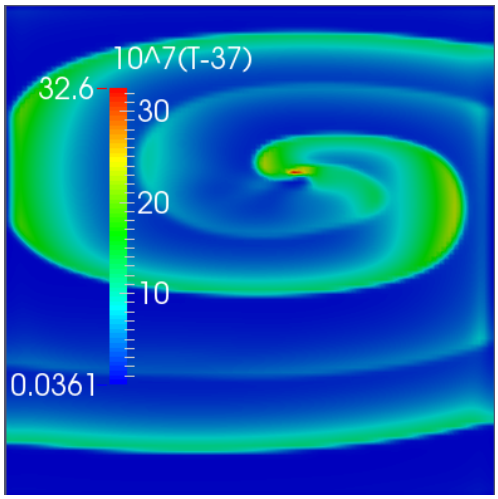

(c) $V_m$ at t=720ms




(d) $10^7(T - T^*)$ at t=720ms




(e) $V_m$ at t=1080ms




(f) $10^7(T - T^*)$ at t=1080ms

Figure 5.11: Spiral waves and its induced heat

In this chapter, we presented two possible ionic models for studying the effect of temperature changes on the electrical activity in the heart. Aliev-Panfilov is a simplified model which has the advantage of computational efficiency. Thus, with Aliev-Panfilov it is possible to study the effects of temperature in 2-D and 3-D tissue within a reasonable time frame using standard computers. However, the added computational complexity of studying an ionic cardiac cell model such as Luo-Rudy can be decreased significantly by using more efficient ODE solvers such as the NIRK4 method which was introduced in the previous chapter. The underlying physiological basis for this model allows for the investigation of effects that are outside the scope of Aliev-Panfilov such as drug treatments. Nevertheless, both models exhibited a decrease in conduction velocity, an increase in action potential duration and a decrease in amplitude when lowering temperature. This is consistent with both experimental results and previous modelling work. Finally, using Pennes' bioheat equation we could simulate the heat produced by the action potential duration due to the joule effect. This could potentially be used as a method of detecting spiral-waves, a phenomena that is present in deadly cardiac arrhythmias. In the following chapter, we will provide a brief conclusion of the results as well as highlighting the possible limitations and future directions suggested by our research.

# Chapter 6

# Conclusion

Modelling the electrical activity of the heart presents numerous numerical challenges. However, these models offer insight into multiple phenomena such as cardiac arrhythmias which have proved difficult to study in vivo. Currently, the most accurate tissue for simulating the action potential in cardiac tissue is that of the bidomain model. This model is comprised of a coupling of the cellular domain, represented as a system of ODEs describing the influence of ionic movements through the membrane on the electric potential, and the tissue domain represented as a system of PDEs. Despite major advances in computer technology, the stiff nature of the potentially large system of ODEs can limit the usage of these models for realistic 3D simulations as the computational time and resources can be very extensive. However, by using more complex and stable numerical methods, it is possible to decrease the computational time of the cardiac cell solving component which can translate to significant decreases in overall simulation time.

In chapter 4 of this thesis, we presented a fourth order nested implicit Runge-Kutta method for solving the cardiac cell models. Due to its internal explicit steps, the NIRK4 method could be simplified into one implicit equation that requires a single application of Newton's method. This fully implicit method was therefore deemed uniquely well suited for cardiac cell modelling

as the increase in stability could be obtained while limiting the increase in computational complexity caused by the large linear systems of other high order implicit methods. Therefore, in order to investigate the performance of NIRK4, we compared the computational times and accuracy of this method with that of the more popular ESDIRK3 and SDIRK4 solvers which have previously been studied as a means of solving these systems of ODEs. In these tests it was found that NIRK4 usually presented a decrease in computational time and increase in accuracy for the Luo-Rudy and Fox cell models. Having been satisfied by its performance when compared to other implicit solvers, we attempted to investigate its capability when contending with the most popular numerical method used in cardiac cell modelling: the Rush-Larsen method of order 1 and 2. Therefore, we compared the step-size and computational time in order to obtain an accuracy within 5%RRMS error of a reference solution using the NIRK4, Rush-Larsen of order 1 and Rush-Larsen of order 2 solvers. In these simulations, it was found that NIRK4 had superior performance when compared to RL1 as well as performing better than RL2 in the majority of instances. This was tested on the Luo-Rudy, Fox, Courtemanche and Tusscher cell models which exhibit a range of complexities.

However, there are some limitations to the results presented in chapter 4. First, most coupling techniques used when combining the cell and tissue models are limited to first and second orders of accuracy. Therefore, the high degree of accuracy of NIRK4 would be restricted to that of this coupling method. Nevertheless, due to the increase in stability of NIRK4 it is believed that the resulting usage of larger step-sizes could produce an overall decrease in computational time. This behavior was observed in the comparisons between NIRK4 and the RL1 and RL2 methods. In order to ensure that the advantages of NIRK4 continue when simulating cardiac tissue, it would be beneficial to compare the performance of NIRK4 with other solvers when simulating cardiac tissue using the bidomain model.

By coupling the NIRK4 solver with implicit elliptic and parabolic PDE solvers, it was possible to accurately simulate cardiac tissue. However, the models presented up to this point did not include any influence that could be caused by temperature effects. This can be of especial interest to heart tissue modellers as recent studies have investigated the influence of temperature on action potential duration: a property associated with the formation of spiral-waves and arrhytmias. Therefore, by including terms which reflect both the linear changes and variations in the rate of changes of ionic conductances due to temperature, it was possible to modify the Aliev-Panfilov and Luo-Rudy models in order to add temperature dependence. With these models, we observed increases in action potential durations and decreases in conduction velocities with the lowering of environmental temperature. This resulted in measured $Q_{10}$s for the CVs and APDs similar to those previously provided in literature. Furthermore, using Pennes' bioheat equation coupled with the bidomain model and the joule effect, we could simulate the heat produced by the transmembrane potential. This effect can be used as a possible method for detecting spiral waves in cardiac tissue as can be seen in the series of spiral wave images.

Nevertheless, a limitation of the temperature dependent cardiac models presented is the lack of experimental results. As there is very little information available for heat produced by the transmembrane potential in the heart, it was necessary to verify our results with that of nerve models when choosing the values of the constants in the bioheat equation. Furthermore, the values of the $Q_{10}$s in the Luo-Rudy model were assumed to be the same regardless of the ionic conductances which is not experimentally accurate. This could have a non-physiological influence on the properties of the action potential results. Nevertheless, until more experimental data is available it is necessary to make reasonable assumptions in order to obtain results which exhibit realistic behavior. In future studies, it will be advantageous to study the possible usage of temperature modifications as a means of controlling spiral waves.

Using a first order splitting technique with implicit PDE solvers and a new efficient and stable fourth order Runge-Kutta method, it was possible to simultaneously simulate the action potential dependence on temperature as well as the heat produced by this electric wave. With this tool, the investigation of interesting temperature based phenomena can be studied.

# Bibliography


[1] National Vital Statistics Reports (NVSR) U.S. National Center for Health Statistics. Deaths: Final data for 2007. 58(19), 2010.

[2] A.L. Hodgkin and A.F. Huxley. A quantitative description of membrane current and its application to conduction and excitation in nerve. J. Physiol., 117:500544, 1952.

[3] S. Linge, J. Sundnes, M. Hanslien, G.T. Lines, and A. Tveito. Numerical solution of the bidomain equations. Phil. Trans. R. Soc. A, 367(1895):19311950, 2009.

[4] L. Tung. A bidomain model for describing ischemic myocardial D-C potentials. Ph.D. thesis, M.I.T.., 1978.

[5] "Alan L. Hodgkin - Facts". Nobelprize.org. Nobel Media AB 2013. Web. 13 Nov 2013. <http://www.nobelprize.org/nobel_prizes/medicine/laureates/1963/hodgkin-facts.html>

[6] A. L. Hodgkin, A. F. Huxley and B. Katz. Measurement of Current-Voltage Relations in the Membrane of the Giant Axon of Loligo. J. Physiol. ii6, 424-448, 1952.

[7] CH Luo and Y Rudy. A model of the ventricular cardiac action potential. Depolarization, repolarization, and their interaction. Circulation Research Vol 68 No 6, 1501-1526, June 1991.

[8] Joseph M. Pastore, Steven D. Girouard, Kenneth R. Laurita, Fadi G. Akar, and David S. Rosenbaum. Mechanism Linking T-Wave Alternans to the Genesis of Cardiac Fibrillation. Circulation. 99: 1385-1394, 1999.

[9] Jeffrey J. Fox, Jennifer L. McHarg, Robert F. Gilmour. Ionic mechanism of electrical alternans. American Journal of Physiology - Heart and Circulatory Physiology, 282(2)H516-H530. Feb 2002.

[10] M. Courtemanche, R. Ramirez, and S. Nattel. Ionic mechanisms underlying human atrial action potential properties: insights from a mathematical model. Am J Physiol Heart Circ Physiol. 275: 301-321. 1998.

[11] K. H. W. J. ten Tusscher, A. V. Panfilov. Alternans and spiral breakup in a human ventricular tissue model. American Journal of Physiology - Heart and Circulatory Physiology. 291(3) H1088-H1100. Sep 2006.

[12] Nagumo, J., Arimoto, S., Yoshizawa, S. An Active Pulse Transmission Line Simulating Nerve Axon. Proceedings of the IRE . vol.50, no.10. pp.2061-2070, Oct. 1962.

[13] Youssef Belhamadia, Andr Fortin, Yves Bourgault. Towards accurate numerical method for monodomain models using a realistic heart geometry. Mathematical Biosciences. 220 : 89101. 2009.

