Augmenting Low-Fidelity Electrophysiological Cardiac Cell Models with Neural Networks

Laura Kraft

Supervisor: Dr Christopher Marcotte

Computer Science Department

Durham University

Abstract

Cardiac alternans, a beat-to-beat variation in action potential (AP) duration or amplitude, is a precursor to life-threatening arrhythmias. Computational models such as Mitchell–Schaeffer (MS) and Iyer–Mazhari–Winslow (IMW) simulate cardiac electrophysiology numerically at different levels of fidelity. While the IMW model reproduces detailed underlying ion-channel dynamics, the simplified MS model prioritises speed over realism. This work explores augmenting the MS model with neural networks (NNs) to improve fidelity without introducing full biophysical complexity.

The MS model was implemented in Julia, and the IMW structure imported from the CellML library. Both were solved using adaptive solvers across a range of Basic Cycle Lengths (BCLs), generating AP data for comparison. AP characteristics were analysed and morphology plots created. The MS model was tuned to approximate IMW behaviour. NNs were trained to predict the subsequent AP directly from short histories, bypassing Ordinary Differential Equation integration and to augment the MS model to emulate the IMW model.

Both models reproduced alternans in AP duration (APD) and amplitude (APA) under appropriate stimulus forcing. The IMW model generated physiologically realistic AP morphologies, including the phase-1 notch and prolonged plateau. Tuning timescales brought the MS model APD in line with the IMW model. APD, APA, and Diastolic Interval maps offered insights into discrepancies between the different models. Training NNs enabled robust prediction of APD and APA across BCLs. By contrast, the NN-augmented MS model predictions were less precise. The relative decrease in loss did not fully translate into accurate repolarisation dynamics, suggesting that the backpropagation error introduces problems that require further investigation. However, there was strong agreement with IMW depolarisation and early plateau phases.

These results suggest that NN-augmented low-fidelity models may provide biophysical simulation surrogates, provided augmentation targets early AP phases rather than slow repolarisation, thereby informing future studies of cardiac restitution and arrhythmogenesis.

Table of Contents

1 Introduction

Cardiac arrhythmias including too fast, too slow or irregular heartbeats can lead to heart failure, sudden cardiac arrest, and even death (Kulkarni et al., 2019; Fu, 2015). Electrical cardiac alternans is a beat-to-beat variation in the amplitude or duration of the action potential (AP) of the muscle cells of the heart. It can be observed in patients with cardiac arrhythmias and may be a precursor of life-threatening arrythmias, such as ventricular fibrillation (Pigozzo et al., 2023). These changes, seen at cell level, can also be observed in patients as variations in the shape of the curve of the electrocardiogram (ECG) (Cutler and Rosenbaum, 2009).

Mathematical cardiac electrical excitation models describe the electrical activity of the heart at cellular level. They play a crucial role in understanding its electrophysiological behaviour and form the foundation of computational cardiac medicine (Ten Tusscher et al., 2006; Luo and Rudy, 1991). Numerical models make assumptions about observable biological phenomena and the associations they represent. The models describe variables associated with cell membrane potential and ionic concentrations as a set of ordinary differential equations (ODEs). The Iyer-Mazhari-Winslow (IMW) model uses 67 variables (Iyer et al., 2004). Simpler models such as the Mitchell-Schaefer (MS) model improve computational tractability but show limitations in capturing complex dynamics (Mitchell, 2003). Some of these models including the MS and IMW models can reproduce cardiac alternans behaviour in myocytes. For their importance in predicting cardiac arrhythmias, these models should be studied further (Pigozzo et al., 2023).

Recent advancements in artificial intelligence (AI), particularly neural networks (NNs), offer opportunities to refine these models. Machine learning (ML) models use a larger set of parameters to model relationships between in and outputs statistically. NNs do not contain primary information of underlying physical or mathematical laws governing the system. Using simpler models of electrophysiology in tandem with NNs may allow the reproduction of sophisticated model dynamics with computational costs only marginally above those of simple models. In the future, this approach could help scale these models up to support the development of whole organ models and may help to simulate how the electrical behaviour of the heart is influenced by pharmaceuticals.

2 Aims and Problem Statement

This research aims to:

1. Explore existing simple and complex models of cardiac electrical excitation.
   1.1. Investigate the MS and IMW models by solving their functions in Julia while ensuring discontinuities are handled and graphing the results.
   1.2. Explore MS model parameters to inform tuning.
   1.3. Tune the MS model time parameters and gating variable to better approximate the IMW model.
2. Explore AP characteristics.
   2.1. Define Action Potential Duration (APD), Action Potential Activation (APA) and Diastolic Interval (DI) parameters for different Basic Cycle Lengths (BCLs) for all three models and store these values.
   2.2. Compare mMS to IMW and MS by graphing APD/APA/DI maps.
   2.3. Use the Lux framework in Julia to generate NNs with the aim of predicting the next APA and APD for the next BCL.
   2.4. Train these NNs for different threshold voltages to explore the APD and APA for all three models and graph the trained NNs next predicted APD and APA.
3. Augment a simple numerical model of cardiac electrical excitation with a NN to better approximate a complex model.
   3.1. Add and train a NN to represent an extra current to the two current MS model to approximate the IMW model.
   3.2. Evaluate and graph the NN augmented MS system versus the IMW model to see if

$$\frac{d\bar{u}}{dt} = \bar{F}_{MS}(t, \bar{u}) + I_{NN}(\bar{u}), \qquad \bar{u} \in \mathbb{R}^2$$

(where $\bar{F}_{MS}()$ represents the MS model, $\bar{u}$ represents the MS model's variables, t represents time, and $I_{NN}()$ represents the NN current)

can approximate

$$\frac{d\bar{y}}{dt} = \bar{F}_{IMW}(t, \bar{y}), \qquad \bar{y} \in \mathbb{R}^{67}$$

(where $\bar{F}_{IMW}()$ represents the IMW model, $\bar{y}$ represents the IMW model's variables and t represents time).

3 Background

3.1 Cardiac Electrophysiology

3.1.1 Anatomy

The heart sits in the thoracic cavity between the lungs and consists of four chambers. It is a self-adjusting double-pump that propels blood to the lungs in a low-pressure system and to the body in a high-pressure system (Anatomy of the heart, 2012). The heart is made up of three heart cell types: muscle cells or myocytes, electrical cells and fibre cells (Sampson and McGarth, 2015). Myocytes are contractible cells that when working together pump the blood around the body. Electrical processes on the cell wall of the myocyte govern this action.

3.1.2 Cell Membrane Action Potential

Different concentrations of electrolytes inside and outside of a cell membrane create an electrical potential across the membrane. The cell membrane is semi-permeable to charged ions, such as sodium or potassium, which can move through ion channels or pumps embedded in the membrane (Jost, 2010).

When an electrical stimulus arrives at a resting myocyte, sodium channels open to allow sodium ions to move into the cell along its concentration gradient. The cell membrane potential rises rapidly from its negative resting potential (-90mV) to positive in a process called depolarisation (Fig. 1). After completion of depolarisation, calcium enters the cell to stabilise the electric charge across the cell membrane and to close the sodium channels. Calcium moves slowly creating an electrical voltage plateau phase (Jost, 2010). Calcium is also released from stores inside the cell which triggers contraction (Sampson and McGarth, 2015). When contraction is completed, calcium channels close and potassium channels open allowing potassium to leave the cell in a process called repolarisation. This process of depolarisation and repolarisation is called an AP. Once one myocyte depolarises the AP spreads quickly from heart cell to heart cell in a depolarisation wave across the heart (Sampson and McGarth, 2015).

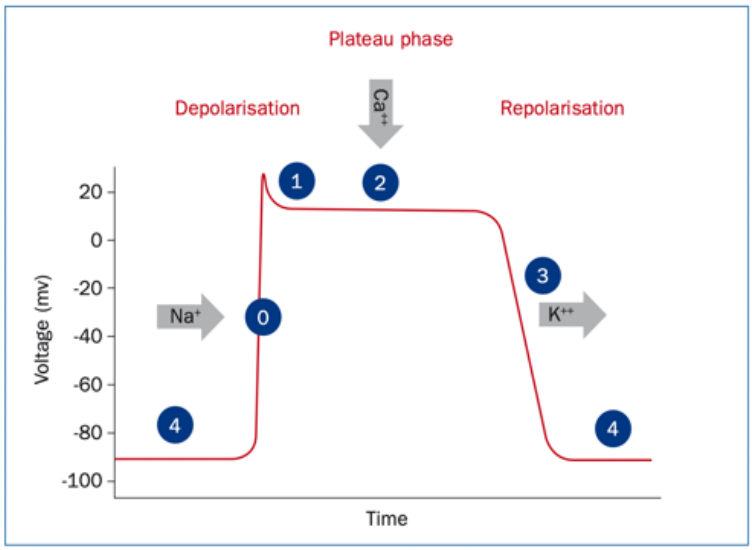

Fig. 1. AP of a myocyte: initial rapid depolarisation due to sodium influx (phase 0); stabilisation period due to initial calcium influx (phase 1); plateau phase characterised by slow calcium influx leading to contraction of the heart (phase 2); repolarisation due to outflow of potassium (phase 3); resting (phase 4) (Sampson and McGarth, 2015).

After depolarisation the cell becomes refractory for a brief time to avoid early re-stimulation. Initially, there is an absolute refractory period when a cell is incapable of re-stimulation followed by a relative refractory period where the cell can be re-stimulated if a large enough stimulus is applied (Sampson and McGarth, 2015). A threshold voltage separates the absolute and relative refractory period.

3.1.3 Action Potential Dynamics

To describe and quantify AP dynamics, voltage characteristics and time intervals are defined as (Fig. 2):

- Action Potential Duration (APD) – time taken for repolarisation after stimulus.
- Diastolic Interval (DI) – time between the end of one AP and the beginning of another.
- Basic Cycle Length (BCL) – time between external stimuli.
- Action Potential Activation (APA) – maximum voltage at depolarisation.
- Resting potential (RP) – lowest voltage at rest.

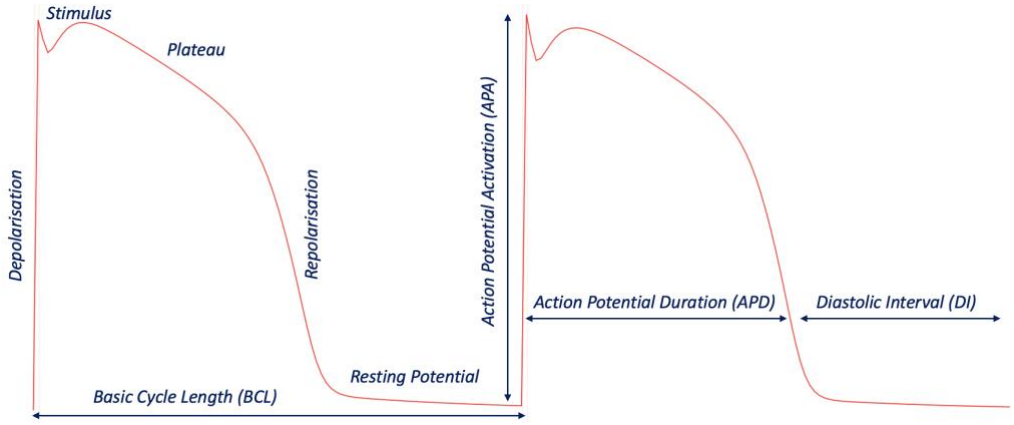

Fig. 2. AP dynamics demonstrating time intervals and voltage characteristics.

3.1.4 The ECG and Cardiac Alternans

The ECG is a summation of the electrical activity of the heart and the APs of all heart cells measured on the chest wall of a patient. By placing electrodes on the chest, the electrical activity of the heart can be shown as waveforms, including the P-wave, the QRS complex and the T-wave. The P-wave corresponds to depolarisation of the atria, the QRS complex is due to depolarisation of the myocytes of the ventricles, and the T-wave represents the repolarisation of the myocytes of the ventricles (Sampson and McGarth, 2015).

