A Cellular Automata Model for Dynamics and Control of Cardiac Arrhythmias

Danny Gallenberger[1], Xiaopeng Zhao[2]
[1] Department of Mathematics and Computer Science, Carroll University, Waukesha, WI
[2] Department of Mechanical, Aerospace, and Biomedical Engineering, University of Tennessee, Knoxville, TN

Abstract:

With 325,000 deaths per year, sudden cardiac arrest is the leading cause of death in the United States. The major contributor to these cardiac deaths are arrhythmias in the heart. For many years, scientists have been studying the onset arrhythmias and any predictive behaviors that can help identify and treat the arrhythmia faster. In this study, we use a cellular automata (CA) model to illustrate the propagation of heart waves across its tissue. Through this model, we examine various scenarios and aim to stabilize and prevent cardiac arrhythmias by applying a control mechanism. Scientists have previously studied the use of feedback control to remedy alternans, but it is only effective for small tissue rather than large tissue. Instead, we examine the method of constant DI control to stabilize arrhythmias.

Contents

1. Introduction

Sudden cardiac arrest and other cardiac arrhythmias are a vital topic of interest among scientists because it is unknown how the onset of an arrhythmia can be predicted. If arrythmias can be identified before they happen, the high death rate in the United States due to sudden cardiac arrest can be reduced significantly. To examine how a heartbeat behaves in the heart, scientists have used two-variable partial differential equation models of cardiac tissue with a fast excitatory current and a slow repolarizing current. These methods, however, are computationally taxing, and it is difficult to adjust all the characteristic wave properties of the heart tissue. Cellular automata (CA) models have been favored over systems of PDEs to analyze wave propagate through cardiac tissue, especially since they can illustrate the states each cell goes through as time passes [2]. This study explores the electrophysiologic characteristics of the heart and how a CA model can be used to represent several scenarios in the heart, including normal conduction, scarred tissue, and alternans. The functions behind the structure of the implemented CA model will be explained in-depth. The CA model will aim to analyze the heart's behavior that occurs due to these abnormalities. Using this data, we will implement controls to stabilize the heart's electrical patterns. Considering the limitations of feedback control, we will explore the heart's rhythm when stimulated using a constant diastolic interval.

1.1. Electrophysiology of the Heart

The heart is made up of four chambers: the right atrium, the left atrium, the right ventricle, and the left ventricle. A heartbeat is essentially an electrical signal that is propagated through the heart's chambers. The signal originates in the sinus node, which is a group of pacemaker cells. The sinus node sends the signal from the right atrium through the atrioventricular (AV) node. It then goes through the His-Purkinje system to the ventricles. The heart contracts as the electrical signal passes through each chamber, allowing blood to flow to the rest of the body [1].

1.2. Cardiac Arrhythmias

Cardiac arrhythmias are essentially a disruption in the heart's normal pattern. There are many kinds of arrhythmias, and some are much more life-threatening than others. Variations of heart rates are categorized by bradycardias and tachycardias. Bradycardia is the condition of having a heart rate of less than 60 beats per minute. This is usually due to low sinus rhythm, causing a secondary, slower pacemaker to take over. Tachycardia is the condition in which one's heart rate is greater than 100 beats per minute and can typically lead to reduced blood flow to the body's organs and fatigue. In serious cases where blood flow is very low, tachycardia can lead to death [3].

Reentrant arrhythmias are arrhythmias where tissue is excited repetitively by waves circulating around an obstacle or freely in the tissue. Two common reentrant arrhythmias are atrial and ventricular fibrillation. Atrial fibrillation is characterized by an extremely rapid atrial heart rate of 350-600 bpm. Waves are continuously circulating in the atria, which compromise their ability to contract. Since the atria serve as filling chambers in the heart, their contraction is not necessary for function, so it is not immediately life-threatening. On the other hand, ventricular

fibrillation is life-threatening right away. Ventricular fibrillation is like atrial fibrillation, except many small waves circulate through the ventricles. This causes the ventricles to not be able to contract. Ventricular fibrillation is much more serious because blood cannot be pumped properly, so blood pressure drops to zero. Defibrillation is the only way to remedy this, and it must be done within minutes of onset [3].

Alternans is a non-reentrant arrhythmia in which action potential is conducted through the tissue, but the action potential durations alternate with every heartbeat. This can be an indicator of ventricular fibrillation in the future. Another non-reentrant arrhythmia is AV heart block. In AV heart block, the sinus node generates a normal rhythm, but not every impulse is sent through the atrioventricular node to the ventricles [3].

1.3. Cellular Automata

A cellular automata (CA) model is a two-dimensional grid of cells where each cell has various states. Each cell changes state based on predefined rules governing the CA model. Typically, these cells change state based on the states of their neighbors. For example, in Figure 1.1, an elementary cellular automata model is shown according to rule 30. Each cell can have a state of either 0 or 1, and the rule number determines which patterns are used to determine the next row of cells. Each group of three cells is examined, and based on the pattern of those cells, the cell directly underneath the middle cell is assigned the appropriate state. This process can continue forever. CA models are effective for modeling complex systems consisting of simple units and are much faster computationally than systems of PDEs [2]. In the next section, we look at how a cellular automata model can be used to simulate wave propagation.

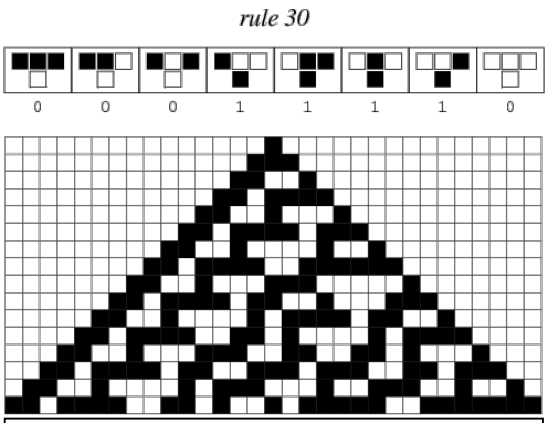


Figure 1.1 An elementary cellular automaton model of rule 30.

2. Methods

To simulate the various heart scenarios, a 50x50 cellular automata model was constructed. Each cell in the diagram represents an individual heart cell. The action potential, or voltage, of each cell can be characterized by four unique states: resting, exciting, absolute refractory, and relative refractory. Thresholds are set for the exciting phase and the refractory phase. In this study, we use an excitation threshold of 0.9 V and a refractory threshold of 0.1 V. When a cell is stimulated, it is in the exciting phase. During this time, the cell can excite any one of its neighboring cells. As time progresses, its voltage will gradually decrease. When the cell's action potential drops below 0.9 V, the cell enters the absolute refractory phase. The cell loses its ability to excite its neighbors, and the cell itself cannot be stimulated. When the action potential drops below 0.1 V, the cell enters the relative refractory phase, where the cell has some excitability

again. Once the action potential reaches 0 V, it has returned to the resting phase, and the cycle continues.