[14] Joakim Sundnes, Robert Artebrant, Ola Skavhaug, Aslak Tveito, A Second-Order Algorithm for Solving Dynamic Cell Membrane Equations, IEEE Trans Biomed Eng. VOL. 56, NO. 10: 2546-2548. October 2009.

[15] Joakim Sundnes, Glenn Terje Lines, Xing Cai, Bjrn Fredrik Nielsen, Kent-Andre Mardal and Aslak Tveito. Computing the Electrical Activity in the Heart. Monographs in Computational Science and Engineering. Vol 1. 2006.

[16] Joakim Sundnes, Glen Lines, Aslak Tveito. Efficient solution of ordinary differential equations modeling electrical activity in cardiac cells. Mathematical Biosciences. 172 55-72, 2001.

[17] Rubin R. Aliev and Alexander V. Panfilov. A Simple Two-variable Model of Cardiac Excitation. Chaos, Solitons and Fractals Vol. 7, No. 3, pp. 293-301, 1996.

[18] E. Hairer and G. Wanner. Solving ordinary differential equations II. Berlin: Springer-Verlag, Springer Series in Computational Mathematics, 1992.

[19] Spiteri, R.J. and Dean, R.C. On the Performance of an ImplicitExplicit Runge–Kutta Method in Models of Cardiac Electrical Activity. IEEE Transactions on Biomedical Engineering. vol.55, no.5. pp.1488,1495. May 2008.

[20] G.Yu. Kulikov, S.K. Shindin, On a family of cheap symmetric one-step methods of order four, in: V.N. Alexandrov, et al. (Eds.), Computational Science ICCS 2006, Proceedings, Part I, 6th International Conference, Reading, UK, May 2831, 2006, Lecture Notes in Computer Science, vol. 3991, Springer-Verlag, Berlin, pp. 781785. 2006.

[21] S. Rush and H. Larsen. A practical algorithm for solving dynamic membrane equations. IEEE Trans. Biomed. Eng., 25(4):389-392, 1978.

[22] L. Tung.A Bi-domain model for describing ischemic myocardial D-C potentials. PhD thesis, MIT, Cambridge, MA, 1978.

[23] D. Bini, C. Cherubini and S. Filippi. Heat transfer in Fitzhugh-Nagumo models. Physical Review E74, 041905. 2006.

[24] RF Jr Gilmour and DR Chialvo. Editorial: electrical restitution, critical mass and the riddle of fibrillation. J Cardiovasc Electrophysiol 10:10871089, 1999.

[25] D. Bini, C. Cherubini and S. Filippi. On vortices heating biological excitable media. Chaos, Solitons and Fractals, 42, 20572066. 2009.

[26] F. H. Fenton, A. Gizzi, C. Cherubini, N. Pomella and S. Filippi. Role of temperature on nonlinear cardiac dynamics. Physical Review E. Vol. 87, No. 4, 042717. 2013.

[27] S. Filippi, A. Gizzi, C. Cherubini, S. Luther and F. Fenton. Mechanistic insights into hypothermic ventricular fibrillation: the role of temperature and tissue size. Europace, 16, 424434. 2014.

[28] A. Pullan, M. Buist, L. Cheng. Mathematially Modelling the Electrical Activity of the Heart: From Cell to Body Surface and Back Again. World Scientific Publishing. 2005.

[29] M. Marsh, S. Ziaratgahi, R. Spiteri. The Secrets to the Success of the RushLarsen Method and its Generalizations. IEEE Transactions On Biomedical Engineering, Vol. 59, No. 9, 2506-2515. 2012.

[30] R. Clayton, A. Panfilov. A guide to modelling cardiac electrical activity in anatomically detailed ventricles. Progress in Biophysics and Molecular Biology, 96, 1943. 2008.

[31] J. Grenier, Y. Belhamadia. Nested Implicit Runge-Kutta Method for Simulating Cardiac Cell Models. Internation Journal of Bioelectromagnetism. Vol. 14, No. 1, 40-45. 2012.

[32] H. Bjørnstad, P. Tande, D. Lathrop and H. Refsum. Effects of temperature on cycle length dependent changes and restitution of action potential

duration in guinea pig ventricular muscle. Cardiovascular Research. 27, 946-950. 1993.

[33] M. Yamazaki, Haruo Honjo, T. Ashihara, M. Harada, I. Sakuma, K. Nakazawa, N. Trayanova, M. Horie, J. Kalifa, J. Jalife, K. Kamiya, I. Kodama. Regional cooling facilitates termination of spiral-wave reentry through unpinning of rotors in rabbit hearts. Heart Rhythm, Vol 9, No 1, 107-114. 2012.

[34] H. Pennes. Analysis of Tissue and Arterial Blood Temperatures in the Resting Human Forearm. Journal of Applied Physiology. Vol 1, No 2, 93-122. 1948.

[35] R. FitzHugh. Theoretical Effect of Temperature on Threshold in the Hodgkin-Huxley Nerve Model. The Journal of General Physiology. Volume 49, 989-1005. 1966.

[36] J.W. Moore. Temperature and drug effects on squid axon membrane ion conductances. Fed Proc. 17:113. 1958.

[37] V. Elharrar and B. Surawicz. Cycle Length Effect on Restitution of Action Potential Duration in Dog Cardiac Fibers. Am J Physiol Heart Circ Physiol, Vol 244, No 6, H782-H792. 1983.

[38] M.P. Nash and A.V Panfilov. Electromechanical Model of Excitable Tissue to Study Reentrant Cardiac Arrhythmias. Progress in Biophysics & Molecular Biology. volume 85, 501-522. 2004.

[39] R. V. Davalos, B. Rubinsky and L. M. Mir. Theoretical Analysis of the Thermal Effects During in Vivo Tissue Electroporation. Bioelectrochemistry. volume 61, number 1-2, 99-107. 2003.

[40] I. Tasaki, K. Kusano and P. M. Byrne. Rapid mechanical and thermal

changes in the garfish olfactory nerve associated with a propagated impulse. Biophysical Journal. Vol 55, 1033-1040. 1989.

[41] I. Tasaki and P. Byrne. Heat Production Associated with a propagated impulse in Bullfrog Myelinated Nerve Fibers. Japanese Journal of Physiology. Vol 42, 805-813. 1992.

[42] P. C. Franzone, L.F. Pavarino and B. Taccardi. Simulating patterns of excitation, repolarization and action potential duration with cardiac Bidomain and Monodomain models. Mathematical Biosciences. 197, 3566. 2005.

[43] P. LeGuyader, P. Savard and F. Trelles. Measurement of Myocardial Conductivities with an Eight-Electrode Technique in the Frequency Domain. Engineering in Medicine and Biology Society. Vol 1, 71-72. 1995.

[44] R.L Mcintosh and V. Anderson. A Comprehensive Tissue Properties Database Provided For The Thermal Assessment of a Human at Rest. Biophysical Reviews and Letters. Vol. 5, No. 3, 129151. 2010.

[45] T. Ashihara, N. Trayanova, K. Nakazawa, M. Yamazaki, H. Honjo, I. Sakuma, K. Kamiya and I. Kodama. Spiral Wave Control by Regional Cooling in a Bidomain Model. Heart Rhythm. Vol 2, No 5, S220. 2005.

[46] T. Kiyosue, M. Arita, H. Muramatsu, A. J. Spindler and D. Noble. Ionic Mechanisms of Action Potential Prolongation at Low Temperature in Guinea-Pig Ventricular Myocytes. Journal of Physiology. 468, pp. 85-106. 1993.

[47] A Gizzi, C. Cherubini, S. Migliori, R. Alloni, R. Portuesi and S. Filippi. On the electrical intestine turbulence induced by temperature changes. Phys. Biol. 7, 016011. 2010

[48] U. Klöckner, A. Schiefer and G. Isenberg. L-type Ca-channels: similar $Q_{10}$ of Ca-, Ba- and Na- conductance points to the importance of ion-channel interaction. Plügers Archiv. Vol. 415, No. 5, 638-641. 1990.

[49] Y. Belhamadia, A. Fortin, Y. Bourgault. On the performance of anisotropic mesh adaptation for scroll wave turbulence dynamics in reactiondiffusion systems. Journal of Computational and Applied Mathematics. Vol. 271, 233246. 2014.

[50] F. Poveda, D. Gil, E. Mart, A. Andaluz, M. Ballester, F. Carreras. Helical Structure of the Cardiac Ventricular Anatomy Assessed by Diffusion Tensor Magnetic Resonance Imaging With Multiresolution Tractography. Rev Esp Cardiol. Vol 66, No. 10, 782-790. 2013.

[51] J. Marcé-Nogué, G. Fortuny, M. Ballester-Rodés, F. Carreras, F. Roure. Computational modeling of electromechanical propagation in the helical ventricular anatomy of the heart. Computers in Biology and Medicine, Vol. 43, 16981703. 2013.