Cardiac alternans, a beat-to-beat variation in the amplitude and duration of the AP of the cardiac myocyte, is also evident on the surface ECG, for instance, in changes of the shape of the T-wave. T-wave alternans has been associated with arrhythmogenesis and is a good predictor of sudden cardiac death (Wilson and Rosenbaum, 2007; Cutler and Rosenbaum, 2009; Tse et al., 2016). On a cellular level this is due to changes in repolarisation time lengths, which increase at higher heart rates.

3.1.5 Action Potential Duration Restitution

Cardiac T-wave alternans is thought to relate to APD restitution. APD restitution is the normal shortening of the APD as a response to increased heart rates and probably represents an adaptive mechanism to preserve the diastole of the heart to allow appropriate ventricular filling, and preserve cardiac output (Wilson and Rosenbaum, 2007). As the heart rate increases, the APD shortens to preserve the DI (Goldhaber et al., 2005). At a constant BCL, APD prolongation during one beat will have to be followed by a short DI at the next beat, which according to restitution will shorten the following APD. This then causes repeated long and short alternating cycles (Wilson and Rosenbaum,

2007). Plotting APD over DI produces a restitution curve. It is thought that alternans occurs when the slope of the restitution curve exceeds one (Goldhaber et al., 2005; Wilson and Rosenbaum, 2007). A steep restitution slope might cause a break of electrical wave behaviour resulting into a fibrillation-like state (Goldhaber et al., 2005; Weiss et al., 2000).

3.2 Mathematical Models of Cardiac Electrical Excitation

Mathematical models of cardiac electrical excitation describe voltage changes at the cell membrane over time, which are typically expressed as a system of non-linear ODEs. These are thought to represent the opening and closing of ion channels, the movements of ions across the membrane and the resultant changes during an AP. Many different mathematical models have been developed (Hodgkin and Huxley, 1990; Ten Tusscher et al., 2006; Luo and Rudy, 1991).

The two different models explored in this project are the Mitchell-Schaeffer (MS) model, selected for its simplicity, which makes it efficient for numerical simulations and the Iyer-Mazhari-Winslow (IMW) model selected for its complexity, which allows better approximations of physiological processes (Mitchell, 2003; Iyer et al., 2004).

3.2.1 Mitchell-Schaeffer Model

The MS model is a simplified model used to reproduce the basic shape and timing of the AP with minimal complexity. It uses two variables; the membrane potential and a gating variable, which control excitability and recovery (Mitchell, 2003). The dynamics are governed by equations that represent the rapid upstroke, plateau, and repolarisation phase without modelling individual currents where:

- $v(t)$ represents the normalised membrane potential (0 at rest, 1 at full depolarisation)
- $h(t)$ represents a gating variable that controls the ability of the cell membrane to be excited again (like a recovery and inactivation gate).

The model equations are:

$$\frac{dv}{dt} = \frac{hv^2(1-v)}{\tau_{in}} - \frac{v}{\tau_{out}} \tag{1}$$

$$\frac{dh}{dt} = \begin{cases} 1 - \frac{\mathrm{h}}{\tau_{open}} \; if \; v < v_{gate} \\ -\frac{\mathrm{h}}{\tau_{close}} \; if \; v \geq v_{gate} \end{cases} \tag{2}$$

The MS model defines three currents, an inflowing, outflowing and stimulus current (Mitchell, 2003). The inflowing current characterises all currents together that depolarise the cell, including the sodium and calcium ion currents. The cubic function $hv^2(1-v)$ describes the voltage-dependence of the current, and the gating variable h controls the opening and closing of the channels. The outflowing current characterises all currents that repolarise the cell, including potassium and chlorine ion currents, and is not controlled by a gating variable. Additionally, a stimulus current is introduced (Ngoma et al., 2015).

In the model:

- τ_{in} controls the rapid upstroke.
- τ_{out} governs the repolarisation speed.
- τ_{open} sets the recovery time of excitability.
- τ_{close} determines how quickly excitability is lost once the membrane depolarises.
- ν_{gate} is a threshold potential for switching $h's$ dynamics.

(Mitchell, 2003)

The gating variable $h(t)$ mimics the refractory behaviour, the period after the AP and therefore avoids detailed channel kinetics. At rest ($v < v_{gate}$), $h(t)$ slowly recovers towards 1 at a rate given by τ_{open}. A value of $h(t) \approx 1$ implies the cell is fully excitable. When the membrane depolarises ($v \geq v_{gate}$), $h(t)$ starts to decrease towards 0 at a rate given by τ_{close}, representing the inactivation of excitability, which is similar to the sodium channel inactivation in a complex model. When $h(t)$ is small, the source term $hv^2(1-v)$ in the ν-equation becomes small, so any new stimulus is too weak to trigger a new AP. This is the absolute refractory period. As $h(t)$ recovers, the cell gradually becomes responsive, producing a relative refractory period until $h(t)$ returns to 1 (Fig. 3). By this mechanism, the model can represent the timing and graded nature of the refractory period with just one additional variable (Mitchell, 2003; Ravon et al., 2019).

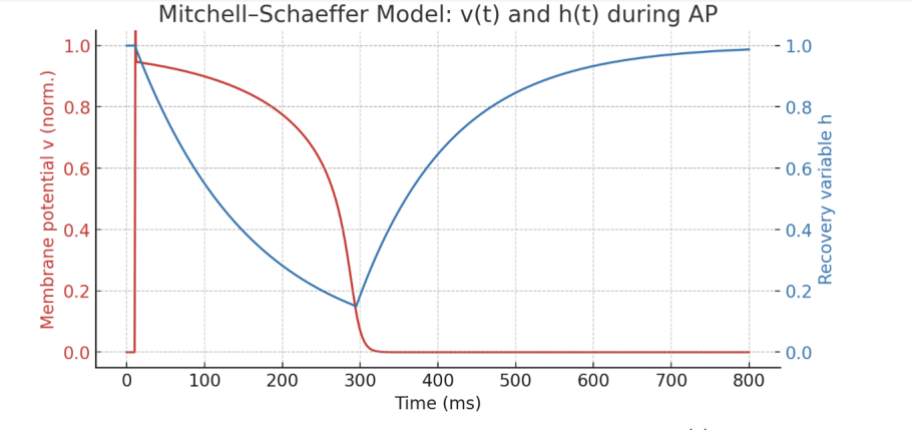

Fig. 3. Mitchell-Schaeffer Model: $v(t)$ and $h(t)$ during the AP. The blue curve shows the gating variable $h(t)$ that makes the model simple. $h(t)$ is dropping during depolarisation and then slowly recovers modelling the refractory period. Parameter values used for diagram: τ_{in} = 0.3, τ_{out} = 6, τ_{open} = 120, τ_{close} = 150, ν_{gate} = 0.13. (Figure adapted from Mitchell, 2003; Ravon et al., 2019; Ngoma et al., 2015)

3.2.2 Iyer-Mazhari-Winslow Model

The IMW model is a complex ionic model of the human ventricular myocyte. It models major currents, such as the fast sodium current, transient outwards potassium current, rapid delayed rectifier potassium current and L-type calcium current by using detailed gating and kinetic formulations (Iyer et al., 2004). It employs continuous-time Markov-chain models for the gating dynamics of these currents and is tailored to human physiological data at physiological temperatures. With 67 state variables it is a high-dimensional model designed to reproduce a wide range of electrophysiological behaviour. Therefore, it is widely used to explore arrhythmias, drug channel interactions and the influence of genetic mutations on ionic channels (Iyer et al., 2004).

The IMW model is implemented as a large coupled system of ODEs that advance the membrane voltage, change ionic concentrations in multiple tissue compartments and analyse state-occupancy probabilities of several ion channels. Numerically, this is treated as a highly stiff system that requires integration with implicit ODE solvers (Omale, 2014). The model is available in the CellML library accessible via https://www.cellml.org (Lloyd et al., 2008).

3.2.3 Challenges with the mathematical models

These mathematical models display different levels of complexity. The IMW model is one of the most detailed ionic myocyte models available and uses Markov-chain kinetics for several channels

with detailed calcium dynamics (Iyer et al., 2004). This complexity is motivated by a need to capture detailed biophysical reactions but introduces computational cost and limits the mathematical analysis of the model. Modelling many ion channels is computationally expensive to run especially when trying to upscale the model to tissue or organ-scale simulations. Multiple parameters are based on experimental data from different labs and even non-human tissue, which means that small changes to parameters could cause large changes in AP morphology (Iyer et al., 2004).

The MS model on the other hand uses a small non-stiff ODE system, which makes simulations of it performant with simple explicit methods and minimal computational cost. It can therefore be used for large-scale tissue simulations, wave propagation studies and cell-level biophysics. However, by using simplified mathematical representations, the MS model is unable to provide insight into the underlying physiological activity of the heart (Amuzescu et al., 2021).

3.3 Deep Learning and Neural Networks

Deep learning is a type of ML where NNs are used to extract patterns from data. NNs consist of connected units called neurons. Each neuron takes inputs, multiplies them by weights, adds a bias, and passes the result through an activation function. Mathematically, this can be expressed as:

$$y = \sum_{i=1}^{n} w_i x_i + b \tag{3}$$

where x_i represents the ith input, w_i is the corresponding weight, b is the bias term and y is the output.

The activation function is a mathematical function that introduces non-linearity. Injecting non-linearity is important for the NN to model more complex relationships beyond linear maps. Many different activation functions exist, such as sigmoid, hyperbolic tangent (tanh) and rectified linear unit (ReLU) functions.

NNs consist of several layers of interconnected neurons: the input layer, the output layer, and multiple hidden layers. They are trained to produce accurate outputs by adjusting the weights and biases defining the NN. This adjustment is guided by the loss function, which measures the difference between the predicted output and the correct answer. The NN is penalised for a large loss and the weights are updated to reduce this error. The loss function defines a loss landscape over the space of possible weight values (Theodoridis, 2025). The goal of training is to minimise this loss, which can be achieved through an optimisation technique called gradient descent. Through gradient

descent, the gradient of the loss with respect to the weights is calculated and the weights are updated by taking small steps in the direction that reduces the loss. This iterative process continues until the loss converges to a local minimum. A common gradient descent optimiser is the Adaptive Moment Estimation (ADAM) optimiser (Kingma and Ba, 2014).

Backpropagation is the algorithm used to efficiently compute these gradients across all layers of the network. The learning rate determines the size of the steps taken during each update and controls how quickly the NN learns (Theodoridis, 2025).

Integrating NNs into differential equation frameworks to model cardiac systems introduces challenges due to the highly non-linear and discontinuous nature of the differential equations required. Additionally, established theoretical guidance is lacking for neural components such as the number of layers, or the selection of activation functions within universal differential equations (UDEs) (Rackauckas, 2020).

3.4 Julia Programming Language

Julia was selected for this project due to its high performance and environment tailored for scientific computing (Modeling and simulation case studies, 2025). Its Scientific Machine Learning (SciML) software ecosystem integrates traditional scientific computing with modern ML methods (Bezanson et al, 2017; Rackauckas, 2022a; Rackauckas, 2022b; Rackauckas, 2025).

Julia's strength is in modelling dynamical systems governed by ordinary differential equations (ODEs) with the DifferentialEquations.jl package (Rackauckas and Nie, 2017). The MS and IMW models simulate changes in the electrical activity of the heart using systems of ODEs which cannot always be solved analytically, so numerical solvers are needed to approximate their solutions (Omale et al., 2014; Denis, 2020). Julia's SciML aids easy integration of NNs into scientific computing workflows. For the deep learning component, the Julia framework Lux.jl was chosen for its functional design and good integration with other packages (Pal, 2025). As well as this Julia offers a range of optimisation libraries such as Optimisation.jl that facilitates NN training (Dixit and Rackauckas, 2023). Julia's Makie.jl, a plotting package leveraging Julia's type system and multiple dispatch paradigm, and the GLMakie.jl backend for Makie were particularly useful (Danisch and Krumbiegel, 2021).

4 Methodology

4.1 Mathematical Model Implementation

Details of many electrophysiological models, such as structural components, the mathematics, and metadata, can be accessed through the CellML Model Repository (Lloyd et al., 2008). Using the repository simplifies the integration of highly complex models as it eliminates the need to manually set-up the model.

In this project the IMW model was accessed through the CellML Model Repository, which simplified the integration of the complex model with 67 variables. As the MS model only has 2 variables, the equations were manually coded and solved, so the CellML model repository was not required.