The cellular automata model for this study was coded in MATLAB, consisting of the main simulation function, a parameter function for each scenario, and various scripts for creating action potential plots, the cellular automata diagram, and electrocardiogram (ECG) plots. Functions are defined in the main simulation to determine how the waves will propagate. These functions are described in detail below (see Appendix B for code).

A couple of simulations that were run include some scarred tissue. Scarred tissue is represented by a group of cells in the cellular automata model whose action potential is constantly set to 0 V. These cells cannot be stimulated. All the scenarios run in the simulation are described in more depth in section 3.

2.1. Functions

In addition to the thresholds previously mentioned, many variables are initialized at the beginning of the simulation and used in all. We require a three-dimensional voltage array to store the action potentials of every cell at every timestep. We must also initialize two-dimensional arrays for DI and APD, which will store the intervals for each cell in the model. The initial DI for every cell is 100, which signifies that they are fully rested. An additional two-dimensional array called duration is used to track the time elapsed since the cell's last excitation. The basic cycle length (BCL) is initialized to determine the time interval in which the stimulus is applied, and a time interval is also predefined to determine how long the simulation will run. These variables are necessary for all the following functions.

2.1.1. Restitution

The restitution curve defines the relationship between the diastolic interval and the action potential duration (APD) at the n-th heartbeat. Specifically, the action potential duration at the n-th beat is dependent on the diastolic interval of the n-1-th beat. The relationship can be expressed as

$$f(D_n) = A_{max} - A_0 e^{-D_n/\tau}$$

where A_{max} is the maximum wave amplitude, A_0 is the initial wave amplitude, D_n is the diastolic interval at the n-th beat, and τ is an arbitrary constant. The longer the diastolic interval, the longer the APD is for the next beat. Conversely, a shorter diastolic interval will result in a shorter APD at the next beat. As $D_n \rightarrow \infty$, $f(D_n) \rightarrow A_{max}$. If $f'(D_n) > |1|$, then the cardiac dynamics are unstable and must be remedied. For this study, we have defined each variable to form the following equation. The graph of this equation is shown in Figure 2.1.

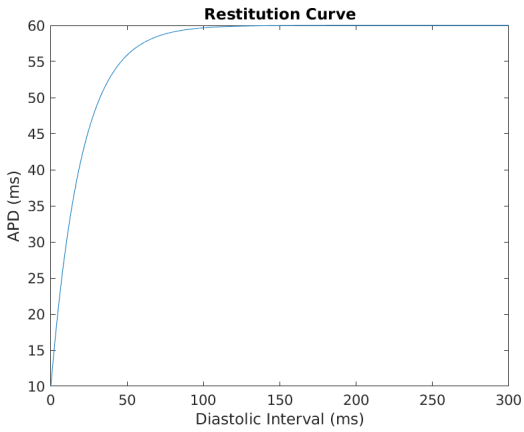


Figure 2.1 Restitution curve as modeled by $f(D_n) = 60 - 50e^{-D_n/20}$

$$f(D_n) = 60 - 50e^{-D_n/20}$$

2.1.2. Initial Wave Form

The formula for the initial wave form signifies what happens to a cell after it has been stimulated. It is defined as

$$f(A,t) = \frac{e^{-t/T(A)}}{\mathrm{c} + e^{-t/T(A)}}$$

$$T(A) = \frac{\mathrm{A}}{\ln(0.9) - \ln(0.1 * c)}$$

where $f(A,t)$ is the action potential based on some wave amplitude A and time t, and $T(A)$ is a time constant based on A. c is an arbitrary constant. At t = 0, the action potential is 1 V, and for each value of t thereafter, the action potential decreases. So, as $t \rightarrow \infty$, $f(A,t) \rightarrow 0$. The greater the amplitude is, the slower the cell depolarizes. For this study, each variable is defined such that

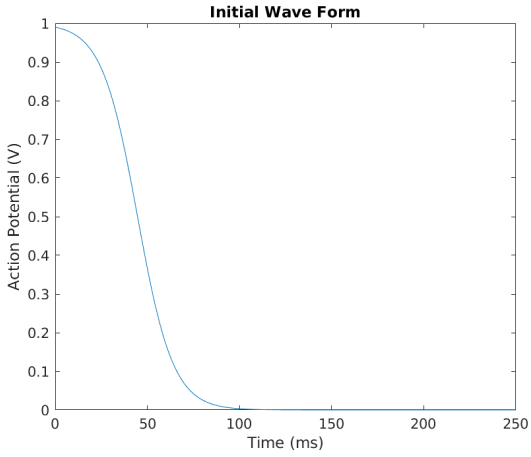


Figure 2.2 Initial wave form as modeled by

$$f(t) = \frac{e^{-t/9.7025}}{0.01 + e^{-t/9.7025}}$$

and $T(66) \approx 9.7025$

$$f(t) = \frac{e^{-t/9.7025}}{0.01 + e^{-t/9.7025}}$$

$$T(66) \approx 9.7025$$

2.1.3. Stimulation and Wave Propagation

At the beginning of the simulation, at t = 0, a 3x3 stimulus in the bottom left of the cardiac tissue will become excited. These cells act as the pacemaker cells in the heart. When a cell depolarizes, the diastolic interval (DI) of the previous heartbeat is determined (except for the initial stimulation) by taking the time elapsed since the cell's last stimulation and subtracting the precalculated action potential duration (APD) of the current cycle. The leftover time is the time the cell has been in the resting phase. The APD of the previous heartbeat is determined from the calculated DI. The action potential of the cell becomes 1 V, and the duration is reset upon stimulation.

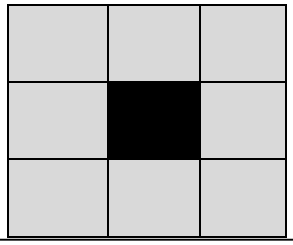


Figure 2.3 The black cell in the middle is the cell being evaluated to see if it becomes excited. The eight gray cells surrounding the black cell are checked to see if they are excited.