# Appendices

# Appendix A

# Cardiac Cell Models

## A.1 Hodgkin-Huxley Model

$$ER = -75 \tag{A.1}$$

$$Cm = 1 \tag{A.2}$$

$$ENa = ER + 115 \tag{A.3}$$

$$EK = ER - 12 \tag{A.4}$$

$$EL = ER + 10.613 \tag{A.5}$$

$$g_{Na} = 120 \tag{A.6}$$

$$g_K = 36 \tag{A.7}$$

$$g_L = 0.3 \tag{A.8}$$

$$\alpha_m = \frac{-0.1(V + 50)}{\exp(-0.1(V + 50)) - 1} \tag{A.9}$$

$$\beta_m = 4\exp(-(V + 75)/18) \tag{A.10}$$

$$\alpha_h = 0.07\exp(-(V + 75)/20) \tag{A.11}$$

$$\beta_h = \frac{1}{(\exp(-0.1(V+45))+1)} \tag{A.12}$$

$$\alpha_n = \frac{(-0.01(V+65))}{(\exp(-0.1(V+65))-1)} \tag{A.13}$$

$$\beta_n = 0.125\exp((V+75)/80) \tag{A.14}$$

$$I_{Na} = g_{Na}m^3h(V-ENa) \tag{A.15}$$

$$I_K = g_K n^4(V-EK) \tag{A.16}$$

$$I_L = g_L(V-EL) \tag{A.17}$$

$$Iion = I_{Na} + I_K + I_L \tag{A.18}$$

$$\frac{dV}{dt} = \frac{-(Iion - Is)}{Cm} \tag{A.19}$$

$$\frac{dm}{dt} = ((\alpha_m(1-m)) - (\beta_m m)) \tag{A.20}$$

$$\frac{dh}{dt} = ((\alpha_h(1-h)) - (\beta_h h)) \tag{A.21}$$

$$\frac{dn}{dt} = ((\alpha_n(1-n)) - (\beta_n n)) \tag{A.22}$$

## A.2 Luo-Rudy I model

$$I = \begin{cases} -25.5 & 0 \le t \le 1 \\ 0 & otherwise \end{cases} \tag{A.23}$$

$$Kplusi = 145.0 \tag{A.24}$$

$$Kpluse = 5.4 \tag{A.25}$$

$$Naplusi = 18.0 \tag{A.26}$$

$$Napluse = 140.0 \tag{A.27}$$

$$gNa = 23.0 \tag{A.28}$$

$$gKp = 1.83 * 10.0^{-2.0} \tag{A.29}$$

$$gb = 3.921 * 10.0^{-2.0} \tag{A.30}$$

$$Cm = 1.0 \tag{A.31}$$

$$Eb = -59.87 \tag{A.32}$$

$$R = 8314.0 \tag{A.33}$$

$$T = 310.0 \tag{A.34}$$

$$F = 96484.6 \tag{A.35}$$

$$PR = 1.833 * 10.0^{-2.0} \tag{A.36}$$

$$ENa = \frac{RT}{F} \log(\frac{Napluse}{Naplusi}) \tag{A.37}$$

$$\alpha_x = (5.0 * 10.0^{-4.0}) \frac{(\exp(0.083(V + 50.0))}{\exp(0.057(V + 50.0)) + 1.0} \tag{A.38}$$

$$\beta_x = (1.3 * 10.0^{-3.0}) \frac{\exp(-0.06(V + 20.0)}{(\exp(-0.04(V + 20.0)) + 1.0} \tag{A.39}$$

$$\alpha_m = \frac{0.32(V + 47.13)}{1.0 - \exp(-0.1(V + 47.13))} \tag{A.40}$$

$$\beta_m = 0.08 \exp(\frac{-V}{11.0}) \tag{A.41}$$

$$\alpha_d = (0.095) \frac{\exp(-0.01(V - 5.0))}{(\exp(-0.072(V - 5.0)) + 1.0)} \tag{A.42}$$

$$\beta_d = 0.07 \frac{\exp(-0.017(V + 44.0))}{\exp(0.05(V + 44.0)) + 1.0} \tag{A.43}$$

$$\alpha_f = 0.012 \frac{\exp(-0.008(V + 28.0))}{\exp(0.15(V + 28.0)) + 1.0} \tag{A.44}$$

$$\beta_f = 0.0065 \frac{\exp(-0.02(V + 30.0))}{\exp(-0.2(V + 30.0)) + 1.0} \tag{A.45}$$

$$\alpha_h = \begin{cases} 0 & V \geq -40 \\ alphah = 0.135 \exp(\frac{-80.0-V}{6.8}); & otherwise \end{cases} \tag{A.46}$$

$$\beta_h = \begin{cases} \frac{1.0}{0.13(1.0+\exp(\frac{-(V+10.66)}{11.1}))} & V \geq -40 \\ 3.56 \exp(0.079V) + (3.1(10.0^{5.0}) \exp(0.35V)) & otherwise \end{cases} \tag{A.47}$$

$$\alpha_j = \begin{cases} 0 & V \geq -40 \\ \frac{(-1.2714(10.0^{5.0}) \exp(0.2444V) - 3.474(10.0^{-5.0}) \exp(-0.04391V))(V+37.78)}{1+\exp(0.311(V+79.23))} & otherwise \end{cases} \tag{A.48}$$

$$\beta_j = \begin{cases} \frac{0.3 \exp(-2.535*(10.0^{-7.0})V)}{1.0+\exp(-0.1(V+32.0))}; & V \geq -40 \\ \frac{0.1212 \exp(-0.01052V)}{1.0+\exp(-0.1378(V+40.14))}; & otherwise \end{cases} \tag{A.49}$$

$$Esi = 7.7 - (13.0287 \log(C_{ai})) \tag{A.50}$$

$$Isi = 0.09d * f(V - Esi) \tag{A.51}$$

$$gK1 = 0.6047\sqrt{\frac{Kpluse}{5.4}} \tag{A.52}$$

$$EK1 = \frac{RT}{F} \log(\frac{Kpluse}{Kplusi}) \tag{A.53}$$

$$EK = \frac{RT}{F} \log(\frac{(Kpluse) + (PR * Napluse)}{Kplusi + (PR * Naplusi)}) \tag{A.54}$$

$$Xi = \begin{cases} 1 & V \leq -100 \\ \frac{2.837(\exp(0.04(V+77.0))-1.0)}{(V+77.0)\exp(0.04(V+35.0))} & otherwise \end{cases} \tag{A.55}$$

$$gK = 0.282\sqrt{\frac{Kpluse}{5.4}} \tag{A.56}$$

$$IK = gK * x * Xi(V - EK) \tag{A.57}$$

$$gamma = \exp(0.06175 * (V - EK1 - 594.31)) \tag{A.58}$$

$$\alpha_{K1} = \frac{1.02}{1 + \exp(0.2385 * (V - EK1 - 59.215))} \tag{A.59}$$

$$\beta_{K1} = \frac{0.49124 \exp(0.08032(V - EK1 + 5.476)) + gamma}{1.0 + \exp(-0.5143(V - EK1 + 4.753))} \tag{A.60}$$

$$K1infinity = \frac{\alpha_{K1}}{\alpha_{K1} + \beta_{K1}} \tag{A.61}$$

$$IK1 = gK1 * K1infinity(V - EK1) \tag{A.62}$$

$$EKp = EK1 \tag{A.63}$$

$$Kp = \frac{1.0}{1.0 + \exp(\frac{7.488 - V}{5.98})} \tag{A.64}$$

$$IKp = gKp * Kp * (V - EKp) \tag{A.65}$$

$$Ib = gb * (V - Eb) \tag{A.66}$$

$$INa = gNa(m^{3.0}) * h * j * (V - ENa) \tag{A.67}$$

$$Iion = INa + Isi + IK + IK1 + IKp + Ib \tag{A.68}$$

$$\frac{dV}{dt} = \frac{-(Iion + I)}{Cm} \tag{A.69}$$

$$\frac{dm}{dt} = \alpha_m(1.0 - m) - \beta_m * m \tag{A.70}$$

$$\frac{dh}{dt} = \alpha_h(1.0 - h) - \beta_h * h \tag{A.71}$$

$$\frac{dj}{dt} = \alpha_j * (1.0 - j) - \beta_j * j \tag{A.72}$$

$$\frac{dd}{dt} = \alpha_d(1.0 - d) - \beta_d * d \tag{A.73}$$

$$\frac{df}{dt} = \alpha_f * (1.0 - f) - \beta_f * f \tag{A.74}$$

$$\frac{dx}{dt} = \alpha_x(1.0 - x) - \beta_x * x \tag{A.75}$$

$$\frac{dC_{ai}}{dt} = -0.0001 * Isi + 0.07 * (0.0001 - C_{ai}) \tag{A.76}$$

## A.3 Fox model

$$I = \begin{cases} -80 & 0 \leq t \leq 1 \\ 0 & otherwise \end{cases} \tag{A.77}$$

$$\frac{dV}{dt} = -(i_{Na}+i_{Ca}+i_{CaK}+i_{Kr}+i_{Ks}+i_{to}+i_{K1}+i_{Kp}+i_{NaCa}+i_{NaK}+i_{pCa}+i_{Nab}+i_{Cab}+I) \tag{A.78}$$

$$E_{Na} = \frac{RT}{F} ln(Na_o/Na_i) \tag{A.79}$$

$$i_{Na} = g_{Na} m^3 hj(V - E_{Na}) \tag{A.80}$$

$$E0_m = V + 47.13 \tag{A.81}$$

$$\alpha_m = \frac{0.32 E0_m}{1 - \exp(-(0.1) * E0_m)} \tag{A.82}$$

$$\beta_m = 0.08 \exp(-(V)/11) \tag{A.83}$$

$$\frac{dm}{dt} = \alpha_m(1 - m) - \beta_m m \tag{A.84}$$

$$\alpha_h = 0.135 \exp((V + 80 - shift_h)/-(6.8)) \tag{A.85}$$

$$\beta_h = \frac{7.5}{1 + \exp(-(0.1)(V + 11 - shift_h))} \tag{A.86}$$

$$\frac{dh}{dt} = \alpha_h(1-h) - \beta_h h \tag{A.87}$$

$$\alpha_j = 0.175\frac{\exp(\frac{V+100-shift_j}{-23})}{1+\exp(0.15(V+79-shift_j))} \tag{A.88}$$