The MS model produces voltage values that are already normalised between 0 and 1. In contrast, the IMW model generates voltages in the original physiological range. To ensure a fair comparison between the two models, the IMW voltage values were rescaled to the same normalised range. These values were obtained by solving the base model for an infinite-period stimulus, meaning that the model was stimulated once and then allowed to return fully to its resting state, effectively generating a single isolated AP.

Using the DifferentialEquations.jl package, a composite solver approach combining a fast solver for non-stiff systems with a more stable solver for stiff systems was implemented, allowing for dynamic switching between the two as needed during integration (Rackauckas et al., 2025a; Rackauckas et al., 2025b). This adaptive strategy ensures computational efficiency and numerical stability. The Tsitouras 5/4 order Runge-Kutta Solver (Tsitouras, 2011) and Rosenbrock Solver (Shampine and Reichelt, 1997) were employed.

A critical aspect of accurately modelling the heart's electrical activity is handling external stimuli such as pulsed stimuli. These discontinuous inputs introduce sharp changes, or discontinuities, into the system's behaviour. It is required to explicitly pass the time points of these discontinuities to the ODE solver using the $tstops$ argument, so that the integration can correctly switch between the unstimulated and stimulated responses (Rackauckas et al., 2025a; Rackauckas et al., 2025b).

Julia's Makie.jl, a plotting package, was used to visualise model outputs and to create most of the figures in this report (Danisch and Krumbiegel, 2021).

4.2 Mitchell-Schaeffer Model Parameter Exploration

The MS model includes four time-scales τ_{in}, τ_{out}, τ_{open} and τ_{close} that control the rates of the gating processes during the cardiac AP (see chapter 3.2.1.) These model parameters were explored by changing the values of the parameters and plotting the curves. This was to inform selection of parameter values for MS model tuning.

4.3 Mitchell-Schaeffer Model Tuning

To better approximate the MS model to the IMW model, the parameters of the simpler MS model were tuned. As the τ parameters determine the duration and amplitude of the AP potentials, and their long-term evolution, fine-tuning them allows the MS model AP trace to better approximate that of the IMW model.

Parameter fine-tuning was performed with the Nelder-Mead optimiser (Singer and Nelder, 2009), which chooses the best τ parameters to minimise the difference between the MS and IMW AP traces. Once optimised, further improvements might be achieved by adding a NN-based current to capture dynamics not easily modelled by the MS model alone.

4.4 Parameter Predictions with Neural Networks

AP data was generated by solving the MS and IMW ODE systems for different BCLs. Key electrophysiological features were extracted based on a voltage threshold, which marks the point of repolarisation.

Time intervals were categorised based on the time point the membrane voltage crosses this threshold. The APD was defined as the time interval the voltage remained above the threshold, while the DI was the time interval the voltage remained below it. Changing the threshold affects the vertical positioning of the reference points, as it determines the point at which repolarisation is considered complete (shown in Fig 4). The maximum voltage reached during the AP is classified as the APA. These values were then saved to a file. The file format consists of a single BCL value followed by comma separated APD, DI and APA values.

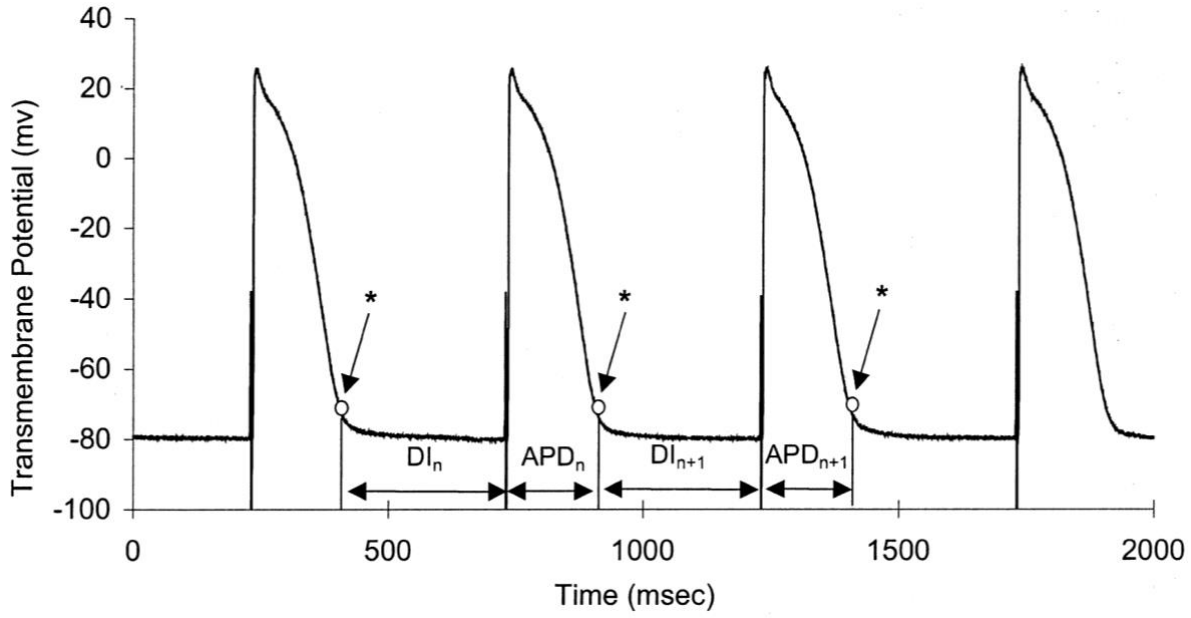

Fig. 4. Voltage threshold representation marked with (*) on the AP voltage trace (Wu and Patwardhan, 2004)

A NN was implemented using Julia's Lux package to predict the next APA and APD. The inputs of the NN were defined as the current APD, APA, and BCL, and the expected output, the next APD and APA. The trained NN can extrapolate these parameters to the next BCLs, if desired, without having to solve the full ODE model again. Mathematically, this can be written as:

$$APD_{n+1},\ APA_{n+1} = NN(APD_n\ , APA_n\ , DI_n) \tag{4}$$

4.5 Augmenting the Mitchell-Schaeffer Model with Neural Networks

A UDE can be defined as a differential equation where part of the dynamics is modelled by a NN (Rackauckas, 2020). In the present study, we restrict the augmentation to the addition of a NN current (I_{NN}), which influences the dynamics of the transmembrane potential to produce a universal ordinary differential equation (UODE). For further implementations of this method see Rackauckas, 2022a.

Julia's Lux package was used to define a NN and integrate it into the MS model ODE system (Pal, 2025). To initialise the un-optimised weights and biases of the NN a fixed random number generator seed was used. This ensures reproducibility of the training process and enables fair comparisons between model configurations. The NN takes the voltage and gating variable as inputs and produces

a single output, representing the modification of the MS model voltage changes over time. The original derivatives are computed using the MS ODE model, and the NN is trained to learn a correction term that modifies these derivatives, effectively capturing additional dynamics not captured in the base MS model.

Training is guided by a simple, standard mean squared error (MSE) loss function between the prediction from the hybrid model and IMW solution. Gradients of the loss are computed with respect to the NN parameters to optimise the correction.

Unlike standard backpropagation through individual NN layers, the weights are updated based on how changes in the NN parameters influence the entire trajectory of the ODE solution. This is made possible through DiffEqSensitivity.jl, which enables backpropagation through the ODE solver using techniques like adjoint sensitivity analysis.

The resulting optimisation problem is defined using Optimization.jl, where automatic differentiation was chosen to be handled by Zygote.jl, enabling differentiable programming for the many parameters involved. The problem is solved using a combination of the ADAM and Limited-memory Broyden-Fletcher-Goldfarb-Shanno (LBFGS) solvers. ADAM is better suited for navigating complex landscapes in early training, while LBFGS is more appropriate in later training as it can achieve more precise convergence in fewer iterations near local minima. So, using ADAM for initial iterations ensures that a good region of the parameter space is reached, and then switching to LBFGS for the final iterations optimises and accelerates convergence. This approach has been shown to achieve a smaller loss in the same number of iterations (Fig. 5) (Urbán et al., 2025). A callback function was used to record the loss at each iteration, so that the optimisation could be monitored. This enabled the loss to be plotted so that the training could be assessed.

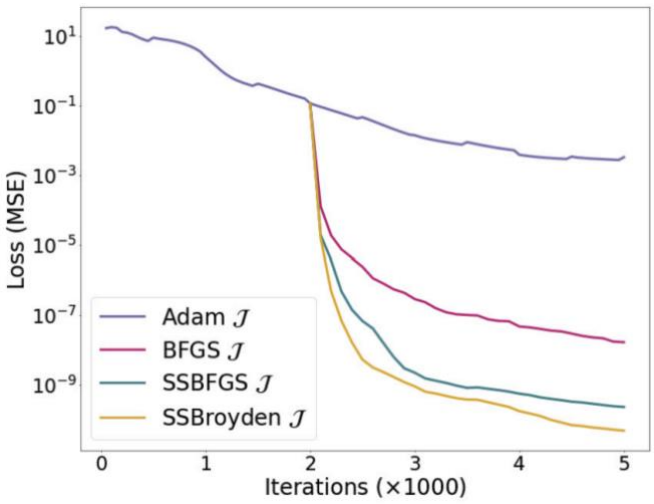

Fig. 5. Plot showing improved reduction in loss achieved when combining ADAM with the Broyden-Fletcher-Goldfarb-Shanno optimiser when initiated after initial iterations. (Urbán et al., 2025)

5 Data Analysis and Results

5.1 Exploring Existing Models

When cardiac cells are subjected to periodic external stimulation, different dynamical responses may be observed. At slower stimulation rates, every stimulus produces an identical AP, which is called a 1:1 response. As the stimulation rate increases, the membrane potential fails to recover fully between stimuli, resulting in only every second stimulus triggering an AP, known as a 2:1 response. This phenomenon is associated with cardiac alternans, a beat-to-beat alternation in the duration or amplitude of APs, despite a constant pacing rate. Due to restitution, APDs shorten when the heart rate increases (Asghari- Targhi, 2017). These phenomena could be demonstrated when plotting the MS model and IMW model voltage traces against varying BCLs.

5.1.1 Mitchell-Schaeffer Model

To show the functionality of the MS model implementation the voltage traces were plotted for varying BCLs (Fig. 6). The plots confirm the capability of the MS model to simulate cardiac alternans behaviour.

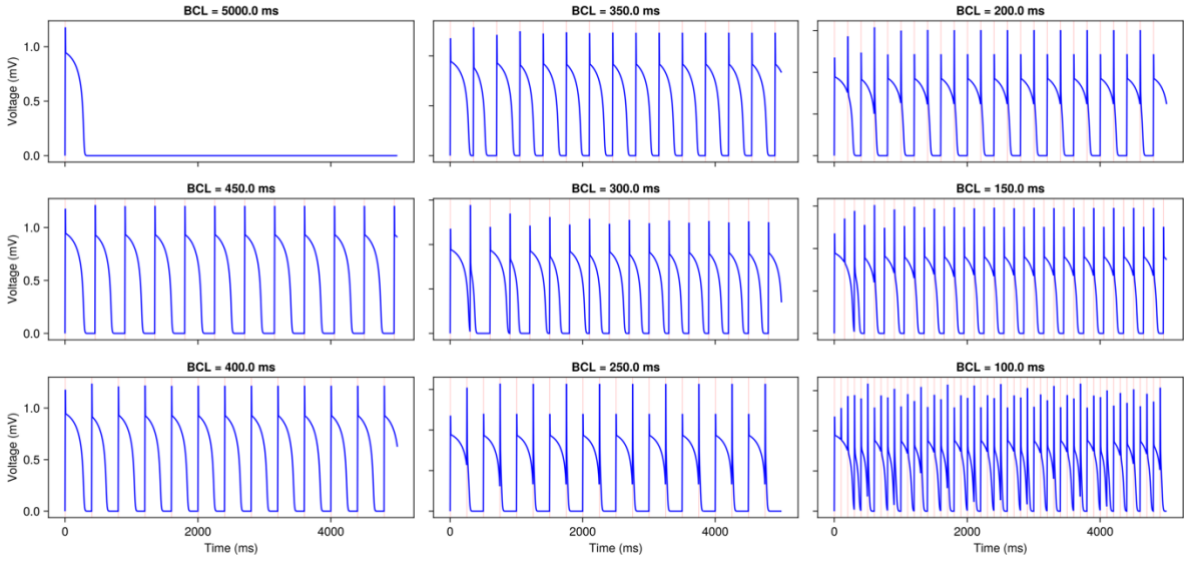

Fig. 6. Plot of normalised MS model voltage traces at varying BCLs. Alternans is shown for a single beat at 400ms and 350ms, for several beats at 300ms BCL, a 2:1 response is shown at 250ms and 200ms BCL and APs of varying duration at 100ms BCL.