After the initial stimulation, time progresses. At each timestep, every heart cell is checked to see whether they will become excited. The propagation function comes into play at this point in the simulation. In this function, the first condition that is checked is whether the cell can become excited. A heart cell cannot be stimulated if it is above the refractory threshold. If the action potential of the cell is less than or equal to the refractory

threshold, then each neighboring cell of the cell being evaluated is checked to see if it is excited. The eight neighbors of a cell are illustrated in Figure 2.3. A bordering cell has five neighbors, and a corner cell only has three. If at least three neighboring cells have an action potential that is greater than the excitation threshold, then the cell being evaluated becomes excited. If the action potential of the cell being evaluated is greater than the refractory threshold, then the time elapsed since excitation increases, and the voltage progresses based on the APD and the current duration. The result is a propagating wave originated from the stimulus, traveling up and to the right throughout the tissue.

If scar cells exist in the tissue, those cells are set to 0 V at every time step. The entire process repeats for the next timestep. When the t value reaches the next stimulation time, the pacemaker cells become stimulated again, and the pattern continues (See Appendix A for the flowchart of the functionality).

2.2. Constant DI

To stabilize heart rhythms that are not normal, we will be implementing a control mechanism to remedy these. Constant DI was used as a control in this study. Instead of stimulating the heart based on a constant cycle length, the heart can be stimulated based on a constant DI interval, and not necessarily by a predefined time interval. We can examine the heart's electrical activity through an ECG diagram as shown in Figure 2.4 [3]. The ECG calculations are described below.

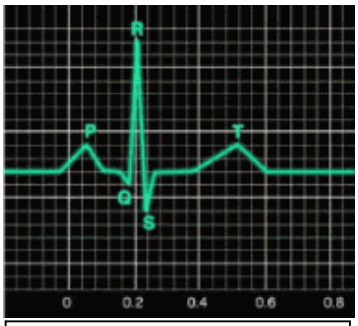


Figure 2.4 An electrocardiogram (ECG) with wave labels depicting the heart's electrical activity.

2.2.1. Electrocardiogram

The electrocardiogram (ECG) is a diagram that illustrates the electrical activity of the heart at each beat. It measures the voltage difference between two points outside the heart tissue, using electrodes placed on the surface of the body [4]. The ECG represents the voltage as it depends on time, and is calculated as follows

$$\mathrm{ECG} = \Phi_e(\mathrm{B}) - \Phi_e(\mathrm{A})$$

$$\Phi_e(x', y') = \int (-\nabla \mathrm{V_m}) \cdot \left(\nabla \frac{1}{\mathrm{r}}\right) dx\, dy$$

$$r = [(x - x')^2 + (y - y')^2]^{1/2}$$

where $\Phi_e(A)$ and $\Phi_e(B)$ are the transmembrane potentials at points (x',y') located outside the heart tissue. Points are not chosen inside the tissue because the denominator will result in zero. r is the distance between (x',y') and some point (x,y) in the heart tissue. $\nabla \mathrm{V_m}$ is the gradient of V and results in a vector of the slopes of the action potential at each heart cell. $\nabla \frac{1}{\mathrm{r}}$ is the gradient of the distances from the chosen point to each point in the tissue. We use ECG to identify abnormalities in the heart's rhythm and use the control mechanisms to resolve them.

3. Results

There are four scenarios that were simulated in this study:

- Normal conduction: the heart exhibits a normal heartbeat with a consistent wave propagation, as shown in Figure 3.1.

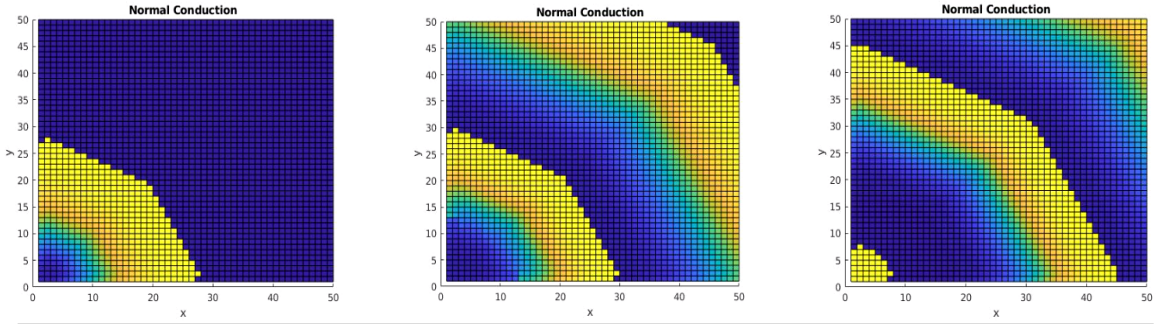


Figure 3.1 Snapshots of the cellular automata model for normal conduction of the heart.

- Normal conduction with scar: the heart exhibits a normal heartbeat with a consistent wave propagation, but there is scarred tissue for x ∈ [10,15] and y ∈ [15,20], except for (10,15), (10,20), (15,15), and (15,20). This is shown in Figure 3.2.

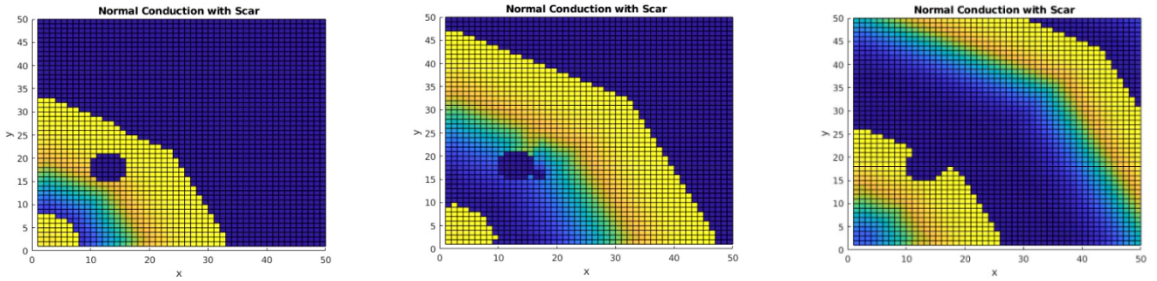


Figure 3.2 Snapshots of the cellular automata model for normal conduction of the heart with scar tissue.

- Spiral wave with scar: the heart exhibits a normal heartbeat with a consistent wave propagation, but there is scarred tissue for x ∈ [10,15] and y ∈ [5,10], except for (10,5), (10,10), (15,5), and (15,10). This is shown in Figure 3.3.