$$\beta_j = 0.31 + \exp(-(0.1)(V+32-shift_j)) \tag{A.89}$$

$$\frac{dj}{dt} = \alpha_j(1-j) - \beta_j j \tag{A.90}$$

$$i_{K1} = \frac{g_{K1}K1_{infinity}K_o}{(K_o+K_{mK1})(V-E_K)} \tag{A.91}$$

$$K1_{infinity} = \frac{1}{2+\exp(\frac{1.62F}{RT}(V-E_K))} \tag{A.92}$$

$$E_K = \frac{RT}{F}ln(\frac{K_o}{K_i}) \tag{A.93}$$

$$R_V = \frac{1}{1+2.5\exp(0.1(V+28))} \tag{A.94}$$

$$i_{Kr} = g_{Kr}R_V X_{kr}\sqrt{K_o/4}(V-E_K) \tag{A.95}$$

$$X_{krinf} = \frac{1}{1+\exp(-(2.182)-0.1819V)} \tag{A.96}$$

$$tau_{Xkr} = 43 + \frac{1}{\exp(-(5.495)+0.1691V)+\exp(-(7.677)-0.0128V)} \tag{A.97}$$

$$\frac{dX_{kr}}{dt} = \frac{X_{krinf} - X_{kr}}{tau_{Xkr}} \tag{A.98}$$

$$E_{Ks} = \frac{RT}{F} ln(\frac{K_o + 0.01833 Na_o}{K_i + 0.01833 Na_i}) \tag{A.99}$$

$$i_{Ks} = g_{Ks} X_{ks}^2 (V - E_{Ks}) \tag{A.100}$$

$$X_{ksinfinity} = \frac{1}{1 + \exp(\frac{V-16}{-13.6})} \tag{A.101}$$

$$tau_{Xks} = \frac{1}{0.0000719 \frac{V-10}{1-\exp(-(0.148)(V-10))} + 0.000131 \frac{V-10}{\exp(0.0687(V-10))-1}} \tag{A.102}$$

$$\frac{dX_{ks}}{dt} = \frac{X_{ksinfinity} - X_{ks}}{tau_{Xks}} \tag{A.103}$$

$$i_{to} = g_{to} X_{to} Y_{to} (V - E_K) \tag{A.104}$$

$$\alpha_{Xto} = 0.04516 \exp(0.03577 V) \tag{A.105}$$

$$\beta_{Xto} = 0.0989 \exp(-0.06237 V) \tag{A.106}$$

$$\frac{dX_{to}}{dt} = \alpha_{Xto}(1 - X_{to}) - beta_{Xto} X_t o \tag{A.107}$$

$$\alpha_{Yto} = \frac{0.005415 \exp(\frac{V+33.5}{-5})}{1 + 0.051335 \exp(\frac{V+33.5}{-5})} \tag{A.108}$$

$$\beta_{Yto} = \frac{0.005415 \exp(\frac{V+33.5}{5})}{1 + 0.051335 \exp(\frac{V+33.5}{5})} \tag{A.109}$$

$$\frac{dY_{to}}{dt} = \alpha_{Yto}(1 - Y_{to}) - \beta_{Yto} Y_{to} \tag{A.110}$$

$$i_{Kp} = g_{Kp} K p_V (V - E_K) \tag{A.111}$$

$$Kp_V = \frac{1}{1 + \exp(\frac{7.488 - V}{5.98})} \tag{A.112}$$

$$f_{NaK} = \frac{1}{1 + 0.1245 \exp(-(0.1) V \frac{F}{RT}) + 0.0365 sigma \exp(\frac{-VF}{RT})} \tag{A.113}$$

$$sigma = \frac{1}{7} * (\exp(\frac{Na_o}{67.3}) - 1) \tag{A.114}$$

$$i_{NaK} = \frac{i_{NaKmax} f_{NaK} (1 + (\frac{K_{mNai}}{Na_i}^{1.5})) K_o}{K_o + K_{mKo}} \tag{A.115}$$

$$\begin{aligned} i_{NaCa} = & \frac{K_{NaCa}}{((K_{mNa}^3) + (Na_o^3))(K_{mCa} + Ca_o)(1 + K_{sat} \exp((eta - 1) V \frac{F}{RT}))} * \\ & (\exp(eta * V \frac{F}{RT})(Na_i^3) Ca_o - \exp((eta - 1) V \frac{F}{RT})(Na_o^3) Ca_i) \end{aligned} \tag{A.116}$$

$$i_{pCa} = \frac{i_{pCamax} Ca_i}{K_{mpCa} + Ca_i} \tag{A.117}$$

$$E_{Ca} = \frac{RT}{2F} ln(\frac{Ca_o}{Ca_i}) \tag{A.118}$$

$$i_{Cab} = g_{Cab}(V - E_{Ca}) \tag{A.119}$$

$$i_{Nab} = g_{Nab}(V - E_{Na}) \tag{A.120}$$

$$i_{Ca} = i_{Camax} f df_{Ca} \tag{A.121}$$

$$i_{Camax} = \frac{\frac{P_{Ca}}{C_{sc}} 4V * F^2}{RT} (Ca_i * \exp(2V\frac{F}{RT}) - 0.341 Ca_o) \tag{A.122}$$

$$i_{CaK} = \frac{\frac{P_{CaK}}{C_{sc}} f * d * \frac{f_{Ca}}{1+\frac{i_{Camax}}{i_{Cahalf}}} 1000V * \frac{F^2}{RT}(K_i * \exp(V\frac{F}{RT}) - K_o)}{\exp(V\frac{F}{RT}) - 1} \tag{A.123}$$

$$f_{infinity} = \frac{1}{1 + \exp(\frac{V+12.5}{5})} \tag{A.124}$$

$$tau_f = 30 + \frac{200}{1 + \exp(\frac{V+20}{9.5})} \tag{A.125}$$

$$\frac{df}{dt} = \frac{f_{infinity} - f}{tau_f} \tag{A.126}$$

$$d_{infinity} = \frac{1}{1 + \exp(\frac{V+10}{-(6.24)})} \tag{A.127}$$

$$E0_m = V + 40 \tag{A.128}$$

$$tau_d = \frac{1}{\frac{0.25\exp(-(0.01)V)}{1+\exp(-(0.07)V)} + \frac{0.07\exp(-(0.05)*E0_m)}{1+\exp(0.05E0_m)}} \tag{A.129}$$

$$\frac{d(d)}{dt} = \frac{d_{infinity} - d}{tau_d} \tag{A.130}$$

$$f_{Cainfinity} = \frac{1}{1 + (\frac{Ca_i}{K_{mfCa}})^3} \tag{A.131}$$

$$tau_{fCa} = 30 \tag{A.132}$$

$$\frac{df_{Ca}}{dt} = \frac{f_{Cainfinity} - f_{Ca}}{tau_{fCa}} \tag{A.133}$$

$$J_{up} = \frac{V_{up}}{1 + (\frac{K_{mup}}{Ca_i})^2} \tag{A.134}$$

$$gamma = \frac{1}{1 + (\frac{2000}{Ca_{SR}})^3} \tag{A.135}$$

$$J_{rel} = \frac{P_{rel} f * d * f_{Ca}(gamma * Ca_{SR} - Ca_i)}{1 + 1.65 \exp(\frac{V}{20})} \tag{A.136}$$

$$J_{leak} = P_{leak}(Ca_{SR} - Ca_i) \tag{A.137}$$

$$\beta_{SR} = \frac{1}{1 + \frac{CSQN_{tot} K_{mCSQN}}{(K_{mCSQN} + Ca_{SR})^2}} \tag{A.138}$$

$$\frac{dCa_{SR}}{dt} = \frac{\beta_{SR}(J_{up} - J_{leak} - J_{rel})V_{myo}}{V_{SR}} \tag{A.139}$$

$$\beta_i = \frac{1}{1 + \frac{CMDN_{tot} * K_{mCMDN}}{(K_{mCMDN} + Ca_i)^2}} \tag{A.140}$$

$$\frac{dCa_i}{dt} = \beta_i(J_{rel}+J_{leak}-J_{up}-(\frac{A_{Cap}*C_{sc}}{2F*V_{myo}}*(i_{Ca}+i_{Cab}+i_{pCa}-2*i_{NaCa})) \quad \text{(A.141)}$$

| variable | value | variable | value | variable | value |
|---|---|---|---|---|---|
| $V$ | -94.7 | $R$ | 8.314 | $T$ | 310 |
| $F$ | 96.5 | $g_{Na}$ | 12.8 | $m$ | 0.00024676 |
| $h$ | 0.99869 | $shift_h$ | 0 | $j$ | 0.99887 |
| $shit_j$ | 0 | $g_{K1}$ | 2.8 | $K_{mK1}$ | 13 |
| $g_{Kr}$ | 0.136 | $X_{Kr}$ | 0.229 | $g_{Ks}$ | 0.0245 |
| $X_{ks}$ | 0.0001 | $g_{to}$ | 0.23815 | $X_{to}$ | 0.00003742 |
| $Y_{to}$ | 1 | $g_{Kp}$ | 0.002216 | $i_{NaKmax}$ | 0.693 |
| $K_{mNai}$ | 10 | $K_{mKo}$ | 1.5 | $K_{mCa}$ | 1380 |
| $K_{mNa}$ | 87.5 | $K_{NaCa}$ | 1500 | $K_{sat}$ | 0.2 |
| $eta$ | 0.35 | $K_{mpCa}$ | 0.05 | $i_pCamax$ | 0.05 |
| $g_{Cab}$ | 0.0003842 | $g_{Nab}$ | 0.0031 | $P_{Ca}$ | 0.0000226 |
| $P_{CaK}$ | 0.000000579 | $i_{Cahalf}$ | -0.265 | $C_{sc}$ | 1 |
| $f$ | 0.983 | $d$ | 0.0001 | $f_{Ca}$ | 0.942 |
| $Ca_i$ | 0.0472 | $K_{mCMDN}$ | 2 | $CMDN_{tot}$ | 10 |
| $V_{myo}$ | 0.00002584 | $A_{Cap}$ | 0.0001534 | $Ca_{SR}$ | 320 |
| $P_{leak}$ | 0.000001 | $K_{mCSQN}$ | 600 | $CSQN_{tot}$ | 10000 |
| $V_{up}$ | 0.1 | $K_{mup}$ | 0.32 | $Na_i$ | 10 |
| $K_i$ | 149.4 | $K_o$ | 4 | $K_{mfCa}$ | 0.18 |
| $P_{rel}$ | 6 | $V_{SR}$ | 0.000002 | $Na_o$ | 138 |
| $Ca_o$ | 2000 | | | | |