5.1.2 Iyer-Mazhari-Winslow Model

To show the functionality of the IMW model implementation the voltage traces were plotted for varying BCLs (Fig. 7).

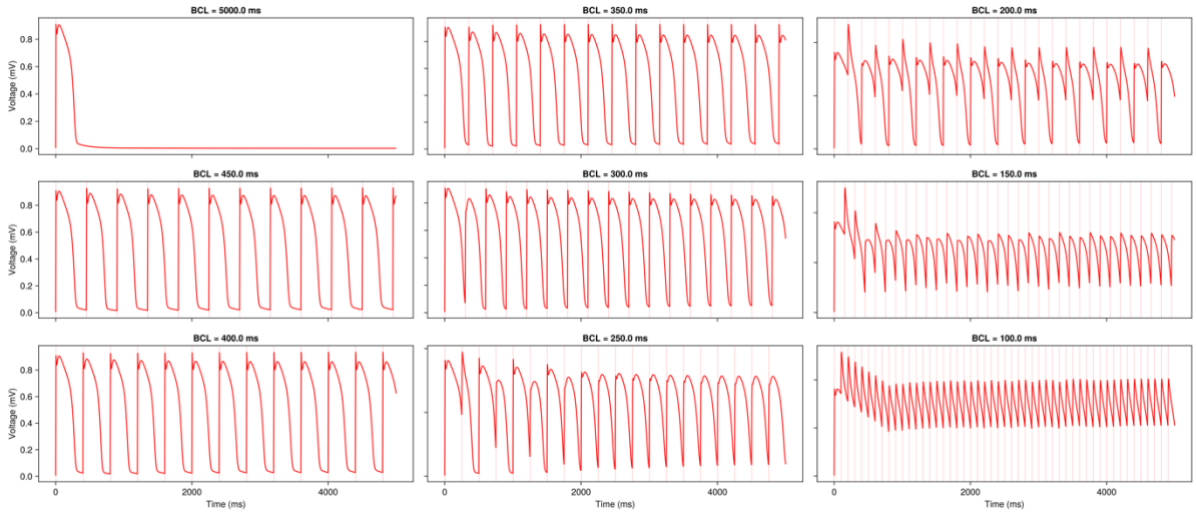

Fig. 7. Plot of the normalised IMW model voltage traces at varying BCLs. Alternans is shown for a single beat at 300ms, for several beats at 250ms BCL, a 2:1 response is shown at 200ms BCL and fibrillation at 100ms.

5.1.3 Comparing the Mitchell- Schaeffer and Iyer-Mazhari-Winslow Models

When compared to the MS model AP shape, the IMW model AP exhibits a brief dip immediately following depolarisation, known as the phase-1 notch, simulating the transient outward potassium current. This demonstrates the better physiological accuracy of the IMW model when compared to the simpler MS model. At higher pacing rates this notch persists but becomes less prominent. This is followed by long, flat plateau phase simulating the sustained balance between inward calcium and outward potassium currents (Winslow, 2011). The MS model AP does not demonstrate a phase-1 notch and has a gradual falling edge during repolarisation, giving a triangular AP trace. The variation of the AP amplitude is always much higher when simulated by the MS model when compared to the IMW model.

Overall, the voltage traces of the two models are similar in shape, but the trace of the IMW model better approximates the electrophysiological processes of the heart muscle cell membrane AP.

To facilitate comparison, the voltage traces of the two models, were both plotted on the same graph (Fig. 8).

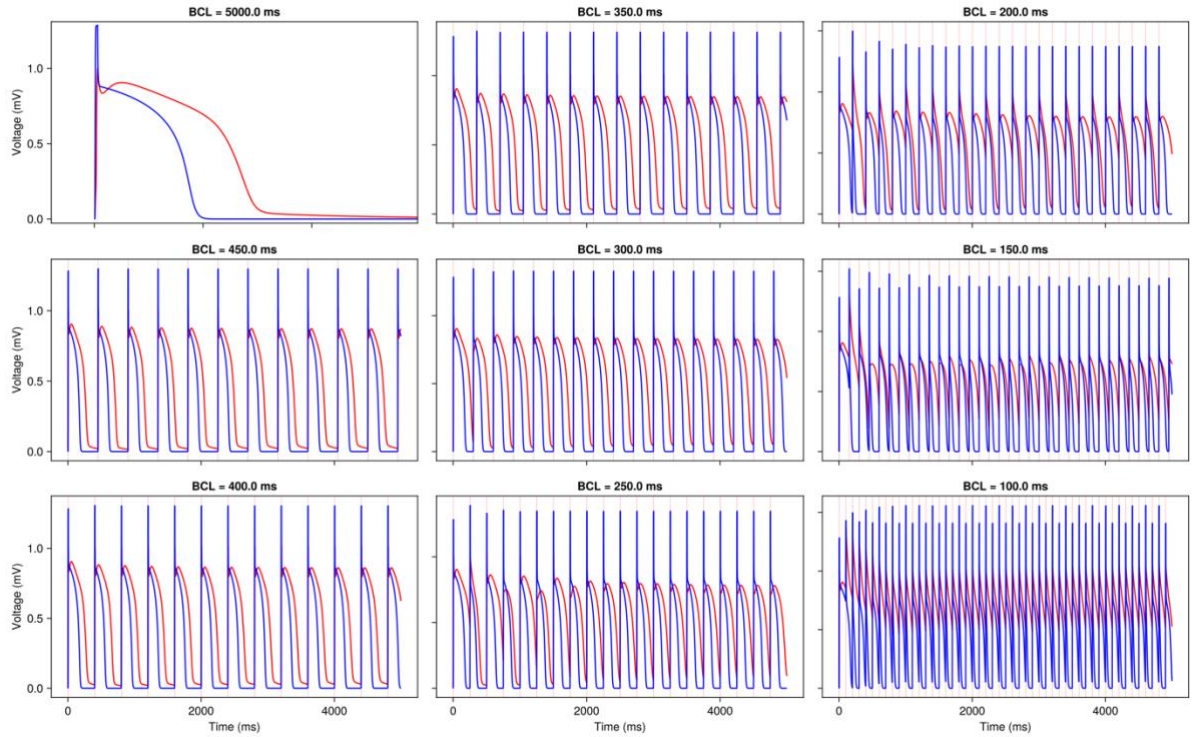

Fig. 8. Plot of the IMW model voltage traces in red and the MS model voltage traces in blue at varying BCLs demonstrating the phase-1 notch and more sustained plateau of the IMW model AP compared to the more triangular shaped AP simulated by the MS model. The IMW model trace is more physiological.

5.2 Mitchell-Schaeffer Model Parameter Exploration

To explore the influence of the different τ variables on the shape of the AP, voltage traces for the MS model were plotted for different τ variable values (Fig. 9). These demonstrate that the constants τ_{in}, τ_{out} and τ_{close} have significant influence on the plateau phase. A low value of τ_{in}, a high value of τ_{out} and a high value of τ_{close} produce an AP with a prolonged plateau phase. However, varying τ_{open} has no influence on the shape of the AP. However, at low τ_{open} values, the model produces alternans behaviour with repeat APs of shorter duration. These are also produced early before the voltage trace return to the baseline, simulating the relative refractory period of the cardiac myocyte. These traces suggest that the alternans behaviour of the MS model is probably simulated by the τ_{open} parameters.

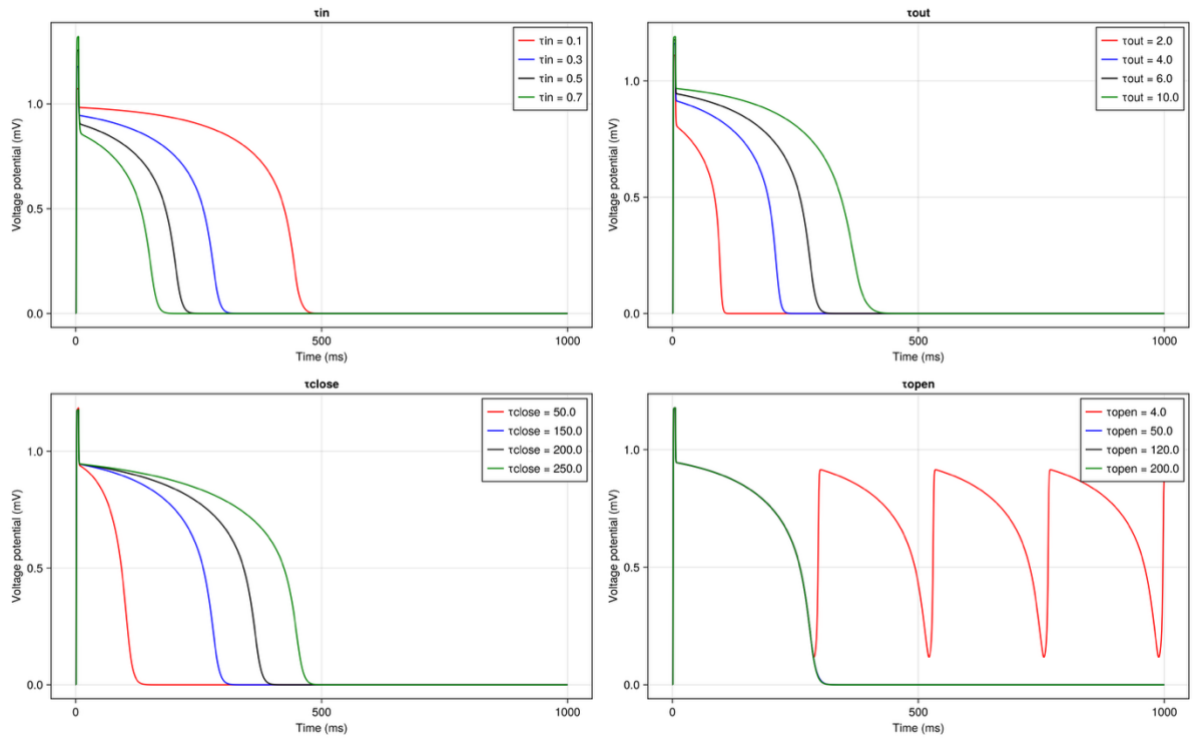

Fig. 9. Voltage traces at BCL 5000.0ms plotted for varying values of the constants τ_{in}, τ_{out}, τ_{close} and τ_{open}. Graphs show influence of the constants on the AP shape. τ_{in}, τ_{out}, and τ_{close} prolong the plateau phase of the AP. τ_{open} has no influence on the AP shape but simulates alternans at low values.

5.3 Approximating the Mitchell- Schaeffer Model to the Iyer-Mazhari-Winslow Model

5.3.1 Tuning the Mitchell- Schaeffer Model

The MS model was tuned to more closely emulate the IMW model, with the primary goal of aligning APDs while adapting the overall shape of the AP more to the IMW model. To tune the model, the IMW solutions are sampled at a discrete set of times, and the MS solutions are solved to those same time points. A loss using the MSE is formed between the two sets. The inputs consist only of the time-scale parameters of the MS model, and thus the tuning is restricted to a low-dimensional space compared with the NN augmentation used for the MS model. The voltage traces of this modified Michell-Schaeffer (mMS) model were plotted against varying BCLs (Fig. 10).

At long BCLs, the mMS model traces match the traces of the IMW model better with similar upstroke timings, plateau shape morphology and repolarisation profile. However, as pacing accelerates, the match deteriorates slightly, with APs simulated by the mMS model losing much of its plateau shape. Yet, under these faster pacing rates, the mMS model reproduces IMW's action potential amplitude

far more accurately, maintaining closer agreement even at the highest pacing frequencies. The optimisation function weights all traces the same. The traces with short BCLs, have more APs, thus if the measure is fitting the AP, then the optimisation targets the higher frequency forcing to the detriment of the typical AP. As shown in chapter 5.1.3 the original MS model better simulates the IMW model than the mMS model.