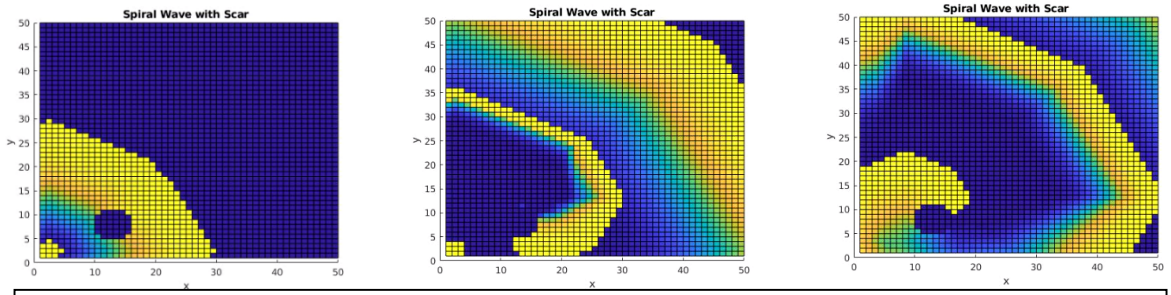


Figure 3.3 Snapshots of the cellular automata model for a spiral wave around scarred tissue in the heart.

- Alternans: the heart's rhythm alternates between long waves and short waves, as shown in Figure 3.4 (this scenario was simulated on a 25x25 grid).

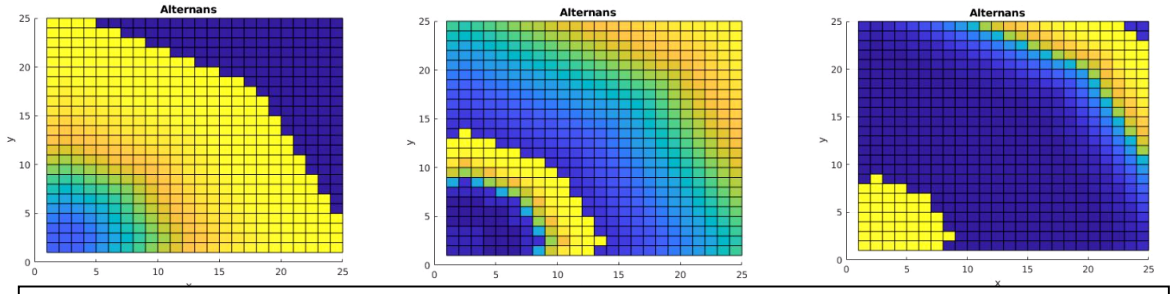
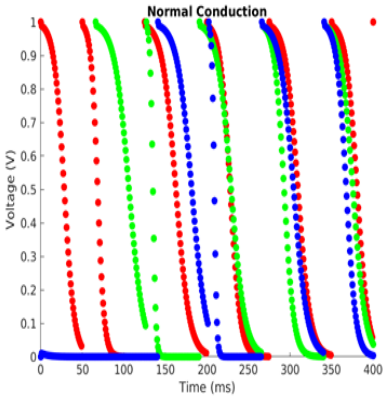
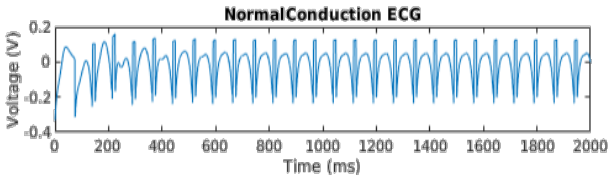
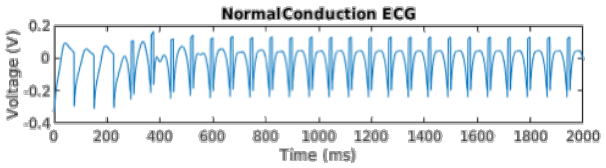
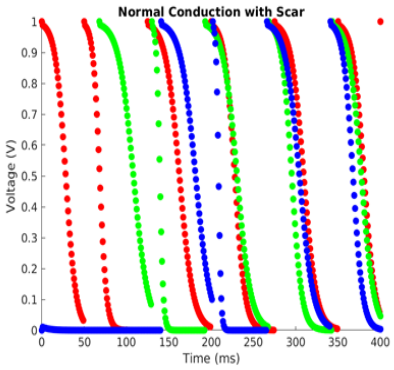
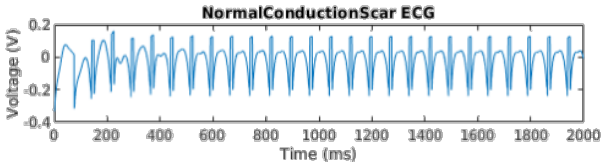
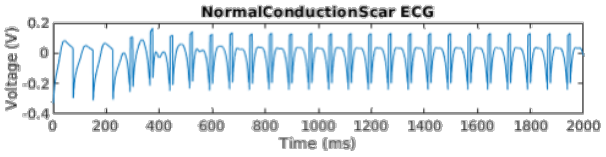
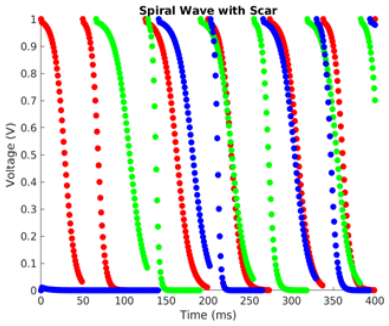
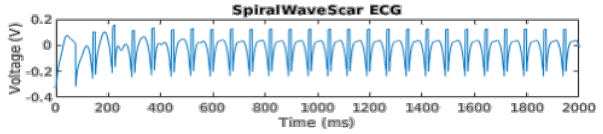
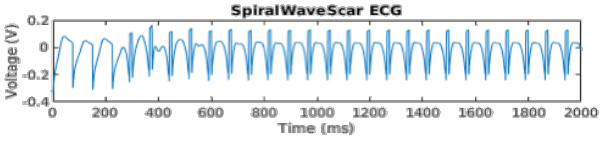


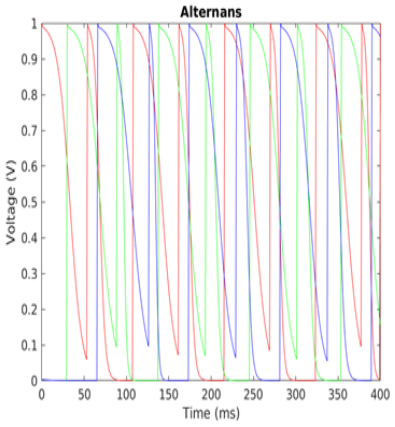
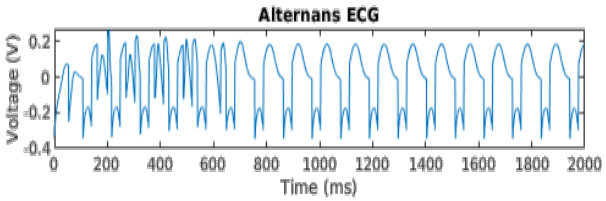
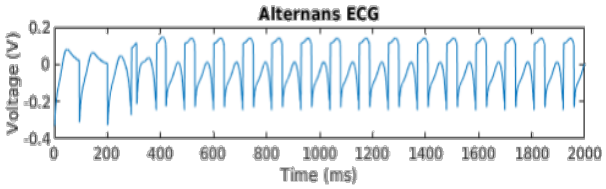
Figure 3.4 Snapshots of the cellular automata model for alternans in the heart.