## A.4 Courtemanche model

| variable | value | variable | value | variable | value |
|---|---|---|---|---|---|
| $V$ | -81.18 | $R$ | 8.3143 | $T$ | 310.0 |
| $F$ | 96.4867 | $Cm$ | 100.0 | $stim_{start}$ | 50.0 |
| $stim_{duration}$ | 2.0 | $stim_{amplitude}$ | -2000.0 | $g_{Na}$ | 7.8 |
| $m$ | 2.908e-3 | $h$ | 9.649e-1 | $j$ | 9.775e-1 |
| $g_{K1}$ | 0.09 | $K_{Q10}$ | 3.0 | $g_{to}$ | 0.1652 |
| $oa$ | 3.043e-2 | $oi$ | 9.992e-1 | $ua$ | 4.966e-3 |
| $ui$ | 9.986e-1 | $g_{Kr}$ | 0.029411765 | $xr$ | 3.296e-5 |
| $g_{Ks}$ | 0.12941176 | $xs$ | 1.869e-2 | $g_{CaL}$ | 0.12375 |
| $d$ | 1.367e-4 | $f$ | 9.996e-1 | $f_{Ca}$ | 7.755e-1 |
| $Km_{Nai}$ | 10.0 | $Km_{Ko}$ | 1.5 | $i_{NaKmax}$ | 0.59933874 |
| $g_{BNa}$ | 0.0006744375 | $g_{BCa}$ | 0.001131 | $g_{BK}$ | 0.0 |
| $I_{NaCamax}$ | 1600.0 | $K_{mNa}$ | 87.5 | $K_{mCa}$ | 1.38 |
| $K_{sat}$ | 0.1 | $gamma$ | 0.35 | $i_{CaPmax}$ | 0.275 |
| $K_{rel}$ | 30.0 | $u$ | 2.35e-112 | $v$ | 1.0 |
| $w$ | 0.9992 | $tau_{tr}$ | 180.0 | $I_{upmax}$ | 0.005 |
| $K_{up}$ | 0.00092 | $Ca_{upmax}$ | 15.0 | $CMDN_{max}$ | 0.05 |
| $TRPN_{max}$ | 0.07 | $CSQN_{max}$ | 10.0 | $Km_{CMDN}$ | 0.00238 |
| $Km_{TRPN}$ | 0.0005 | $Km_{CSQN}$ | 0.8 | $Na_i$ | 1.117e1 |
| $Ca_i$ | 1.013e-4 | $K_i$ | 1.39e2 | $Ca_{rel}$ | 1.488 |
| $Ca_{up}$ | 1.488 | $V_{cell}$ | 20100.0 | $Na_o$ | 140.0 |
| $Ca_o$ | 1.8 | $K_o$ | 5.4 | | |

$$E_{Na} = \frac{RT}{F} log(\frac{Na_o}{Na_i}) \tag{A.142}$$

$$i_{Na} = Cm * g_{Na} m^3 h * j * (V - E_{Na}) \tag{A.143}$$

$$\alpha_m = \begin{cases} 3.2 & V = -47.13 \\ 0.32\frac{V+47.13}{/}1.0 - exp(-(0.1)(V+47.13)) & otherwise \end{cases} \tag{A.144}$$

$$\beta_m = 0.08\exp(\frac{-(V)}{11.0}) \tag{A.145}$$

$$m_{inf} = \frac{\alpha_m}{\alpha_m + \beta_m} \tag{A.146}$$

$$tau_m = \frac{1.0}{\alpha_m + \beta_m} \tag{A.147}$$

$$\frac{dm}{dt} = \frac{m_{inf} - m}{tau_m} \tag{A.148}$$

$$\alpha_h = \begin{cases} 0.135\exp(\frac{V+80.0}{-(6.8)}) & V < -40 \\ 0 & otherwise \end{cases} \tag{A.149}$$

$$\beta_h = \begin{cases} 3.56\exp(0.079V) + 3.1E5 * \exp(0.35V) & V < -40 \\ \frac{1.0}{0.13(1.0+\exp(\frac{V+10.66}{-11.1}))} & otherwise \end{cases} \tag{A.150}$$

$$h_{inf} = \frac{\alpha_h}{\alpha_h + \beta_h} \tag{A.151}$$

$$tau_h = \frac{1.0}{\alpha_h + \beta_h} \tag{A.152}$$

$$\frac{dh}{dt} = \frac{h_{inf} - h}{tau_h} \tag{A.153}$$

$$\alpha_j = \begin{cases} \frac{(-(1.2714E5)\exp(0.2444*V)-(3.474E-5)\exp(-(0.04391)V))(V+37.78)}{1.0+\exp(0.311(V+79.23))} & V < -40 \\ 0 & otherwise \end{cases} \tag{A.154}$$

$$\beta_j = \begin{cases} 0.1212\frac{\exp(-(0.01052)V)}{1.0+\exp(-(0.1378)(V+40.14))} & V < -40 \\ 0.3\frac{\exp(-(2.535E-7)V)}{1.0+\exp(-(0.1)(V+32.0))} & otherwise \end{cases} \tag{A.155}$$

$$j_{inf} = \frac{\alpha_j}{\alpha_j + \beta_j} \tag{A.156}$$

$$tau_j = \frac{1.0}{\alpha_j + \beta_j} \tag{A.157}$$

$$\frac{dj}{dt} = \frac{j_{inf} - j}{tau_j} \tag{A.158}$$

$$E_K = \frac{RT}{F} log(\frac{K_o}{K_i}) \tag{A.159}$$

$$i_{K1} = Cm * g_{K1} \frac{V - E_K}{1 + \exp(0.07(V + 80.0))} \tag{A.160}$$

$$i_{to} = Cm * g_{to} * oa^{3.0} * oi(V - E_K) \tag{A.161}$$

$$\alpha_{oa} = 0.65(\exp(\frac{V - -(10.0)}{-(8.5)}) + \exp(\frac{V - -(10.0) - 40.0}{-(59.0)}))^{-(1.0)} \tag{A.162}$$

$$\beta_{oa} = 0.65(2.5 + \exp(\frac{V - -(10.0) + 72.0}{17.0}))^{-(1.0)} \tag{A.163}$$

$$tau_{oa} = \frac{(\alpha_{oa} + \beta_{oa})^{-(1)}}{K_{Q10}} \tag{A.164}$$

$$oa_{infinity} = (1.0 + \exp(\frac{V - -(10.0) + 10.47}{-(17.54)})^{-(1.0)} \tag{A.165}$$

$$\frac{doa}{dt} = \frac{oa_{infinity} - oa}{tau_{oa}} \tag{A.166}$$

$$\alpha_{oi} = (18.53 + 1.0 * \exp(\frac{V - -(10.0) + 103.7}{10.95}))^{-(1.0)} \tag{A.167}$$

$$\beta_{oi} = (35.56 + 1.0 * \exp(\frac{V - -(10.0) - 8.74}{-(7.44)}))^{-(1.0)} \tag{A.168}$$

$$tau_{oi} = \frac{(alpha_{oi} + beta_{oi})^{-(1.0)}}{K_{Q10}} \tag{A.169}$$

$$oi_{infinity} = (1.0 + \exp(\frac{V - -(10.0) + 33.1}{5.3}))^{-(1.0)} \tag{A.170}$$

$$\frac{doi}{dt} = \frac{oi_{infinity} - oi}{tau_{oi}} \tag{A.171}$$

$$g_{Kur} = 0.005 + \frac{0.05}{1.0 + \exp(\frac{V-15.0}{-(13.0)})} \tag{A.172}$$

$$i_{Kur} = Cm * g_{Kur} * ua^{3.0} * ui(V - E_K) \tag{A.173}$$

$$\alpha_{ua} = 0.65(\exp(\frac{V - -(10.0)}{-(8.5)}) + \exp(\frac{V - -(10.0) - 40.0}{-(59.0)}))^{-(1.0)} \tag{A.174}$$

$$\beta_{ua} = 0.65(2.5 + \exp(\frac{V - -(10.0) + 72.0}{17.0}))^{-(1.0)} \tag{A.175}$$

$$tau_{ua} = \frac{(alpha_{ua} + beta_{ua})^{-(1.0)}}{K_{Q10}} \tag{A.176}$$

$$ua_{infinity} = (1.0 + \exp(\frac{V - -(10.0) + 20.3}{-(9.6)}))^{-(1.0)} \tag{A.177}$$

$$\frac{dua}{dt} = \frac{ua_{infinity} - ua}{tau_{ua}} \tag{A.178}$$

$$\alpha_{ui} = (21.0 + 1.0 * \exp(\frac{V - -(10.0) - 195.0}{-(28.0)}))^{-(1.0)} \tag{A.179}$$

$$\beta_{ui} = \frac{1.0}{\exp(\frac{V - -(10.0) - 168.0}{-(16.0)})} \tag{A.180}$$