Ngoma et al. (2015) found that differentiable optimisation methods for the MS model, for instance, with gradient methods did not work because the sensitivities of the solution with respect to the model parameters grow rapidly with the regularisation parameters.

As the Nelder-Mead optimisation method used for tuning is a heuristic derivate-free optimisation method it was hoped it would work better than demonstrated here.

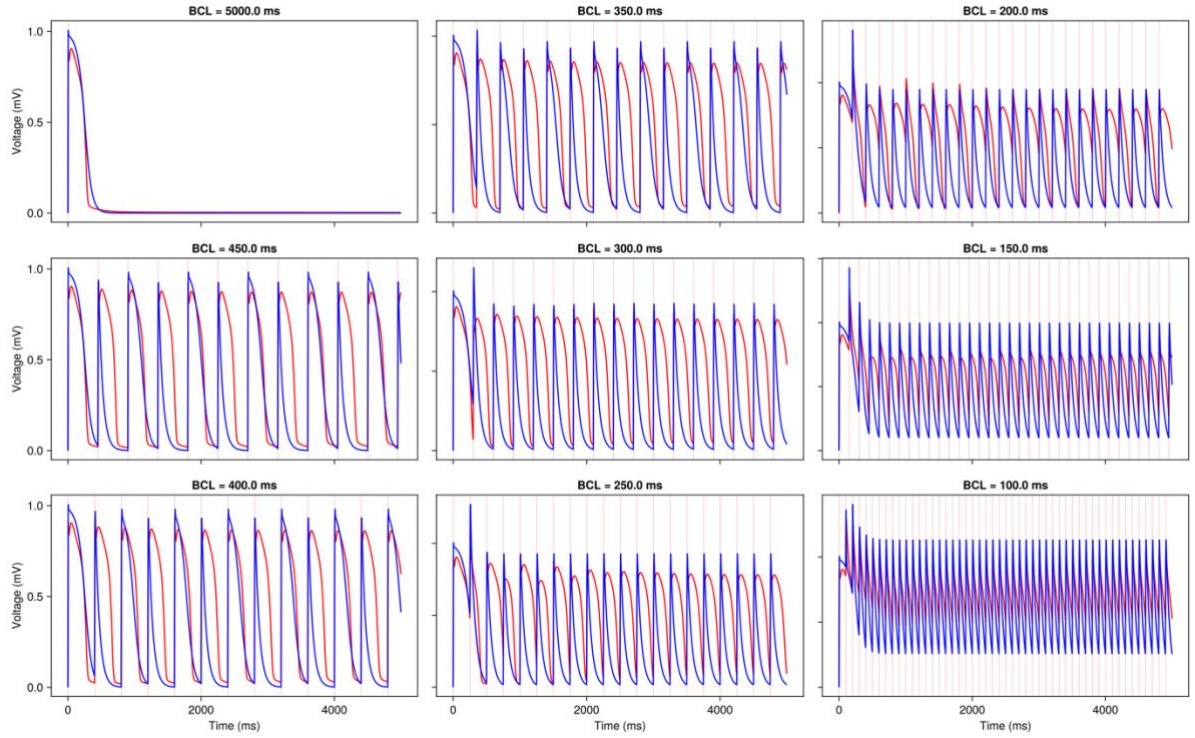

Fig. 10. Plot of the IMW model voltage traces in red and the mMS model voltage traces in blue for varying cycle lengths. At long BCLs, the mMS model traces better match the IMW model traces. However, when pacing accelerates the match deteriorates.

At long BCLs the mMS model reproduces the IMW voltage traces well, but the match progressively degrades as the pacing rate increases. This suggests that the tuning procedure failed to produce parameter sets that generalise across the full range of pacing conditions. This deviation from intended behaviour highlights two potential issues. Firstly, there is a sampling bias in the optimisation data, which could cause the optimiser to favour parameter regions that work well for those areas while underperforming for shorter BCLs. Secondly, there is a lack of globalisation in

tuning, which means that parameter adjustments that would improve high-rate behaviour are overlooked. Moreover, the optimisation landscape is quite complex. By optimising over a trajectory, we observe very different effects for short- and long-time deviations.

When the optimisation drives the weights and biases of the augmenting NN to zero, then the mMS model reduces to the original MS model. When this occurs, the additional degree of freedom introduced into mMS is not utilised, and the effective tuned model is indistinguishable from MS.

5.3.2 Comparison of the modified Mitchell-Schaeffer Model to the other Models

Plots of the APD, APA and DI of the IMW, MS, and mMS models against different BCLs were generated to compare their electrophysiological behaviour across a range of pacing cycle lengths (Fig. 12). The resultant maps allow the study of how beat-to-beat changes unfold. The APD, APA, and DI maps are particularly useful to evaluate alternans development in the system at decreasing BCL.

When analysing the APD, the MS model exhibits a bounded ceiling and therefore does not match IMW's extreme behaviour at short cycle lengths. Nevertheless, it reproduces IMW's behaviour at long BCLs, where both models converge to a stable plateau. The dips in the APD traces simulating alternans are not perfectly aligned between the different models (Fig. 11a). Neither the MS model nor the mMS model fully reproduce IMW model's extreme APD at rapid pacing rates. This indicates that while the MS model is well tuned for stability at extreme cycle lengths, the mMS model reflects the IMW model's greater susceptibility to variability under stress, but both models incompletely capture the full dynamic range of the IMW model restitution.

At very short BCLs, the MS model demonstrates a pronounced reduction in APA, reflecting diminished excitability relative to the IMW model with values recovering smoothly as the pacing interval increases. By contrast, the mMS model better replicates IMW model's variability at faster pacing, capturing its dynamic fluctuations in amplitude, though it differs from IMW in the specific form of its instabilities (Fig. 11b).

With respect to DI, the IMW model displays an almost linear relationship with BCL, reflecting a proportional increase in recovery time as pacing slows. In contrast, the MS model deviates from this linearity at short BCLs. The mMS model diverges even further, with abrupt transitions as BCL decreases. The model cannot respond to every stimulus resulting in alternans. Thus, while the IMW

model maintains a simple linear DI-BCL relationship over most pacing intervals, the MS and mMS models each distort this behaviour in different ways (Fig. 11c).

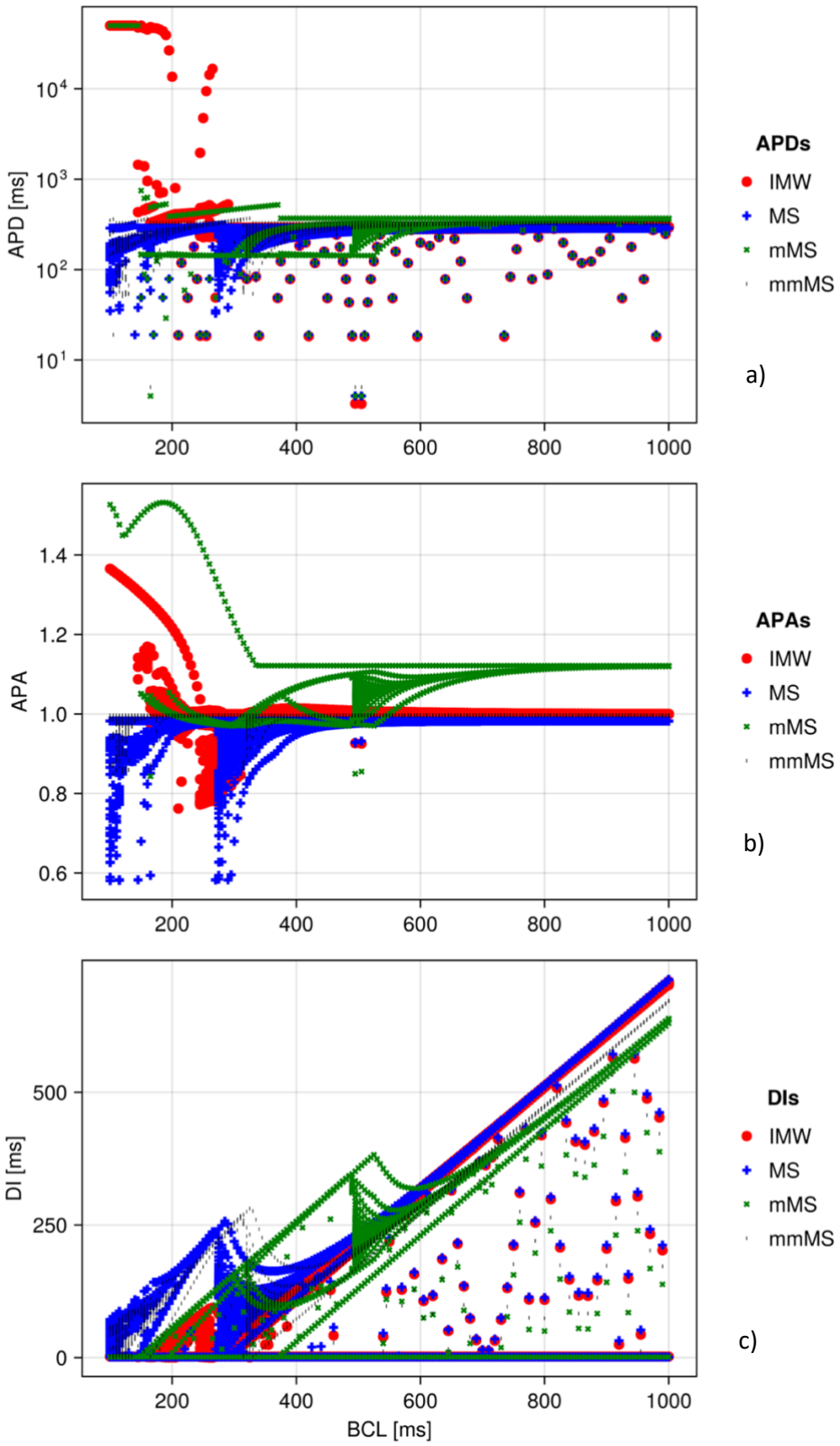

Fig. 11. a) Plot of the APD, b) the APA and c) the DI of the IMW model, MS model and mMS model for a range of BCLs.

5.4 Action Potential Activation and Action Potential Duration Predictions with Neural Networks

A NN was designed and optimised to predict the APD and APA based solely on the preceding APD, APA and DI. The APD versus BCL and APA versus BCL curves were plotted for the MS, IMW and mMS models both as simulated by the model and then with the next beat predicted by the NN (Fig. 12-17).

The best prediction was achieved for the MS model probably aided by the simplicity of the model. The NN predicts the occurrence of the alternans at approximately 100ms and 300ms very accurately as shown on the superimposed plot (Fig. 12c). The NN can act as a valuable tool to explore the degree of memory needed in cardiac restitution. As the NN can achieve accurate predictions using only the previous beat as input, it suggests that short-term memory is sufficient for accurate prediction of APDs and APA. Otherwise, longer-term history would be needed to meaningfully capture and predict the system's non-linear dynamics which will increase computational cost.

The predictions for the IMW model are also good, apart from a slight deviation in the prediction of the APD in the curve at BCLs around 400-500ms (Fig 14-15). However, the predictions achieved for the mMS are less optimal probably related to their complexity (Fig. 16-17).

All NNs used a learning rate of 0.01, the activation function ReLU and a 0.2 voltage threshold. For the MS and IMW model an NN with layers 3-> 16->32->16->2 and 10000 epochs was used, while for the mMS model a slightly larger NN with layers 3-> 16->32->64->32->16->2 with 20000 epochs was used.