3.1. No Control

The normal conduction scenario uses a constant BCL, thus illustrating how the heart typically behaves. Normal conduction with scar and spiral wave with scar each had a BCL of 75ms without control, and alternans had a BCL of 54ms without control. When a scar is placed in the tissue, the propagating waves must work around the inactive tissue. In Figure 3.2, the waves usually work around the scar just fine, but some heart cells around the scar spend more time in the refractory period. This causes the subsequent waves to work around those cells in addition to the scar cells, leading to irregular wave activity in the tissue. When a scar is placed closer to the edge of the tissue, like in Figure 3.3, the waves begin to spiral around the scar continuously without the help of any stimulated pacemaker cells. Alternans (Figure 3.4) are typically a precursor to ventricular fibrillation and must be remedied immediately [5]. Table 3.1 describes how single cells behave in heart tissue in each scenario, as well as the ECG diagram for each scenario. In each single cell plot, a 50x50 cellular automata model was used (25x25 for alternans). The red plot represents the heart cell at location (1,1), the green plot represents the heart cell at location (25,25) [(13,13) for alternans], and the blue plot represents the heart cells at location (50,50) [(25,25) for alternans]. For each ECG calculation, points (0,0) and (51,51) were used for a 50x50 model, and points (0,0) and (101,101) were used for a 100x100 model.

Table 3.1: Heart Simulations Without Control

	Single Cell	Tissue
Normal Conduction		50x50  100x100 
Normal Conduction with Scar		50x50  100x100 
Spiral Wave with Scar		50x50  100x100 

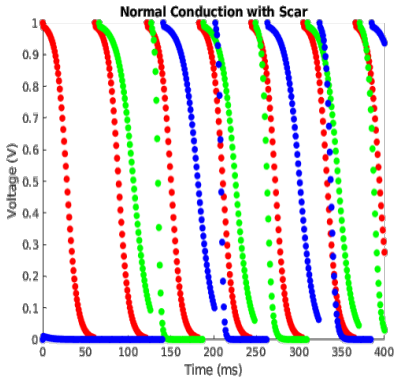
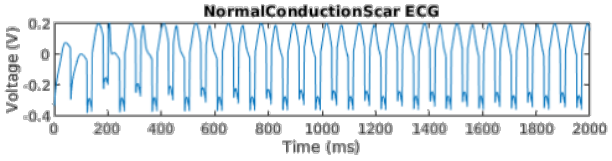
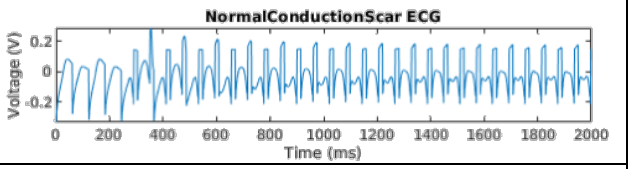
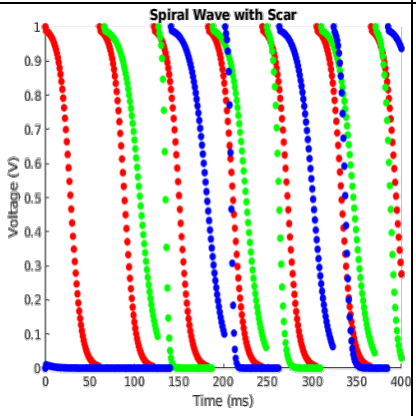
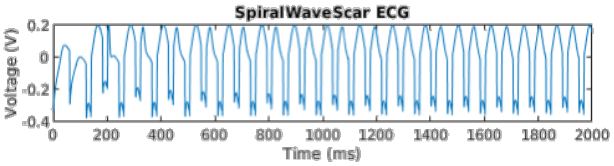
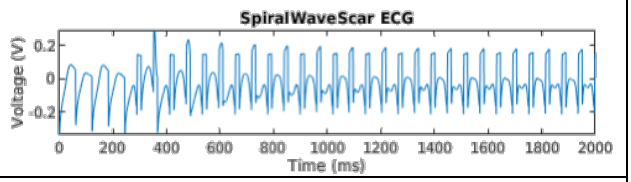
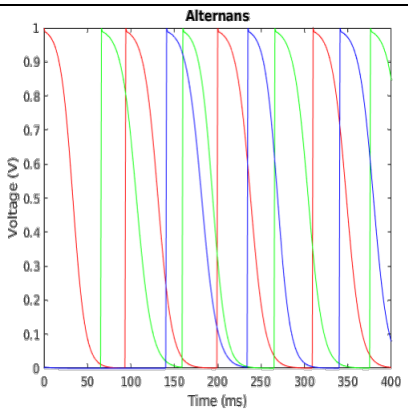
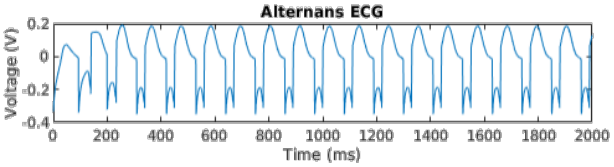
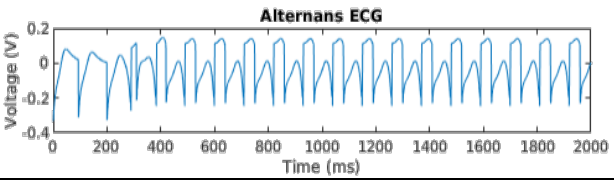
Alternans		50x50  100x100 

In the normal conduction, normal conduction with scar, and spiral wave with scar simulations, a stimulation time of 50 is defined despite the BCL of 75 to help regulate the DI and APD as time progresses. After the activity in the beginning settles down, the normal conduction ECG produces a consistent wave pattern. The diagrams of the scarred tissue are very similar to the normal conduction, but the scars still cause abnormalities previously mentioned. In the alternans ECG, the alternating beats can be seen in the first 600ms of the simulation. It can also be seen in the single cell plots. A possible error may exist in the alternans simulation at larger time intervals. As the APD increases with each passing beat, the pacemaker cells cannot excite at each defined time because they are still in the refractory period. This causes every other stimulus time to be skipped. Increasing the BCL could help resolve this problem.