$$tau_{ui} = \frac{(\alpha_{ui} + \beta_{ui})^{-(1.0)}}{K_{Q10}} \tag{A.181}$$

$$ui_{infinity} = (1 + \exp(\frac{V - -(10.0) - 109.45}{27.48}))^{-(1.0)} \tag{A.182}$$

$$\frac{dui}{dt} = \frac{ui_{infinity} - ui}{tau_{ui}} \tag{A.183}$$

$$i_{Kr} = Cm * g_{Kr} * xr * \frac{V - E_K}{1.0 + \exp(\frac{V + 15.0}{22.4})} \tag{A.184}$$

$$\alpha_{xr} = \begin{cases} 0.0015 & |V + 14.1| < 1E - 10 \\ 0.0003 * \frac{V + 14.1}{1.0 - \exp(\frac{V + 14.1}{-(5.0)})} & otherwise \end{cases} \tag{A.185}$$

$$\beta_{xr} = \begin{cases} 3.7836118E-4 & |V-3.3328| < 1E-10 \\ 0.000073898 * \frac{(V-3.3328)}{\exp(\frac{V-3.3328}{5.1237})-1.0} & otherwise \end{cases} \tag{A.186}$$

$$tau_{xr} = (\alpha_{xr} + \beta_{xr})^{-(1.0)} \tag{A.187}$$

$$xr_{infinity} = (1.0 + \exp(\frac{V+14.1}{-(6.5)}))^{-(1.0)} \tag{A.188}$$

$$\frac{dxr}{dt} = \frac{xr_{infinity} - xr}{tau_{xr}} \tag{A.189}$$

$$i_{Ks} = Cm * g_{Ks} * (xs^{2.0}(V - E_K) \tag{A.190}$$

$$\alpha_{xs} = \begin{cases} 0.00068 & |V-19.9| < 1E-10 \\ 0.00004\frac{(V-19.9)}{1.0-\exp(\frac{V-19.9}{-(17.0)})} & otherwise \end{cases} \tag{A.191}$$

$$\beta_{xs} = \begin{cases} 0.000315 & |V-19.9| < 1E-10 \\ 0.000035\frac{(V-19.9)}{\exp(\frac{V-19.9}{9.0})-1.0} & otherwise \end{cases} \tag{A.192}$$

$$tau_{xs} = 0.5(\alpha_{xs} + \beta_{xs})^{-(1.0)} \tag{A.193}$$

$$xs_{infinity} = (1.0 + \exp(\frac{V-19.9}{-(12.7)}))^{-(0.5)} \tag{A.194}$$

$$\frac{dxs}{dt} = \frac{xs_{infinity} - xs}{tau_{xs}} \tag{A.195}$$

$$i_{CaL} = Cm * g_{CaL} * d * f * f_{Ca}(V - 65.0) \tag{A.196}$$

$$d_{infinity} = (1.0 + \exp(\frac{V + 10.0}{-(8.0)}))^{-(1.0)} \tag{A.197}$$

$$tau_d = \begin{cases} \frac{4.579}{1.0+\exp(\frac{V+10.0}{-(6.24)})} & |V + 10| < 1E - 10 \\ \frac{1.0-\exp(\frac{V+10.0}{-(6.24)})}{0.035(V+10.0)(1.0+\exp(\frac{V+10.0}{-(6.24)}))} & otherwise \end{cases} \tag{A.198}$$

$$\frac{dd}{dt} = \frac{d_{infinity} - d}{tau_d} \tag{A.199}$$

$$f_{infinity} = \frac{\exp(\frac{-(V+28.0)}{6.9})}{1.0 + \exp(\frac{-(V+28.0)}{6.9})} \tag{A.200}$$

$$tau_f = 9.0(0.0197 * \exp(-(0.0337^{2.0})(V + 10)^{2.0}) + 0.02)^{-(1.0)} \tag{A.201}$$

$$\frac{df}{dt} = \frac{f_{infinity} - f}{tau_f} \tag{A.202}$$

$$f_{Cainfinity} = (1.0 + \frac{Ca_i}{0.00035})^{-(1.0)} \tag{A.203}$$

$$tau_{fCa} = 2.0; \tag{A.204}$$

$$\frac{df_{Ca}}{dt} = \frac{f_{Cainfinity} - f_{Ca}}{tau_{fCa}} \tag{A.205}$$

$$sigma = \frac{1.0}{7.0} * (\exp(\frac{Na_o}{67.3}) - 1.0) \tag{A.206}$$

$$f_{NaK} = (1.0 + 0.1245\exp(\frac{-(0.1)FV}{RT}) + 0.0365sigma * \exp(\frac{-(F)V}{RT}))^{-(1.0)} \tag{A.207}$$

$$i_{NaK} = Cm * i_{NaKmax} * f_{NaK} * \frac{1.0}{1.0 + (\frac{Km_N a_i}{Na_i})^1.5} * \frac{K_o}{K_o + Km_{Ko}} \tag{A.208}$$

$$E_{Ca} = \frac{RT}{2.0F} * log(\frac{Ca_o}{Ca_i}) \tag{A.209}$$

$$i_{BNa} = Cm * g_{BNa} * (V - E_{Na}) \tag{A.210}$$

$$i_{BCa} = Cm * g_{BCa}(V - E_{Ca}) \tag{A.211}$$

$$i_{BK} = Cm * g_{BK}(V - E_K) \tag{A.212}$$

$$i_{NaCa} = Cm * I_{NaCamax} \times \frac{\exp(\frac{gamma*F*V}{RT})(Na_i^{3.0})Ca_o - \exp(\frac{(gamma-1.0)*F*V}{(RT)})(Na_o^{3.0})Ca_i}{(K_{mNa}^{3.0} + Na_o^{3.0}) * (K_{mCa} + Ca_o) * (1.0 + K_sat\exp(\frac{(gamma-1.0)V*F}{RT}))} \tag{A.213}$$

$$i_{CaP} = Cm * i_{CaPmax}\frac{Ca_i}{0.0005 + Ca_i} \tag{A.214}$$

$$V_{rel} = 0.0048V_{cell} \tag{A.215}$$

$$i_{rel} = K_{rel}(u^{2.0}) * v * w * (Ca_{rel} - Ca_i) \tag{A.216}$$

$$Fn = 1.0E3(1.0E-15 * V_{rel} * i_{rel} - \frac{1.0E-15}{2.0F} * (0.5i_{CaL} - 0.2i_{NaCa})) \quad \text{(A.217)}$$

$$tau_u = 8.0 \quad \text{(A.218)}$$

$$u_{infinity} = (1.0 + \exp(\frac{-(Fn - 3.4175E-13)}{13.67E-16}))^{-(1.0)} \quad \text{(A.219)}$$

$$\frac{du}{dt} = \frac{u_{infinity} - u}{tau_u} \quad \text{(A.220)}$$

$$tau_v = 1.91 + 2.09(1.0 + \exp(-\frac{Fn - 3.4175E-13}{13.67E-16}))^{-(1.0)} \quad \text{(A.221)}$$

$$v_{infinity} = 1.0 - (1.0 + \exp(-\frac{(Fn - 6.835E-14)}{13.67E-16}))^{-(1.0)} \quad \text{(A.222)}$$

$$\frac{dv}{dt} = \frac{v_{infinity} - v}{tau_v} \quad \text{(A.223)}$$

$$tau_w = \begin{cases} \frac{6.0*0.2}{1.3} & |V - 7.9| < 1E-10 \\ 6.0 * \frac{(1.0-\exp(\frac{-(V-7.9)}{5.0}))}{(1.0+0.3\exp(\frac{-(V-7.9)}{5.0}))1.0(V-7.9)} & otherwise \end{cases} \quad \text{(A.224)}$$

$$w_{infinity} = 1.0 - (1.0 + \exp(\frac{-(V - 40.0)}{17.0}))^{-(1.0)} \quad \text{(A.225)}$$

$$\frac{dw}{dt} = \frac{w_{infinity} - w}{tau_w} \quad \text{(A.226)}$$

$$i_{tr} = \frac{Ca_{up} - Ca_{rel}}{tau_{tr}} \tag{A.227}$$

$$i_{up} = \frac{I_{upmax}}{1 + \frac{K_{up}}{Ca_i}} \tag{A.228}$$

$$i_{upleak} = I_{upmax} * \frac{Ca_{up}}{Ca_{upmax}} \tag{A.229}$$

$$Ca_{CMDN} = CMDN_{max} * \frac{Ca_i}{Ca_i + Km_{CMDN}} \tag{A.230}$$

$$Ca_{TRPN} = TRPN_{max} * \frac{Ca_i}{Ca_i + Km_{TRPN}} \tag{A.231}$$

$$Ca_{CSQN} = CSQN_{max} * \frac{Ca_{rel}}{Ca_{rel} + Km_{CSQN}} \tag{A.232}$$

$$V_i = V_{cell} * 0.68 \tag{A.233}$$

$$V_{up} = 0.0552 V_cell \tag{A.234}$$

$$\frac{dNa_i}{dt} = \frac{-(3.0) * i_{NaK} - (3.0 * i_{NaCa} + i_{BNa} + i_{Na})}{V_i F} \tag{A.235}$$

$$\frac{dK_i}{dt} = \frac{2.0 i_{NaK} - (i_{K1} + i_{to} + i_{Kur} + i_{Kr} + i_{Ks} + i_{BK})}{V_i * F} \tag{A.236}$$

$$B1 = \frac{2.0 i_{NaCa} - (i_{CaP} + i_{CaL} + i_{BCa})}{2.0 V_i * F} + \frac{V_{up} * (i_{upleak} - i_{up}) + i_{rel} * V_{rel}}{V_i} \tag{A.237}$$

$$B2 = 1.0 + TRPN_{max} * \frac{Km_{TRPN}}{(Ca_i + Km_{TRPN})^{2.0}} + CMDN_{max} * \frac{Km_{CMDN}}{(Ca_i + Km_{CMDN})^{2.0}} \tag{A.238}$$