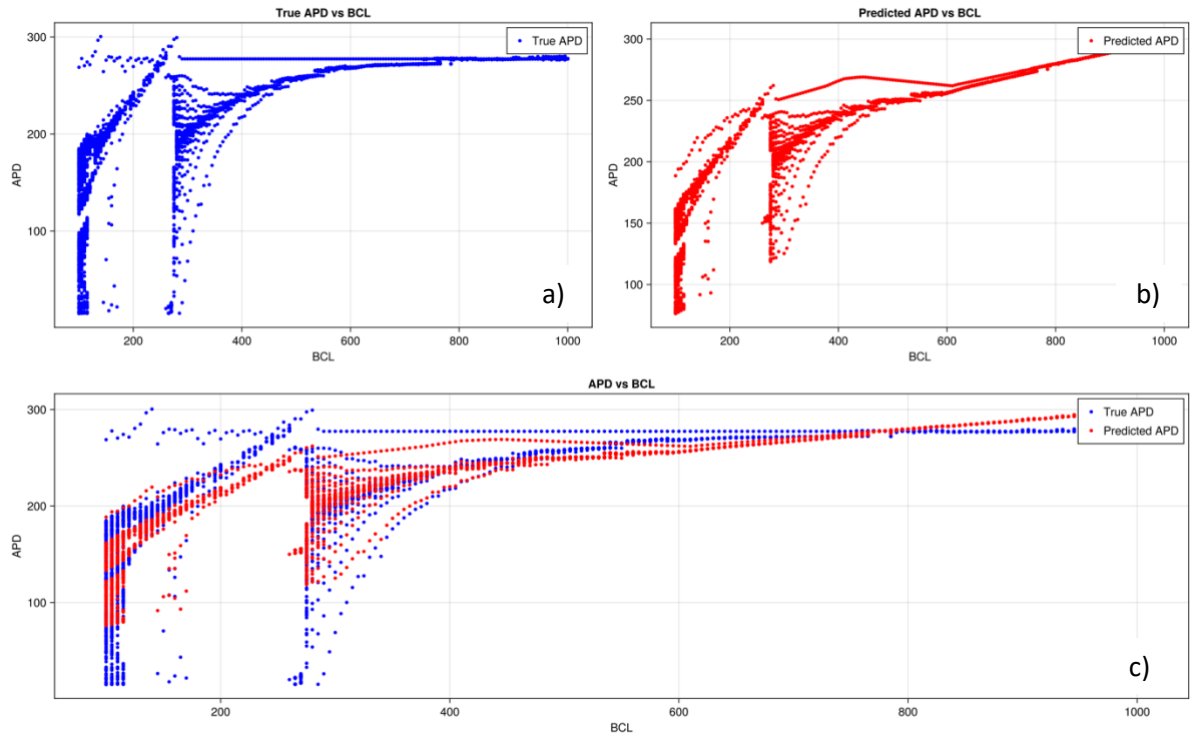

Fig. 12. APD vs BCL plot for the MS model. a) Model simulated next APD, b) NN predicted next APD and c) both plots superimposed on each other.

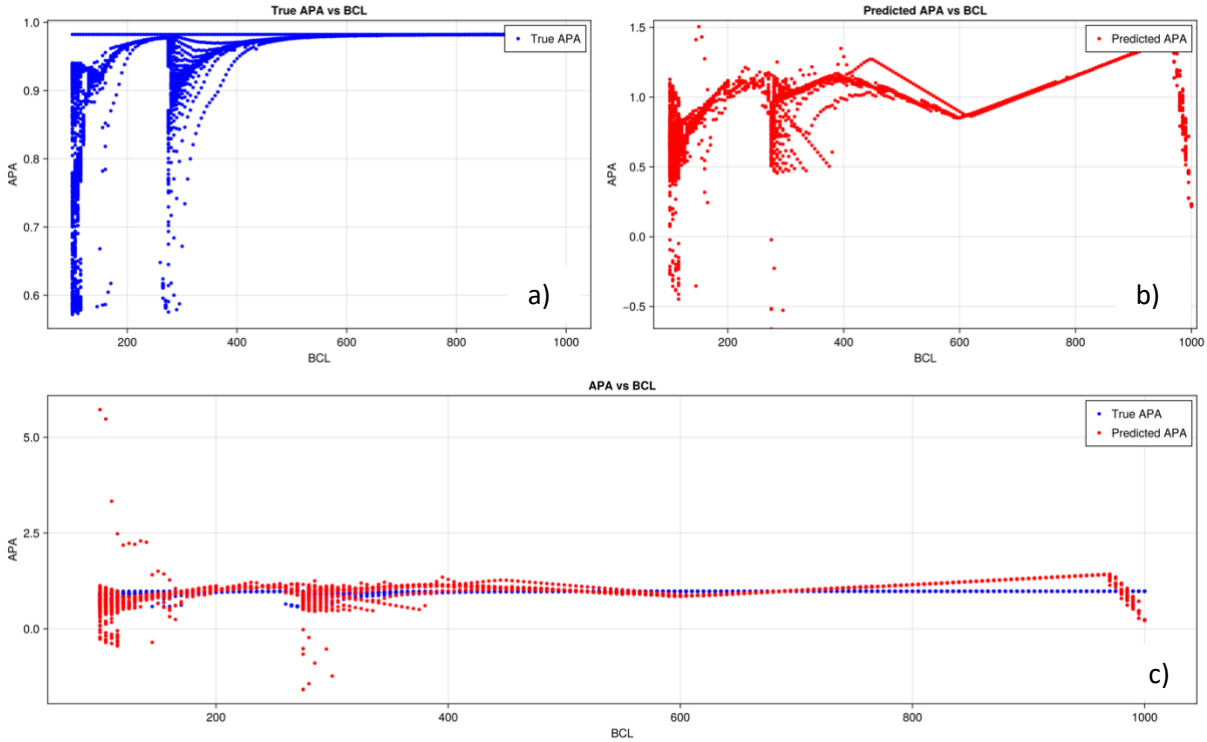

Fig. 13. APA vs BCL plot for the MS model. a) Model simulated next APA, b) NN predicted next APA and c) both plots superimposed on each other.

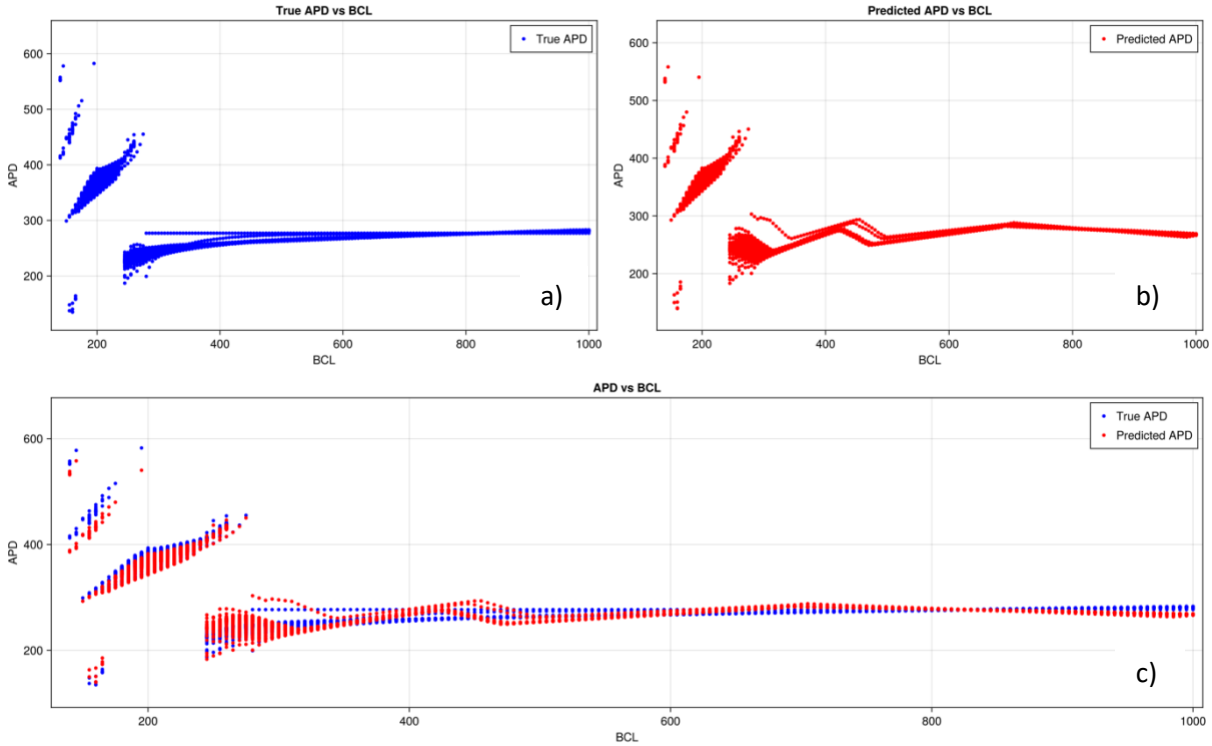

Fig. 14. APD vs BCL for the IMW model. a) Model simulated next APD, b) NN predicted next APD and c) both plots superimposed on each other.

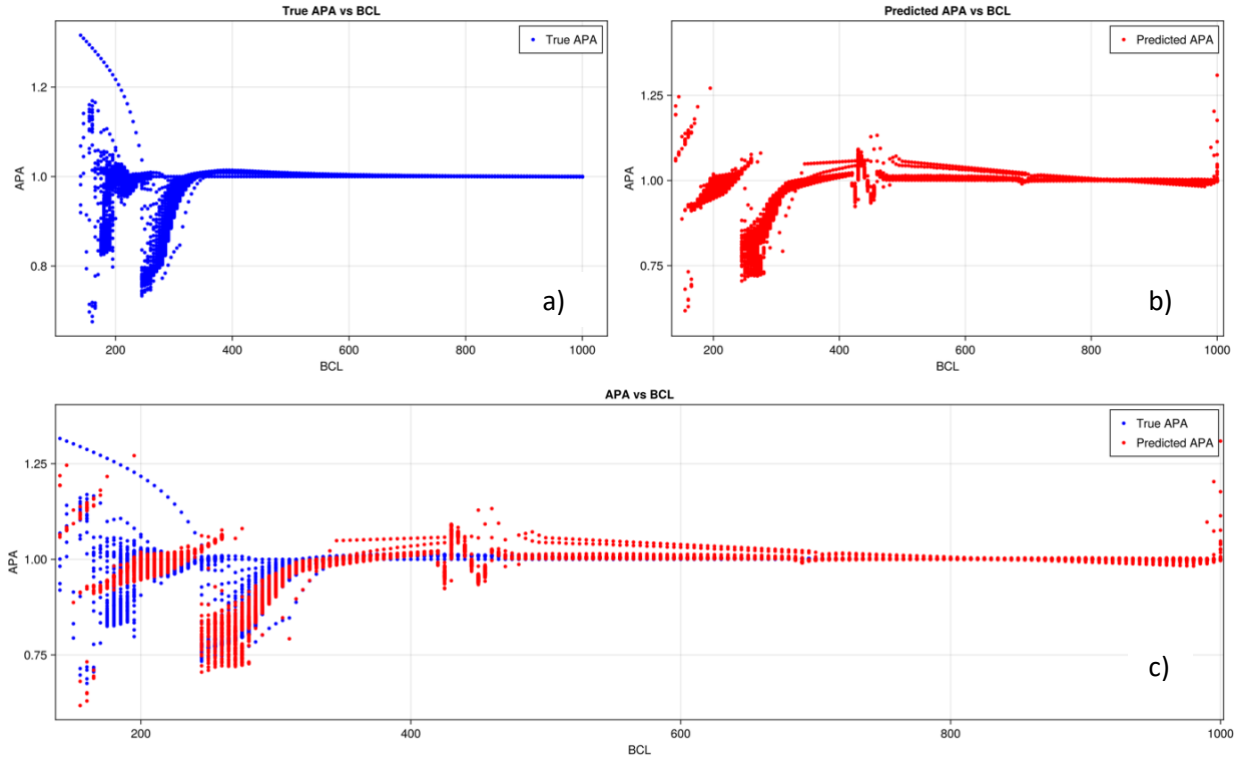

Fig. 15. APA vs BCL for the IMW model. a) Model simulated next APA, b) NN predicted next APA and c) both plots superimposed on each other.