3.2. Constant DI

Constant DI control was implemented for the normal conduction with scar, spiral wave with scar, and alternans scenarios to improve the heart's rhythm. Unlike the simulations without control, stimulation times are not defined at all in the simulations. They were determined based off the DI and APD of the pacemaker cells. When a pacemaker cell stimulates, the DI of the most recent heartbeat is calculated and used to calculate the APD of the next heartbeat. From this information, we calculate the next stimulation time by taking a desired BCL and subtracting the APD at a pacemaker cell. This gives us a DI target variable used to obtain the desired BCL. The stimulation times do not necessarily have equal intervals in this case. At the end of each simulation, the APD at location (1,1) was 56.5466, using the restitution function. The DI target was defined to be 19 for normal conduction with scar and spiral wave with scar, and -2 for alternans. The results of these simulations are shown in Table 3.2.

Table 3.2: Heart Simulations with Constant DI Control

	Single Cell	Tissue
Normal Conduction with Scar		50x50  100x100 
Spiral Wave with Scar		50x50  100x100 
Alternans		50x50  100x100 

In the beginning of each ECG diagram, the heart's electrical activity is not immediately stable. As the APD at the pacemaker cells become constant, the time intervals between each beat are also constant and remain constant for the rest of the simulation. In normal conduction with scar and spiral wave with scar, the intended pattern of the ECG waves is unclear in the beginning and becomes more evident as the time interval is extended. The pattern seems to emerge much later in the 100x100 tissue than in the 50x50 tissue. This could be because there is much more inactive tissue in the beginning of the simulation in a bigger model. It takes more time for the first stimulation to travel to the rest of the tissue. Constant DI control is much more effective when controlling alternans on 50x50 tissue. The ECG wave patterns are clear after only 300ms. As

shown in the single cell plot of alternans, the action potential of each cell no longer illustrates alternans and exemplifies a normal rhythm. However, alternans are not controlled when simulating on 100x100 tissue for reasons that are unclear, perhaps similar to the pitfalls of feedback control.

4. Conclusion

Based on the results from implementing constant DI, arrhythmias can be resolved by controlling the heart's electrical rhythms using DI and APD data. The cellular automata model is very effective when analyzing wave propagation in various heart scenarios. The location of a scar in the heart tissue is crucial when looking for arrhythmias. This was shown when comparing the snapshots of normal conduction with scar and spiral wave with scar. The electrical waves propagate much differently from each other in these two scenarios. Constant DI was the most effective in treating alternans. By modifying the MATLAB code to utilize the APD at a pacemaker cell and obtain a DI target, alternans were eliminated almost immediately in 50x50 tissue. Ultimately, cellular automata models serve as an excellent model to illustrate the electrical activity in the heart. They can help medical professionals identify arrhythmias and catch them before they become life-threatening, thus lowering the mortality rate due to sudden cardiac arrest.

This study can be extended further to simulate models in three dimensions. Since this study only examines two-dimensional models, these diagrams may not give a full illustration of how waves propagate throughout the heart. Additionally, processing large models in MATLAB takes a significant amount of space and time to run on a machine. Hence, running simulations on GPU to test much larger models could be of great benefit to researchers. Finally, there are many other scenarios that can be researched and potentially controlled. Simulations that were explored in this study and not controlled include alternans with an ectopic beat, normal conduction with an ectopic beat, and wave break. A second control mechanism to be explored is constant RT, where the heart is regulated by the time interval between the R and T waves in the ECG.

5. Acknowledgements

This project was sponsored by the National Science Foundation through Research Experience for Undergraduates (REU) award, with additional support from the Joint Institute of Computational Sciences at University of Tennessee Knoxville. In addition, the computing work was also performed on technical workstations donated by the BP High Performance Computing Team.

6. References

[1] Garcia, G. & Sheth, N. (n.d.). *Understanding Cardiac Arrhythmias through Cellular Automata*. University of Tennessee, Knoxville.

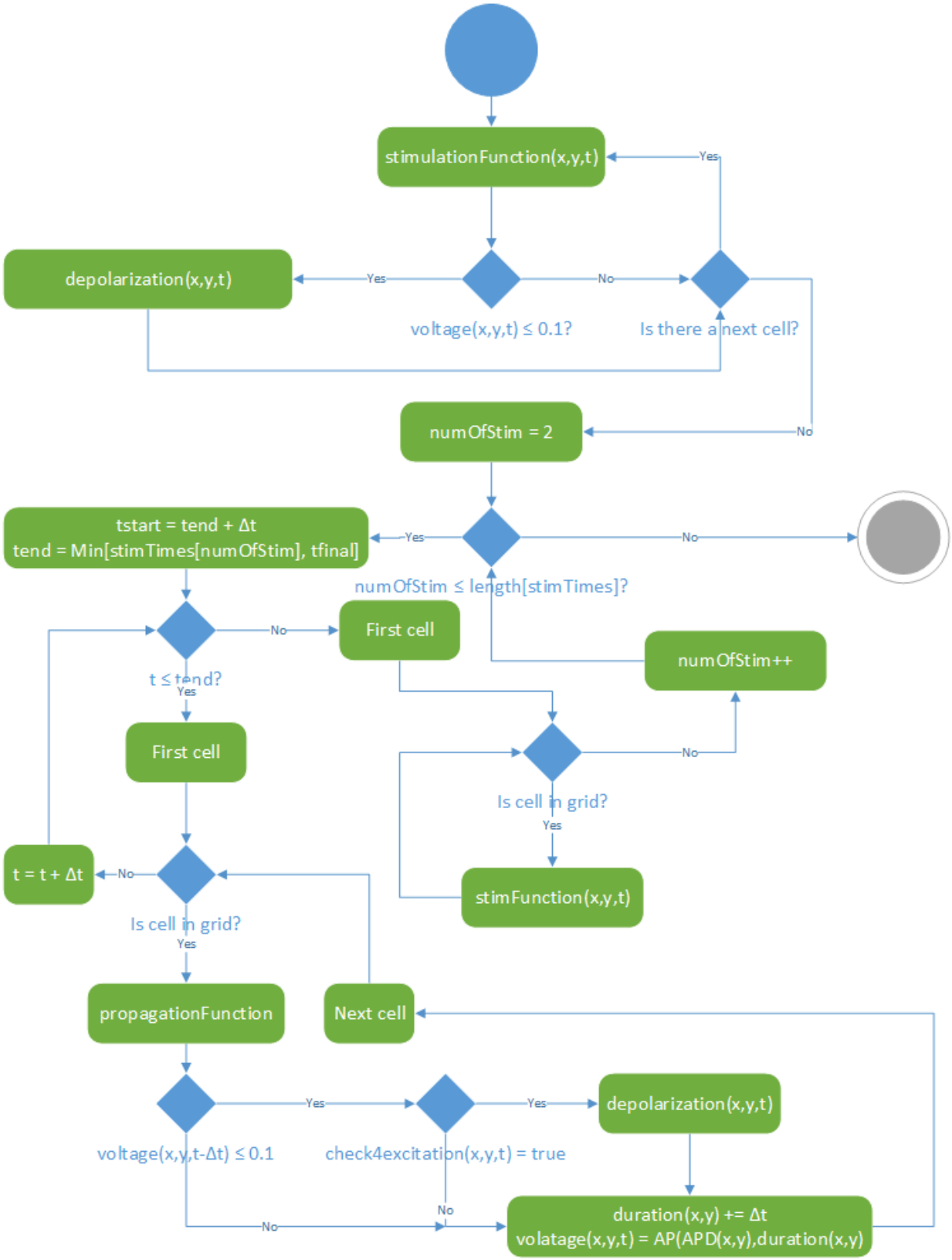
[2] Feldman, A.B., Chernyak, Y.B., & Cohen, R.J. (1999). *Cellular automata model of cardiac excitation waves.* Herzschrittmachertherapie und Elektrophysiologie*,* 10, 92-104.