$$\frac{dCa_i}{dt} = \frac{B1}{B2} \tag{A.239}$$

$$\frac{dCa_{up}}{dt} = i_{up} - (i_{upleak} + i_{tr} * \frac{V_{rel}}{V_{up}}) \tag{A.240}$$

$$\frac{dCa_{rel}}{dt} = (i_{tr} - i_{rel})(1.0 + CSQN_{max} * \frac{Km_{CSQN}}{(Ca_{rel} + Km_{CSQN})^{2.0}})^{-(1.0)} \tag{A.241}$$

$$i_{st} = \begin{cases} stim_{amplitude} & stim_{start} \leq t \leq stim_{start} + stim_{duration} \\ 0 & otherwise \end{cases} \tag{A.242}$$

$$I_{ion} = i_{Na} + i_{K1} + i_{to} + i_{Kur} + i_{Kr} + i_{Ks} + i_{BNa} + i_{BCa} + i_{NaK} + i_{CaP} + i_{NaCa} + i_{CaL} \tag{A.243}$$

$$\frac{dV}{dt} = \frac{-(I_{ion} + i_{st})}{Cm} \tag{A.244}$$

## A.5 Tusscher model

| variable | value | variable | value | variable | value |
|---|---|---|---|---|---|
| $V$ | -85.423 | $R$ | 8314.472 | $T$ | 310.0 |
| $F$ | 96485.3415 | $Cm$ | 0.185 | $V_c$ | 0.016404 |
| $stim_{start}$ | 10.0 | $stim_{duration}$ | 1.0 | $stim_{amplitude}$ | 52.0 |
| $stim_{amplitude}$ | 0 | $P_{kna}$ | 0.03 | $g_{K1}$ | 5.405 |
| $g_{Kr}$ | 0.153 | $Xr1$ | 0.0165 | $Xr2$ | 0.473 |
| $g_{Ks}$ | 0.098 | $Xs$ | 0.0174 | $g_{Na}$ | 14.838 |
| $m$ | 0.00165 | $h$ | 0.749 | $j$ | 0.6788 |
| $g_{bna}$ | 0.00029 | $g_{CaL}$ | 0.0000398 | $d$ | 3.288e-5 |
| $f$ | 0.7026 | $f2$ | 0.9526 | $fCass$ | 0.9942 |
| $g_{bca}$ | 0.000592 | $g_{to}$ | 0.294 | $s$ | 0.999998 |
| $r$ | 2.347e-8 | $P_{NaK}$ | 2.724 | $K_{mk}$ | 1.0 |
| $K_{mNa}$ | 40.0 | $K_{NaCa}$ | 1000.0 | $K_{sat}$ | 0.1 |
| $\alpha$ | 2.5 | $gamma$ | 0.35 | $Km_{Ca}$ | 1.38 |
| $Km_{Nai}$ | 87.5 | $g_{pCa}$ | 0.1238 | $K_{pCa}$ | 0.0005 |
| $g_{pK}$ | 0.0146 | $Ca_o$ | 2.0 | $R_{prime}$ | 0.8978 |
| $Ca_i$ | 0.000153 | $Ca_{SR}$ | 4.272 | $Ca_{ss}$ | 0.00042 |
| $k1_{prime}$ | 0.15 | $k2_{prime}$ | 0.045 | $k3$ | 0.06 |
| $k4$ | 0.005 | $EC$ | 1.5 | $max_{sr}$ | 2.5 |
| $min_{sr}$ | 1.0 | $V_{rel}$ | 0.102 | $V_{xfer}$ | 0.0038 |
| $K_{up}$ | 0.00025 | $V_{leak}$ | 0.00036 | $Vmax_{up}$ | 0.006375 |
| $Buf_c$ | 0.2 | $K_{bufc}$ | 0.001 | $Buf_{sr}$ | 10.0 |
| $K_{bufsr}$ | 0.3 | $Buf_{ss}$ | 0.4 | $K_{bufss}$ | 0.00025 |
| $V_{sr}$ | 0.001094 | $V_{ss}$ | 0.00005468 | $Na_i$ | 10.132 |
| $Na_o$ | 140.0 | $K_i$ | 138.52 | $K_o$ | 5.4 |

$$i_{Stim} = \begin{cases} -stim_{amplitude} & stim_{start} \leq t \leq stim_{start} + stim_{duration} \\ 0 & otherwise \end{cases} \quad \text{(A.245)}$$

$$E_{Na} = \frac{RT}{F} log(\frac{Na_o}{Na_i}) \tag{A.246}$$

$$E_K = \frac{RT}{F} log(\frac{K_o}{K_i}) \tag{A.247}$$

$$E_{Ks} = \frac{RT}{F} log(\frac{K_o + P_{kna} * Na_o}{K_i + P_{kna} * Na_i}) \tag{A.248}$$

$$E_{Ca} = \frac{0.5 * RT}{F} log(\frac{Ca_o}{Ca_i}) \tag{A.249}$$

$$\alpha_{K1} = \frac{0.1}{1.0 + \exp(0.06(V - E_K - 200.0))} \tag{A.250}$$

$$\beta_{K1} = \frac{3.0 \exp(0.0002(V - E_K + 100.0)) + \exp(0.1(V - E_K - 10.0))}{1.0 + \exp(-(0.5)(V - E_K))} \tag{A.251}$$

$$xK1_{inf} = \frac{\alpha_{K1}}{\alpha_{K1} + \beta_{K1}} \tag{A.252}$$

$$i_{K1} = g_{K1} * xK1_{inf} \sqrt{\frac{K_o}{5.4}} (V - E_K) \tag{A.253}$$

$$i_{Kr} = g_{Kr} \sqrt{\frac{K_o}{5.4}} * Xr1 * Xr2(V - E_K) \tag{A.254}$$

$$xr1_{inf} = \frac{1.0}{1.0 + \exp(\frac{-(26.0) - V}{7.0})} \tag{A.255}$$

$$\alpha_{xr1} = \frac{450.0}{1.0 + \exp(\frac{-(45.0) - V}{10.0})} \tag{A.256}$$

$$\beta_{xr1} = \frac{6.0}{1.0 + \exp(\frac{V+30.0}{11.5})} \tag{A.257}$$

$$tau_{xr1} = \alpha_{xr1} * \beta_{xr1} \tag{A.258}$$

$$\frac{dXr1}{dt} = \frac{xr1_{inf} - Xr1}{tau_{xr1}} \tag{A.259}$$

$$xr2_{inf} = \frac{1.0}{1.0 + \exp(\frac{V+88.0}{24.0})} \tag{A.260}$$

$$\alpha_{xr2} = \frac{3.0}{1.0 + \exp(\frac{-(60.0)-V}{20.0})} \tag{A.261}$$

$$\beta_{xr2} = \frac{1.12}{1 + \exp(\frac{V-60.0}{20.0})} \tag{A.262}$$

$$tau_{xr2} = \alpha_{xr2} * \beta_{xr2} \tag{A.263}$$

$$\frac{dXr2}{dt} = \frac{xr2_{inf} - Xr2}{tau_{xr2}} \tag{A.264}$$

$$i_{Ks} = g_{Ks}(Xs^{2.0})(V - E_{Ks}) \tag{A.265}$$

$$xs_{inf} = \frac{1.0}{1.0 + \exp(\frac{-(5.0)-V}{14.0})} \tag{A.266}$$

$$\alpha_{xs} = \frac{1400.0}{\sqrt{1.0 + \exp(\frac{5.0-V}{6.0})}} \tag{A.267}$$

$$\beta_{xs} = \frac{1.0}{1.0 + \exp(\frac{V-35.0}{15.0})} \tag{A.268}$$

$$tau_{xs} = \alpha_{xs} * \beta_{xs} + 80.0 \tag{A.269}$$

$$\frac{dXs}{dt} = \frac{xs_{inf} - Xs}{tau_{xs}} \tag{A.270}$$

$$i_{Na} = g_{Na}(m^{3.0})h * j(V - E_{Na}) \tag{A.271}$$

$$m_{inf} = \frac{1.0}{(1.0 + \exp(\frac{-(56.86)-V}{9.03}))^{2.0}} \tag{A.272}$$

$$\alpha_m = \frac{1.0}{1.0 + \exp(\frac{-(60.0)-V}{5.0})} \tag{A.273}$$

$$\beta_m = \frac{0.1}{1.0 + \exp(\frac{V+35.0}{5.0})} + \frac{0.1}{1.0 + \exp(\frac{V-50.0}{200.0})} \tag{A.274}$$

$$tau_m = \alpha_m * \beta_m \tag{A.275}$$

$$\frac{dm}{dt} = \frac{m_{inf} - m}{tau_m} \tag{A.276}$$

$$h_{inf} = \frac{1.0}{(1.0 + \exp(\frac{V+71.55}{7.43}))^{2.0}} \tag{A.277}$$

$$\alpha_h = \begin{cases} 0.057 \exp(\frac{-(V+80.0)}{6.8}) & V < -40 \\ 0 & otherwise \end{cases} \tag{A.278}$$

$$\beta_h = \begin{cases} 2.7 \exp(0.079V) + 310000.0 \exp(0.3485V) & V < -40 \\ \frac{0.77}{0.13(1.0+\exp(\frac{V+10.66}{-(11.1)}))} & otherwise \end{cases} \tag{A.279}$$