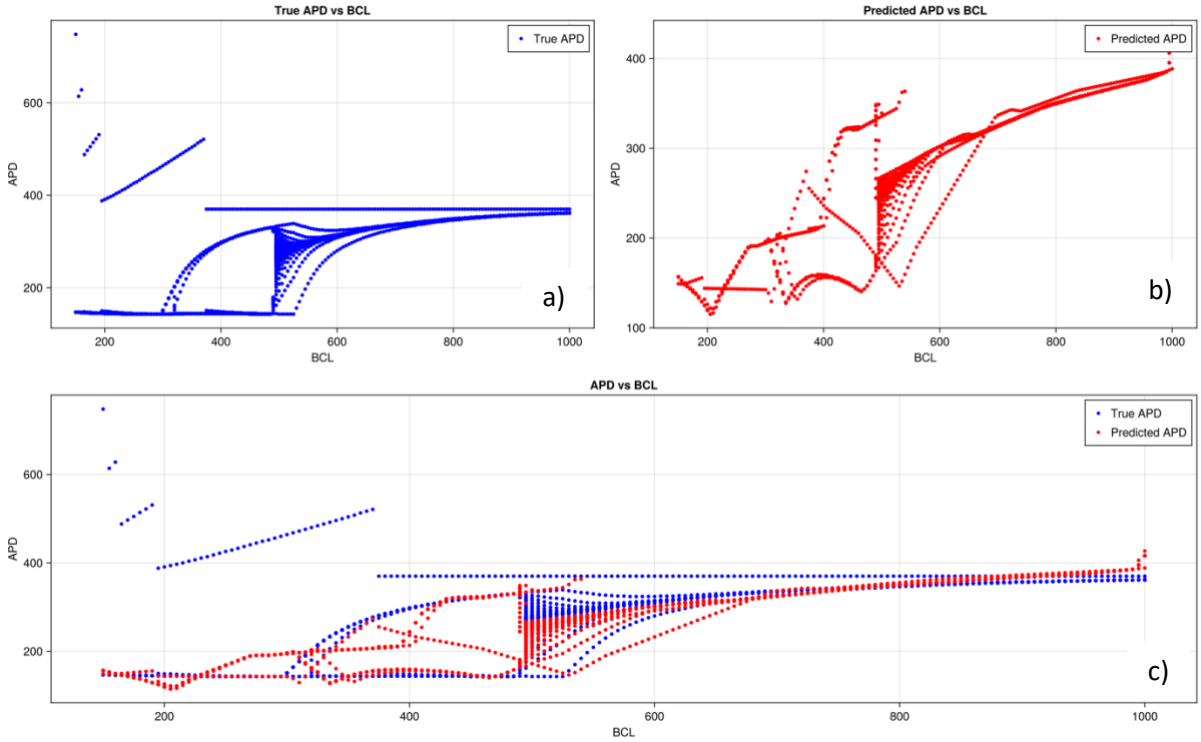

Fig. 16. APD vs BCL for the mMS model. a) Model simulated next APD, b) NN predicted next APD and c) both plots superimposed on each other.

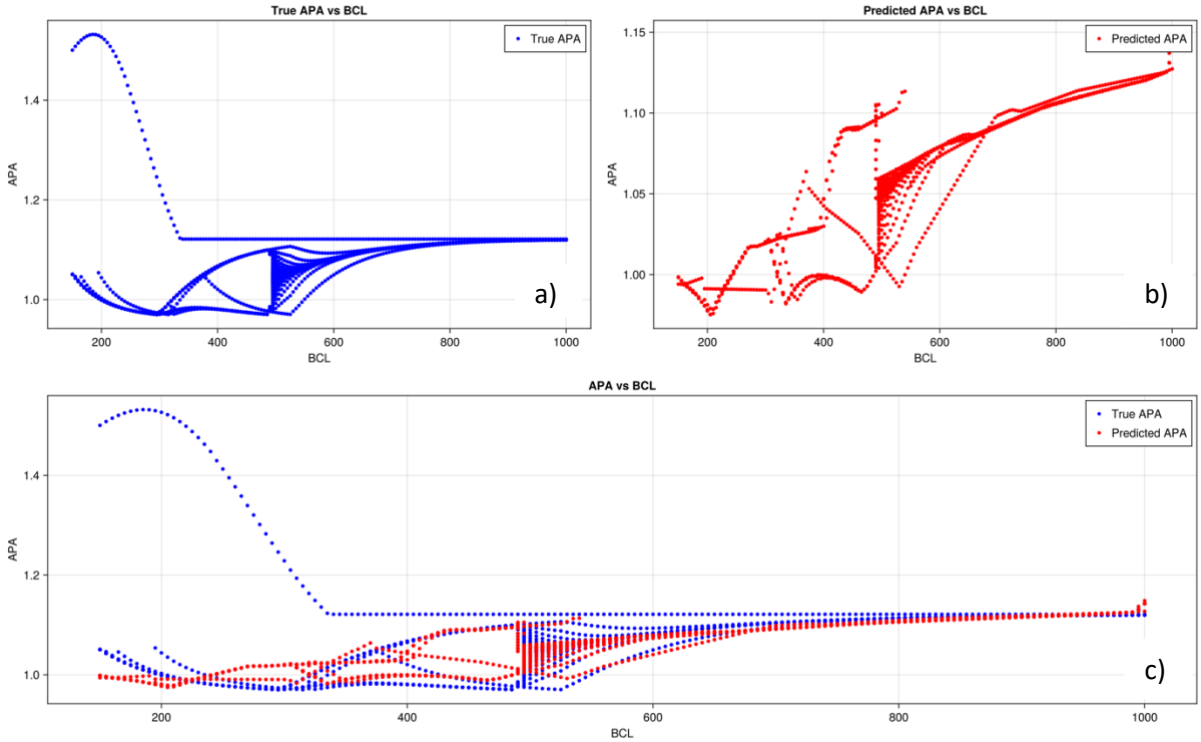

Fig. 17. APA vs BCL for the mMS model. a) Model simulated next APA, b) NN predicted next APA and c) both plots superimposed on each other.

The NNs implemented do not incorporate bifurcation structure and start with poor priors due to random initial weights, essentially making it a brute force approach. Despite all these limitations, they produce largely reasonable predictions, and importantly accurately identify the bifurcation locations and the onset of alternans in BCL to good accuracy.

5.5 Augmenting the Mitchell-Schaefer Model with Neural Networks

The MS model was augmented with a NN to simulate the AP of the IMW model. The resultant AP achieved shows strong agreement in the depolarisation phase but is less precise in the late plateau phase (Fig. 18). The plateau is too long, and the voltage does not return to baseline at the end of the BCL.

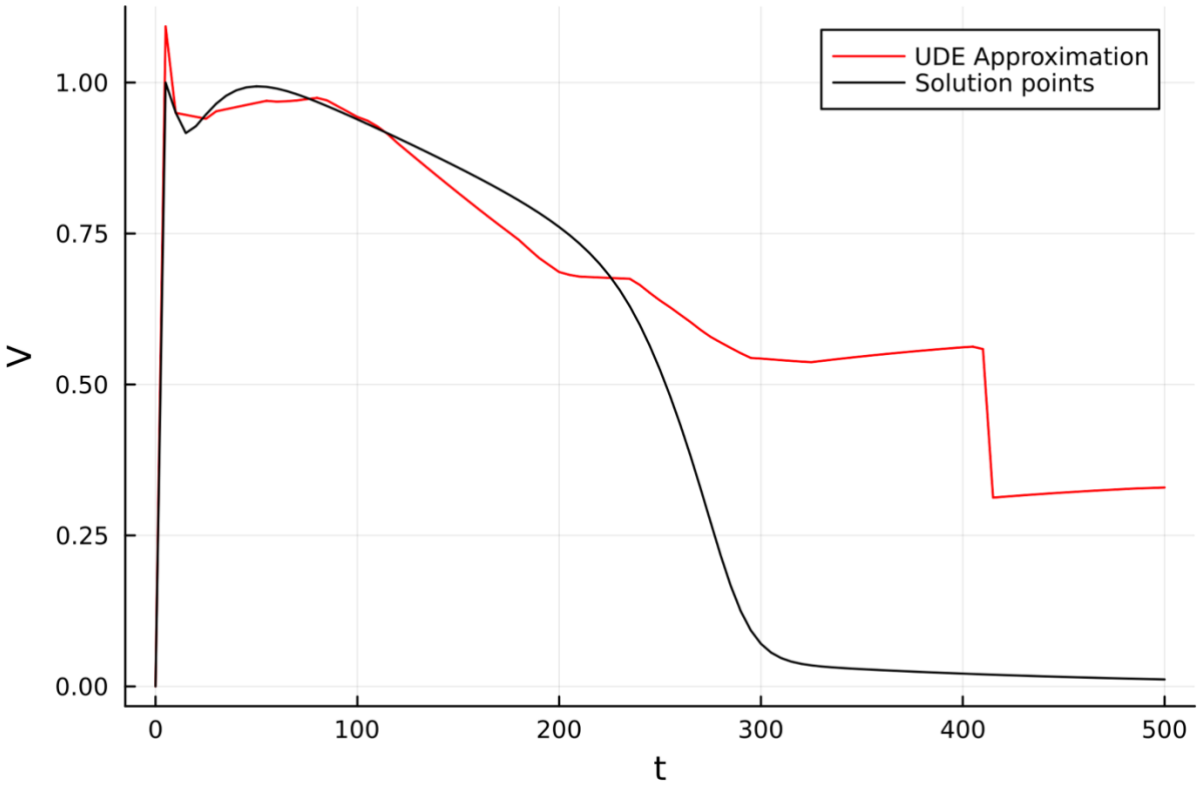

Fig. 18. Plot of the simulated NN predicted AP based on the MS model (shown in red) compared to the IMW model AP (shown in black) demonstrating good similarity in the depolarisation and early plateau phase of the AP but a less ideal fit in the late plateau and repolarisation phases.

This outcome may deviate from expectations due to a lack of expressivity in the NN and the accumulation of gradients over time. The current NN architecture only takes in the variables u and v as inputs and produces a single output, with relatively few hidden layers. Incorporating the derivatives of u and v (du, dv) as additional inputs may improve performance, as could increasing the

network depth. Given the accumulation of gradients over time, it is reasonable that the optimisation favours earlier features over later ones. If the primary deviation between the low- and high-fidelity models is in the depolarisation, then the current approach is likely to be successful given sufficient training data and time. Otherwise, this limitation might be mitigated by reformulating the loss function as a supremum over all data points. Typically, NNs are trained by minimising the average loss over all data points, as done above with the mean squared error (MSE) which is defined as:

$$L_{MSE} = \sum_{i=1}^{N} (y_i - u_i)^2 \tag{5}$$

where y_i is the observed data and u_i is the mMS model output at a time t_i. This formulation ensures that the model is optimised to perform well on average across the dataset. However, when later time steps are more difficult to predict, their errors may be underemphasised, since earlier and easier points dominate the average. The supremum is defined as the maximum error across all data points. So, the MSE loss could be changed to a supremum defined as:

$$L_{max} = \max_i |y_i - u_i| \tag{6}$$

This approach emphasises the worst-case error rather than the average performance, encouraging the network to reduce the largest discrepancies, thereby mitigating the NN's current tendency to fit early features well while neglecting errors at later time steps.

The two approaches can also be combined:

$$L = \sum_{i=1}^{N} (y_i - u_i)^2 + \lambda \max_i |y_i - u_i| \tag{7}$$

where λ is a weighting factor that controls how much the supremum contributes. This is known as L1 regularisation (Vidaurre et al., 2013).

In future work, it may be beneficial to bias the NN correction term I_{NN}, so that it remains strictly perturbative - small compared to the derivative of u from the MS model. This behaviour is consistent with our expectations, but it is not currently enforced. As I_{NN} is being introduced to a system where u already evolves according to dynamics of the MS model, the concern is that I_{NN} could dominate or distort dynamics, rather than just providing the needed refinement. Constraining I_{NN} ensures the MS model remains the primary driver of the dynamics not the NN, which could cause the system to behave in an unphysical way. This could be accomplished by introducing an amplitude factor which

controls the maximal value of I_{NN}, and regularising the contribution of this parameter to the training through a L1 regularisation term.

6 Conclusion

This project explored a simple and a complex model of cardiac electrical excitation by implementing and solving them in Julia. The ODEs in these models feature discontinuous right-hand sides, which pose non-trivial challenges for computation and analysis. Solving them numerically requires specialised techniques, such as adaptive solvers with event detection to accurately resolve the discontinuities. Even then, without sufficient information about the non-autonomous stimulus times, numerical algorithms may not consistently resolve discontinuities correctly, leading to potential inaccuracies.