[3] Flavio H Fenton et al. (2008). *Cardiac arrhythmia*. Scholarpedia, 3(7):1665.

[4] *Electrocardiogram – ECG*. myDr.

[5] Zlochiver, S., Johnson, C., and Tolkacheva, E. G. (2017). *Constant DI pacing suppresses cardiac alternans formation in numerical cable models*. American Institute of Physics. Chaos 27, 093903.

Appendix A


stimulationFunction(x,y,t)
Yes
depolarization(x,y,t)
Yes
No
voltage(x,y,t) ≤ 0.1?
Is there a next cell?
No
numOfStim = 2
tstart = tend + Δt
tend = Min[stimTimes[numOfStim], tfinal]
Yes
No
numOfStim ≤ length[stimTimes]?
No
First cell
numOfStim++
t ≤ tend?
Yes
First cell
No
Is cell in grid?
Yes
t = t + Δt
No
stimFunction(x,y,t)
Is cell in grid?
Yes
propagationFunction
Next cell
Yes
Yes
depolarization(x,y,t)
voltage(x,y,t-Δt) ≤ 0.1
check4excitation(x,y,t) = true
No
No
duration(x,y) += Δt
volatage(x,y,t) = AP(APD(x,y),duration(x,y)

Appendix B

Simulation Function

```
function[] = simulation()
%This function simulates wave propagation through heart tissue.

%Initialization of variables
propagationtime = 1;
refractThreshold = 0.1;
excitationThreshold = 0.9;
DI_target=-2; %Alternans
%DI_target = 19; %Normal conduction with scar and Spiral wave with scar

%Get parameters
[xdim,ydim,BCL,tfinal,stimTimes,dur,ectopicCells,scarCells,tl,fileName] = get_parameters();

t_step = 1;

%Initialize the tissue
voltage = zeros(xdim,ydim,tfinal+1);
DI=ones(xdim,ydim)*100;
APD=restitution(DI);
duration=ones(xdim,ydim)*dur;

%Apply the initial stimulation.
for x = 1:3
        for y = 1:3
            stimFunction(x,y,0);
        end
end

%Initialize start and end times.
tstart = 0;
tend = tstart+APD(1,1)+DI_target;

%Apply wave propagation.
for t = 1:t_step:tfinal
    for x = 1:xdim
        for y = 1:ydim
            propagationFunction(x,y,t);
        end
    end

    %Apply if the scenario includes ectopic cells.
    if(ectopicCells == 1)
        if(t == 130)
            for x = round(xdim/2)-1:round(xdim/2)+1
                for y = round(ydim/2)-1:round(ydim/2)+1
                    stimFunction(x,y,t);
                end
            end
        end
    end
```

```
    %Apply if the scenario includes scar cells.
    if(scarCells == 1)
        for x = 5:10
            for y = 10:15
                if(((x == 5 || x == 10) && (y == 10 || y == 15)))
                    continue
                end
                voltage(x,y,t+1) = 0;
            end
        end
    end

    %Apply the stimulation at tend.
    if t >= tend
        for x = 1:3
            for y = 1:3
                stimFunction(x,y,t);
            end
        end

        %Update time interval.
        tstart=t;
        tend = tstart+APD(1,1)+DI_target;
    end
end

%Save workspace
save(fileName)

%==== Stimulation & Propagation Functions ====
    function [] = stimFunction(x,y,t)
    %Check to see if the cell will stimulate.
        if(voltage(x,y,t+1) <= refractThreshold)
            depolarization(x,y,t);
        end
    end

    function [] = depolarization(x,y,t)
    %Stimulation the cell
        DI(x,y) = duration(x,y) - APD(x,y);
        APD(x,y) = restitution(DI(x,y));
        voltage(x,y,t+1) = 1;
        duration(x,y) = 0;
    end
```

```
    function [] = propagationFunction(x,y,t)
    %Wave propagation
        if(voltage(x,y,t+1-t_step) <= refractThreshold)
            b = check4excitation(x,y,t); %check neighboring cells
            if(b == 1)
                depolarization(x,y,t);
            else
                evolution(x,y,t);
            end
        else
            evolution(x,y,t);
        end
    end

    function [b] = check4excitation(x,y,t)
    %Count the neighboring cells that are excited.
        val = 0;
        for i=x-1:x+1
            for j=y-1:y+1
                %Skip cells outside of the tissue
                %Skip the cell being evaluated
                if(i == 0 || j == 0)
                    continue;
                end
                if(i > xdim || j > ydim)
                    continue
                end
                if(i == x && j == y)
                    continue
                end
                if(voltage(i,j,t+1-propagationtime) > excitationThreshold)
                    val = val + 1;
                end
            end
        end

        %If at least 3 neighboring cells are excited, the evaluated cell
        %becomes excited.
        if(val >= 3)
            b = 1;
        else
            b = 0;
        end
    end

    function [] = evolution(x,y,t)
        %Update the duration and action potential of the cell.
        duration(x,y) = duration(x,y) + t_step;
        voltage(x,y,t+1) = initial_wave_form(APD(x,y), duration(x,y));
    end
end
```