$$tau_h = \frac{1.0}{\alpha_h + \beta_h} \tag{A.280}$$

$$\frac{dh}{dt} = \frac{h_{inf} - h}{tau_h} \tag{A.281}$$

$$j_{inf} = \frac{1.0}{(1.0 + \exp(\frac{V+71.55}{7.43}))^{2.0}} \tag{A.282}$$

$$\alpha_j = \begin{cases} \frac{(-(25428.0)\exp(0.2444V) - 6.948E-6*\exp(-(0.04391)V))(V+37.78)}{1.0+\exp(0.311(V+79.23))} & V < -40 \\ 0 & otherwise \end{cases} \tag{A.283}$$

$$\beta_j = \begin{cases} 0.02424\frac{\exp(-(0.01052)*V)}{1.0+\exp(-(0.1378)(V+40.14))} & V < -40 \\ 0.6\frac{\exp(0.057V)}{1.0+\exp(-(0.1)(V+32.0))} & otherwise \end{cases} \tag{A.284}$$

$$tau_j = \frac{1.0}{\alpha_j + \beta_j} \tag{A.285}$$

$$\frac{dj}{dt} = \frac{j_{inf} - j}{tau_j} \tag{A.286}$$

$$i_{bNa} = g_{bna} * (V - E_{Na}) \tag{A.287}$$

$$\begin{aligned} i_{CaL} = & g_{CaL} * d * f * f2 * fCass * 4.0(V - 15.0)\frac{(F^{2.0})}{RT} \times \\ & \frac{0.25 * Ca_s s * \exp(2.0 * (V - 15.0) * \frac{F}{RT}) - Ca_o}{\exp(2.0 * (V - 15.0) * \frac{F}{RT}) - 1.0} \end{aligned} \tag{A.288}$$

$$d_{inf} = \frac{1.0}{1.0 + \exp(\frac{-(8.0)-V}{7.5})} \tag{A.289}$$

$$\alpha_d = \frac{1.4}{1.0 + \exp(\frac{-(35.0)-V}{13.0})} + 0.25 \tag{A.290}$$

$$\beta_d = \frac{1.4}{1.0 + \exp(\frac{V+5.0}{5.0})} \tag{A.291}$$

$$gamma_d = \frac{1.0}{1.0 + \exp(\frac{50.0-V}{20.0})} \tag{A.292}$$

$$tau_d = \alpha_d * \beta_d + gamma_d \tag{A.293}$$

$$\frac{dd}{dt} = \frac{d_{inf} - d}{tau_d} \tag{A.294}$$

$$f_{inf} = \frac{1.0}{1.0 + \exp(\frac{V+20.0}{7.0})} \tag{A.295}$$

$$tau_f = 1102.5 \exp(\frac{-((V + 27.0)^{2.0})}{225.0}) + \frac{200.0}{1.0 + \exp(\frac{13.0-V}{10.0})} + \frac{180.0}{1.0 + \exp(\frac{V+30.0}{10.0})} + 20.0 \tag{A.296}$$

$$\frac{df}{dt} = \frac{f_{inf} - f}{tau_f} \tag{A.297}$$

$$f2_{inf} = \frac{0.67}{1.0 + \exp(\frac{V+35.0}{7.0})} + 0.33 \tag{A.298}$$

$$tau_{f2} = 562.0 \exp(\frac{-((V + 27.0)^{2.0})}{240.0}) + \frac{31.0}{1.0 + \exp(\frac{25.0-V}{10.0})} + \frac{80.0}{1.0 + \exp(\frac{V+30.0}{10.0})} \tag{A.299}$$

$$\frac{df2}{dt} = \frac{f2_{inf} - f2}{tau_{f2}} \tag{A.300}$$

$$fCass_{inf} = \frac{0.6}{1.0 + (\frac{Ca_{ss}}{0.05})^{2.0}} + 0.4 \tag{A.301}$$

$$tau_{fCass} = \frac{80.0}{1.0 + (\frac{Ca_{ss}}{0.05})^{2.0}} + 2.0 \tag{A.302}$$

$$\frac{dfCass}{dt} = \frac{fCass_{inf} - fCass}{tau_{fCass}} \tag{A.303}$$

$$i_{bCa} = g_{bca} * (V - E_{Ca}) \tag{A.304}$$

$$i_{to} = g_{to} * r * s(V - E_K) \tag{A.305}$$

$$s_{inf} = \frac{1.0}{1.0 + \exp(\frac{V+20}{5.0})} \tag{A.306}$$

$$tau_s = 85.0 \exp(\frac{-((V + 45.0)^{2.0})}{320.0}) + \frac{5.0}{1.0 + \exp(\frac{V-20.0}{5.0})} + 3.0 \tag{A.307}$$

$$\frac{ds}{dt} = \frac{s_{inf} - s}{tau_s} \tag{A.308}$$

$$r_{inf} = \frac{1.0}{1.0 + \exp(\frac{20.0-V}{6.0})} \tag{A.309}$$

$$tau_r = 9.5 \exp(\frac{-((V + 40.0)^{2.0})}{1800.0}) + 0.8 \tag{A.310}$$

$$\frac{dr}{dt} = \frac{r_{inf} - r}{tau_r} \tag{A.311}$$

$$i_{NaK} = P_{NaK} * \frac{K_o}{K_o + K_{mk}} * \frac{Na_i}{Na_i + K_{mNa}} \times \frac{1}{1.0 + 0.1245 \exp(-(0.1) * V * \frac{F}{RT}) + 0.0353 \exp(\frac{-(V)F}{RT})} \tag{A.312}$$

$$i_{NaCa} = K_{NaCa} \times \frac{(\exp(\frac{gamma * V * F}{RT}) * (Na_i^{3.0}) * Ca_o - \exp(\frac{(gamma - 1.0) * V * F}{RT})(Na_o^{3.0}) * Ca_i * \alpha)}{((Km_{Nai}^{3.0}) + (Na_o^{3.0})) * (Km_{Ca} + Ca_o) * (1.0 + K_{sat} * \exp(\frac{(gamma - 1.0) * V * F}{RT}))} \tag{A.313}$$

$$i_{pCa} = g_{pCa} * \frac{Ca_i}{Ca_i + K_{pCa}} \tag{A.314}$$

$$i_{pK} = g_{pK} * \frac{(V - E_K)}{1.0 + \exp(\frac{25.0 - V}{5.98})} \tag{A.315}$$

$$i_{up} = \frac{Vmax_{up}}{1.0 + \frac{(K_{up})^{2.0}}{(Ca_i)^{2.0}}} \tag{A.316}$$

$$i_{leak} = V_{leak} * (Ca_{SR} - Ca_i) \tag{A.317}$$

$$i_{xfer} = V_{xfer} * (Ca_{ss} - Ca_i) \tag{A.318}$$

$$kcasr = max_{sr} - \frac{max_{sr} - min_{sr}}{1.0 + (\frac{EC}{Ca_{SR}})^{2.0}} \tag{A.319}$$

$$k1 = \frac{k1_{prime}}{kcasr} \tag{A.320}$$

$$k2 = k2_{prime} * kcasr \tag{A.321}$$

$$\frac{dR_{prime}}{dt} = -(k2) * Ca_{ss} * R_{prime} + k4 * (1.0 - R_{prime}) \tag{A.322}$$

$$O = k1 * (Ca_{ss}^{2.0}) * \frac{R_{prime}}{k3 + k1 * (Ca_{ss}^{2.0})} \tag{A.323}$$

$$i_{rel} = V_{rel} * O * (Ca_{SR} - Ca_{ss}) \tag{A.324}$$

$$Ca_{ibufc} = \frac{1.0}{1.0 + Buf_c * \frac{K_{bufc}}{(Ca_i + K_{bufc})^{2.0}}} \tag{A.325}$$

$$Ca_{srbufsr} = \frac{1.0}{1.0 + Buf_{sr} * \frac{K_{bufsr}}{(Ca_{SR} + K_{bufsr})^{2.0}}} \tag{A.326}$$

$$Ca_{ssbufss} = \frac{1.0}{1.0 + Buf_{ss} * \frac{K_{bufss}}{(Ca_{ss} + K_{bufss})^{2.0}}} \tag{A.327}$$

$$\frac{dCa_i}{dt} = Ca_{ibufc} \times (\frac{(i_{leak} - i_{up}) * V_{sr}}{V_c} + i_{xfer} - 1.0 * (i_{bCa} + i_{pCa} - 2.0 * i_{NaCa}) * \frac{Cm}{2.0 * 1.0 * V_c * F}) \tag{A.328}$$

$$\frac{dCa_{SR}}{dt} = Ca_{srbufsr} * (i_{up} - (i_{rel} + i_{leak})) \tag{A.329}$$

$$\frac{dCa_{ss}}{dt} = Ca_{ssbufss} * (-(1.0) * i_{CaL} * \frac{Cm}{2.0 * 1.0 * V_{ss} * F} + i_{rel} * \frac{V_{sr}}{V_{ss}} - i_{xfer} * \frac{V_c}{V_{ss}}) \tag{A.330}$$

$$\frac{dNa_i}{dt} = -(1.0) * \frac{(i_{Na} + i_{bNa} + 3.0 * i_{NaK} + 3.0 * i_{NaCa})}{(1 * V_c * F) * Cm} \tag{A.331}$$

$$\frac{dK_i}{dt} = -(1.0) * \frac{(i_{K1} + i_{to} + i_{Kr} + i_{Ks} + i_{pK} + i_{Stim} - 2.0 * i_{NaK})}{(1.0 * V_c * F) * Cm} \tag{A.332}$$

$$I_{ion} = i_{K1} + i_{to} + i_{Kr} + i_{Ks} + i_{CaL} + i_{NaK} + i_{Na} + i_{bNa} + i_{NaCa} + i_{bCa} + i_{pK} + i_{pCa} \tag{A.333}$$

$$\frac{dV}{dt} = -(I_{ion} + i_{Stim}) \tag{A.334}$$