Both models can simulate alternans making them important tools to investigate cardiac electrophysiological behaviour. The different models were compared by analysing maps that assess APA, APD and DI against BCLs. Maps produced show the development and cessation of alternans as the stimulus period is decreased. Tuning the MS model to better resemble the IMW model led to anomalous poor matching of the statistical properties of the maps. This result indicates that straight-forward optimisation of model parameters by comparing temporal traces can have deleterious effects on the overall statistical matching of important solution features in the presence of unknown non-autonomous stimulus forcing.

It was shown that NNs can predict the next APA and APD for both the MS and IMW models reasonably across a range of stimulus periods, which might be a valuable tool to explore the degree of memory needed in cardiac restitution. Yet the short-term matching of these sequences indicates that long-term prediction is still a challenge.

Augmenting low -fidelity cardiac models, such as the MS model, with strongly restricted NN currents may be a viable route to reproducing high-fidelity ODE model outputs with lesser computational cost. However, these initial investigations highlight several issues with relying on this process, not least the multi-stage tuning and training, and the eventual lack of good reproduction during later phases of the ODE solving. Furthermore, lack of a priori guidance on the structure and reliability of the NN for UODEs suggest that more systematic approaches which embed domain knowledge into the NN may be required for the technique to be useful and better approximate the shape of the AP. While uninformed NN methods can recover significant bifurcation features from discrete data, incorporating bifurcation normal forms into NNs is an ongoing effort for these models.

7 Acknowledgments

I would like to thank my supervisor Dr Christopher Marcotte for his help, guidance and kindness throughout the project without which I would not have been able to complete it. He taught me many new coding skills which will be very useful for my future career and helped me when I got stuck. I am also very grateful to the Durham Computer Science Department for providing a comfortable space to work and the Laidlaw Foundation for facilitating this opportunity.

8 Appendix: Abbreviations

ADAM	Adaptive Moment Estimation
AI	Artificial Intelligence
AP	Action Potential
APD	Action Potential Duration
APA	Action Potential Activation
BCL	Basic Cycle Length
DI	Diastolic Interval
ECG	Electrocardiogram
IMW	Iyer-Mazhari-Winslow
LBFGS	Limited-memory Broyden-Fletcher-Goldfarb-Shanno
ML	Machine Learning
mMS	Modified Mitchell-Schaeffer
MS	Mitchell-Schaeffer
MSE	Mean Squared Error
NN	Neural Network
ODE	Ordinary Differential Equation
ReLU	Rectified Linear Unit
RP	Resting Potential
SciML	Scientific Machine Learning
UDE	Universal Differential Equation
UODE	Universal Ordinary Differential Equation

9 References

'Anatomy of the heart' (2012) *Surgery (Oxford)*, vol. 30, no. 1, pp. 5-8.

Asghari- Targhi, A. (2017) 'Action potential duration alternans in mathematical models of excitable cells.' PhD thesis. University of Glasgow. Available at: https://theses.gla.ac.uk/8460/ (Accessed: 29 August 2025).

Amuzescu, B. *et al.* (2021) 'Evolution of mathematical models of Cardiomyocyte Electrophysiology', *Mathematical Biosciences*, 334. doi:10.1016/j.mbs.2021.108567.

Bezanson, J. *et al.* (2017) 'Julia: A fresh approach to numerical computing', *SIAM Review*, 59(1), pp. 65–98. doi:10.1137/141000671.

Cutler, M.J. and Rosenbaum, D.S. (2009) 'Explaining the clinical manifestations of T wave alternans in patients at risk for sudden cardiac death', *Heart Rhythm*, 6(3). doi:10.1016/j.hrthm.2008.10.007.

Danisch, S. and Krumbiegel, J. (2021) 'Makie.jl: Flexible high-performance data visualization for julia', *Journal of Open Source Software*, 6(65), p. 3349. doi:10.21105/joss.03349.

Denis, B. (2020) 'An overview of numerical and analytical methods for solving ordinary differential equations', *University of Kisubi*. doi:10.13140/RG.2.2.11758.64329

Dixit, V.K. and Rackauckas, C. (2023) 'Optimization.jl: A Unified Optimization Package (v3.12.1).' Available at: https://doi.org/10.5281/zenodo.7738525 (Accessed 14 September 2025).

Fu, D. (2015) 'Cardiac arrhythmias: Diagnosis, symptoms, and treatments', *Cell Biochemistry and Biophysics*, 73(2), pp. 291–296. doi:10.1007/s12013-015-0626-4.

Goldhaber, J.I. *et al.* (2005) 'Action potential duration restitution and alternans in rabbit ventricular myocytes', *Circulation Research*, 96(4), pp. 459–466. doi:10.1161/01.res.0000156891.66893.83.

Hodgkin, A. and Huxley, A. (1990) ‘A quantitative description of membrane current and its application to conduction and excitation in nerve’, *Bulletin of Mathematical Biology*, 52(1–2), pp. 25–71. doi:10.1016/s0092-8240(05)80004-7.

Iyer, V., Mazhari, R. and Winslow, R.L. (2004) ‘A computational model of the human left-ventricular epicardial myocyte’, *Biophysical Journal*, 87(3), pp. 1507–1525. doi:10.1529/biophysj.104.043299.

Jost, N. (2010) ‘Cardiac Action Potential and the Underlying Ion Currents’, pp11-26.

Kingma, D., Ba, J. (2014) ‘Adam: A method for stochastic optimization’ [Preprint]. doi:10.48550/arXiv.1412.6980.

Kulkarni, K. *et al.* (2019) ‘Cardiac alternans: Mechanisms and clinical utility in arrhythmia prevention’, *Journal of the American Heart Association*, 8(21). doi:10.1161/jaha.119.013750.

Lloyd, C.M. *et al.* (2008) ‘The CellML Model Repository’, *Bioinformatics*, 24(18), pp. 2122–2123. doi:10.1093/bioinformatics/btn390.

Luo, C.H. and Rudy, Y. (1991) ‘A model of the ventricular cardiac action potential. depolarization, repolarization, and their interaction.’, *Circulation Research*, 68(6), pp. 1501–1526. doi:10.1161/01.res.68.6.1501.

Mitchell, C. (2003) ‘A two-current model for the dynamics of cardiac membrane’, *Bulletin of Mathematical Biology*, 65(5), pp. 767–793. doi:10.1016/s0092-8240(03)00041-7.

‘Modeling and simulation case studies’ (2025) *JuliaHub*. Available at: https://info.juliahub.com/case-studies (Accessed: 31 August 2025).

Ngoma, D.V., Bourgault, Y. and Nkounkou, H. (2015) ‘Parameter Identification for a Non-differentiable Ionic Model used in Cardiac Electrophysiology’, *Applied Mathematical Sciences*, 9, pp. 7483–7507. doi:10.12988/ams.2015.510657.

Omale, D., Ojih, P.B. and Ogwo, M.O. (2014) 'Mathematical analysis of stiff and non-stiff initial value problems of ordinary differential equation using MATLAB', *International journal of scientific & engineering research*, *5*(9), pp.49-59.

Pal, A. (2025) 'Why we wrote lux?', *Lux.jl*. Available at: https://lux.csail.mit.edu/v0.5.0/introduction/overview/#why-use-lux-over-flux (Accessed: 29 August 2025).

Pigozzo, R.B., dos Santos, R.W. and Rocha, B.M. (2023) 'Sensitivity analysis of a cardiac electrophysiology model for the occurrence of electrical alternans', *Lecture Notes in Computer Science*, pp. 44–58. doi:10.1007/978-3-031-37105-9_4.

Rackauckas, C. (2022a) 'Automatically discover missing physics by embedding machine learning into differential equations', *Overview of Julia's SciML*. Available at: https://docs.sciml.ai/Overview/stable/showcase/missing_physics/ (Accessed: 01 September 2025).

Rackauckas, C. (2022b) 'SciML: Open source software for scientific machine learning', *SciML*. Available at: https://sciml.ai/ (Accessed: 01 September 2025).

Rackauckas, C. (2025) 'Detailed overview of the SciML software ecosystem'*, Overview of Julia's SciML*. Available at: https://docs.sciml.ai/Overview/dev/overview/ (Accessed: 03 August 2025).

Rackauckas, C. and Nie, Q. (2017) 'DifferentialEquations.jl – a performant and feature-rich ecosystem for solving differential equations in Julia', *Journal of Open Research Software*, 5(1), p. 15. doi:10.5334/jors.151.

Rackauckas, C. *et al.* (2020) 'Universal differential equations for scientific machine learning', *PNAS* [Preprint]. doi:10.21203/rs.3.rs-55125/v1.

Rackauckas, C. *et al.* (2025a) 'Common Solver Options (Solve Keyword Arguments)', *DifferentialEquations.jl*. Available at: https://docs.sciml.ai/DiffEqDocs/stable/basics/common_solver_opts/#solver_options (Accessed: 03 August 2025).

Rackauckas, C. *et al.* (2025b) 'ODE Solvers', *DifferentialEquations.jl*. Available at: https://docs.sciml.ai/DiffEqDocs/stable/solvers/ode_solve/ (Accessed: 03 August 2025).

Ravon, G. *et al.* (2019) 'Impact of the endocardium in a parameter optimization to solve the inverse problem of electrocardiography', *Frontiers in Physiology*, 9. doi:10.3389/fphys.2018.01946.

Sampson, M. and McGrath, A. (2015) 'Understanding the ECG. Part 1: Anatomy and physiology', *British Journal of Cardiac Nursing*, 10(11), pp. 548–554. doi:10.12968/bjca.2015.10.11.548.

Shampine, L.F. and Reichelt, M.W. (1997) 'The matlab ode suite', *SIAM Journal on Scientific Computing*, 18(1), pp. 1–22. doi:10.1137/s1064827594276424.

Singer, S. and Nelder, J. (2009) 'Nelder-mead algorithm'. *Scholarpedia*, *4*(7), p.2928.

Ten Tusscher, K. *et al.* (2006) 'Comparison of electrophysiological models for human ventricular cells and tissues', *Progress in Biophysics and Molecular Biology*, 90(1–3), pp. 326–345. doi:10.1016/j.pbiomolbio.2005.05.015.

Theodoridis, S. (2025) 'Chapter 18 - Neural networks and deep learning: part I', in *Machine Learning: From the Classics to Deep Networks, Transformers, and Diffusion Models*. Third Edition. London: Elsevier Academic Press.

Tse, G. *et al.* (2016) 'Cardiac dynamics: Alternans and arrhythmogenesis', *Journal of Arrhythmia*, 32(5), pp. 411–417. doi:10.1016/j.joa.2016.02.009.

Tsitouras, Ch. (2011) 'Runge–Kutta pairs of Order 5(4) satisfying only the first column simplifying assumption', *Computers & Mathematics with Applications*, 62(2), pp. 770–775. doi:10.1016/j.camwa.2011.06.002.

Urbán, J.F., Stefanou, P. and Pons, J.A. (2025) 'Unveiling the optimization process of physics informed neural networks: How accurate and Competitive Can PINNS be?', *Journal of Computational Physics*, 523, p. 113656. doi:10.1016/j.jcp.2024.113656.

Vidaurre, D., Bielza, C. and Larrañaga, P. (2013). 'A Survey of L1 Regression.' *International Statistical Review*, 81(3), pp.361–387. doi:10.1111/insr.12023.

Weiss, J.N. *et al.* (2000) 'Ventricular fibrillation: how do we stop the waves from breaking?', *Circulation Research*, 87(12), pp. 1103–1107. doi:10.1161/01.res.87.12.1103.

Winslow, R.L. *et al*. (2011) 'Integrative modeling of the cardiac ventricular myocyte. *Wiley Interdisciplinary Reviews: Systems Biology and Medicine*, *3*(4), pp.392-413.

Wilson, L.D. and Rosenbaum, D.S. (2007) 'Mechanisms of arrythmogenic cardiac alternans', *Europace*, 9(Supplement 6), pp. vi77–vi82. doi:10.1093/europace/eum210.

Wu, R. and Patwardhan, A. (2004) 'Restitution of action potential duration during sequential changes in diastolic intervals shows multimodal behavior', *Circulation Research*, 94(5), pp. 634–641. doi:10.1161/01.res.0000119322.87051.a9.