Parameter Function

```
function [xdim,ydim,BCL,tfinal,stimTimes,dur,ectopicCells,scarCells,tl,fileName] = get_parameters()
%Sets parameters for the specified scenario

scenario = 1;

switch scenario
%Normal Conduction
    case 1
        xdim = 50;
        ydim = 50;

        BCL = 75;
        tfinal = 2000;
        stimTimes = [0,50];
        dur = restitution(100) + 20;

        ectopicCells = 0; %enter 1 to simulate with ectopic cells
        scarCells = 0; %enter 1 for simulate with scarred cells

        tl = 'Normal Conduction';
        fileName = 'NormalConduction';

%Normal Conduction with Scar
    case 2
        xdim = 50;
        ydim = 50;

        BCL = 75;
        tfinal = 2000;
        stimTimes = [0,50];
        dur = restitution(100) + 20;

        ectopicCells = 0; %enter 1 to simulate with ectopic cells
        scarCells = 1; %enter 1 for simulate with scarred cells

        tl = 'Normal Conduction with Scar';
        fileName = 'NormalConductionScar';

%Spiral Wave with Scar
    case 3
        xdim = 50;
        ydim = 50;

        BCL = 75;
        tfinal = 2000;
        stimTimes = [0,50];
        dur = restitution(100) + 20;

        ectopicCells = 0; %enter 1 to simulate with ectopic cells
        scarCells = 1; %enter 1 for simulate with scarred cells

        tl = 'Spiral Wave with Scar';
        fileName = 'SpiralWaveScar';
```

```
%Alternans
    case 4
        xdim = 50;
        ydim = 50;

        BCL = 54;
        tfinal = 2000;
        stimTimes = [0];
        dur = restitution(100) + 30;

        ectopicCells = 0; %enter 1 to simulate with ectopic cells
        scarCells = 0; %enter 1 for simulate with scarred cells

        tl = 'Alternans';
        fileName = 'Alternans';

%Alternans (ectopic)
    case 5
        xdim = 50;
        ydim = 50;

        BCL = 53;
        tfinal = 2000;
        stimTimes = [0];
        dur = restitution(100) + 20;

        ectopicCells = 1; %enter 1 to simulate with ectopic cells
        scarCells = 0; %enter 1 for simulate with scarred cells

        tl = 'Alternans (ectopic)';
        fileName = 'AlternansEctopic';

%Normal (ectopic)
    case 6
        xdim = 50;
        ydim = 50;

        BCL = 75;
        tfinal = 2000;
        stimTimes = [0];
        dur = restitution(100) + 20;

        ectopicCells = 1; %enter 1 to simulate with ectopic cells
        scarCells = 0; %enter 1 for simulate with scarred cells

        tl = 'Normal Conduction (ectopic)';
        fileName = 'NormalEctopic';
```

```
%Wave Break
    case 7
        xdim = 50;
        ydim = 50;

        BCL = 53;
        tfinal = 2000;
        stimTimes = [0];
        dur = restitution(100) + 20;

        ectopicCells = 0; %enter 1 to simulate with ectopic cells
        scarCells = 0; %enter 1 for simulate with scarred cells

        tl = 'Wave Break';
        fileName = 'WaveBreak';

    otherwise
        return;

end

end
```

Restitution Function

```
function [y] = restitution(Dn)
%This function graphs the action potential duration (APD) as a function
%of the diastolic interval.

Amax = 60;
A0 = 50;
tau = 20;

f = @(Dn) Amax - A0.*exp(-Dn./tau);
y = f(Dn);


end
```

Initial Wave Form Function

```
function [ap] = initial_wave_form(A,t)
%This function gives the action potential of a heart cell for a value of t

c = 0.01;

tc = A / (log(0.9)-log(0.1*c));
f = @(t) exp(-t/tc) / (c+exp(-t/tc));

ap = f(t);

end
```

ECG Script

```
clear
load NormalConduction
%load NormalConductionScar
%load SpiralWaveScar
%load Alternans
%load NormalConductionNoControl
%load NormalConductionScarNoControl
%load SpiralWaveScarNoControl
%load AlternansNoControl

%pointA=[0,0]; pointB=[51,51];
pointA=[0,0]; pointB=[101,101];

%ECG
ecg=(1:tfinal+1)*0;
for t = 1:tfinal+1
    potentialA = phi(pointA(1),pointA(2),xdim,ydim,t,voltage);
    potentialB = phi(pointB(1),pointB(2),xdim,ydim,t,voltage);
    ecg(t) = potentialB - potentialA;
end

figure();
subplot(3,1,1);
plot(0:tfinal, real(ecg));
xlabel(' Time (ms) ')
ylabel(' Voltage (V) ')
title([fileName ' ECG'])
```

Φ_e Function

```
function [potential] = phi(x_p,y_p,xdim,ydim,t,V)
%Calculates the potential phi at a specific location (x_p,y_p). The point
%(x_p,y_p) must be outside the domain.

gradV = zeros(xdim,ydim);
gradrr = zeros(xdim,ydim);

%gradient of 1/r
for x = 1:xdim
    for y = 1:ydim
        denom = ((x-x_p)^2+(y-y_p)^2)^(3/2);
        gradrr(x,y) = ((x_p-x)+(y_p-y)*1i)/denom;
    end
end

%gradient of V at interior points
for x = 2:xdim-1
    for y = 2:ydim-1
        gradV(x,y) = (V(x+1,y,t)-V(x-1,y,t))/2+(V(x,y+1,t)-V(x,y-1,t))/2*1i;
    end
end

%gradient of V at the left edge and the right edge
for y = 2:ydim-1
    gradV(1,y) = 0+(V(1,y+1,t)-V(1,y-1,t))/2*1i;
    gradV(xdim,y) = 0+(V(xdim,y+1,t)-V(xdim,y-1,t))/2*1i;
end

%gradient of V at the top boundary and bottom boundary
for x = 2:xdim-1
    gradV(x,1) = (V(x+1,1,t)-V(x-1,1,t))/2+0*1i;
    gradV(x,ydim) = (V(x+1,ydim,t)-V(x-1,ydim,t))/2+0*1i;
end

%corner points
gradV(1,1) = 0;
gradV(1,ydim) = 0;
gradV(xdim,1) = 0;
gradV(xdim,ydim) = 0;

dtmp = dot(gradV,gradrr);
potential = sum(sum(dtmp));

end
```