Alex Thomason
BME 506 – Cardiac Electrical Events
Final Project

Energy Saving Algorithm – Finding Chronaxie from Patient Myocardial Capture Data

Table of Contents

Repository Link: https://github.com/AlexThomason/bme_506_final_project/tree/README

I. Definitions

- **Cardiac pacemaker** – A medical device that generates electrical impulses delivered by electrodes to cause the heart muscle chambers to contract and therefore pump blood; by doing so this device replaces and/or regulates the function of the electrical conduction system of the heart.
- **Capture** – Depolarization of the atria and/or ventricles by an electrical stimulus delivered by an artificial pacemaker
- **Threshold** – The minimum amount of electrical energy needed to consistently capture the heart outside of the heart's refractory period
- **Pulse Duration** – Length of time [ms] that the pacing pulse occurs
- **Pulse Strength** - Amplitude of Volts [V] or Amps [mA] of a pacing pulse. In this paper, pulse strength will be referred to in Volts [V].
- **Output** – stated as ___V @ ___ms
- **Safety Margin** – Programming the pacemaker at a high enough output (amplitude and duration) above the threshold to make sure that we capture the heart consistently. Having a high enough safety margin is important to consistently capture the heart if the patient's threshold fluctuates.
    - Industry standard = either 2x threshold voltage or 3x pulse width
- **Rheobase** - The lowest stimulus current that continues to capture the heart when the stimulus duration is made very long.
- **Chronaxie** - The pulse duration at twice the rheobase
- **Pacing Energy** – Energy consumption of a pacing pulse
- **Most energy Efficient Pacing** – Occurs at chronaxie
- **Strength Duration Curve** – Curve that relates pacing threshold to pulse duration. The plot of a strength duration curve is pulse amplitude (either voltage [V] or current) vs pulse duration [ms]. In this paper, the strength-duration curve will be Pulse Amplitude [V] vs Pulse Duration [ms].

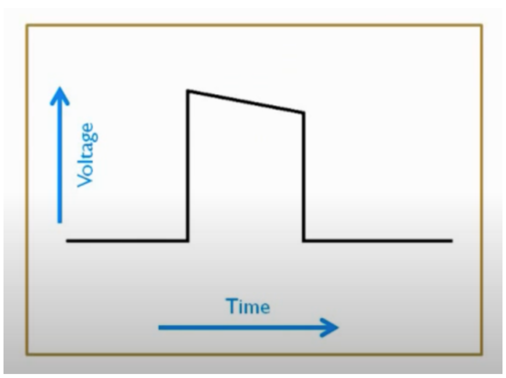


Figure 1: Shape of a stimulating pulse

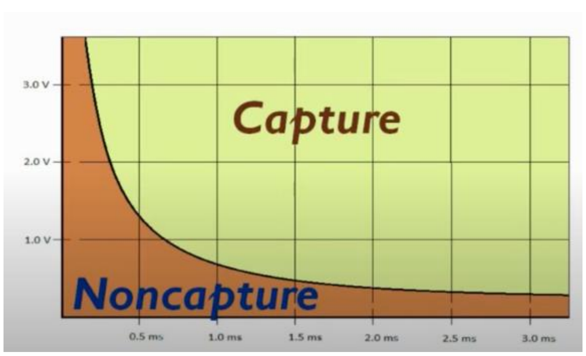


Figure 2: Strength Duration Curve

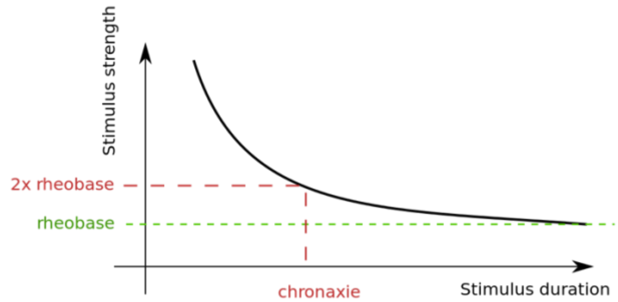


Figure 3. Rheobase and Chronaxie on Strength-Duration Curve

II. Relevant Equations

- E = Energy
- I = Current
- R = Resistance or impedance
  - Normal lead impedance ranges from about 300-1500 ohms
- V = Voltage
- t = time
- I = V/R
- $E = \frac{V^2 * t}{R}$
- Strength-Duration Curve Formula: V = Vr * (1 + t_c/t)
  - V = threshold Voltage at pulse duration t
  - Vr = rheobase Voltage
  - t_c = chronaxie pulse duration
  - t = stimulation pulse duration

III. Introduction

When pacing the heart, the pacemaker must capture the heart 100% of the time. The pacemaker should be programmed to have a pacing output that is high enough to stimulate the myocardial tissue consistently but low enough to avoid unnecessarily draining battery life. Because a pacemaker battery has a limited charge, battery life management is a critical consideration in the design to maximize device longevity. Hardware developments (including high-energy-density battery, high impedance, and low threshold leads) and software improvements (for pacing with lowest feasible energy) have been developed to enhance the lifetime of the pacemaker.

A pacemaker battery life is dependent on programmable parameters, principally pulse amplitude and pulse duration. High factory default settings cause an excessive current drain. The physician is responsible for setting the voltage and duration to the appropriate values directly after implantation. Additionally, patients need to have post-implantation pacing checks to ensure those values are reasonable and safe. The frequency of these checks ranges between 3 months to a year (depending on the device) [2]. Physicians must instill a safety margin of voltage or pulse duration when programming pacemakers to ensure consistent myocardial tissue capture. The current industry standard for safety margin is two times the voltage threshold or three times the pulse-width threshold. An important equation to consider is the energy equation:

$$E = \frac{V^2 * t}{R}$$

Because the voltage term is squared, if the voltage doubles, then energy quadruples. If time doubles, energy doubles. Minimum pulsing energy occurs at chronaxie. In a study of 229 pacemaker patients, most patients had a pulse duration set at 0.45 ms or 0.5 ms (common factory defaults) from implantation even though the pacing voltage was set to be double the threshold voltage [1]. The pulse durations were too long and wasteful of energy.

The pacing threshold exhibits significant inter-individual variations. After implantation, the pacemaker must be programmed with various parameters to fit each patient. Even in an individual, the pacing threshold may vary over time because of spontaneous threshold rise after implantation, micro-dislodgment of the pacemaker lead, diurnal changes, and changes secondary to drugs or myocardial ischemia [4]. These varying pacing thresholds may raise safety risks due to narrowing pacing stimulation safety margins. Conversely, unnecessarily high pacing output shortens battery life. The ability to automatically track threshold and to adjust the pacing outputs accordingly will maximize patient safety and minimize battery drain for pacing [4]. Additionally, optimizing threshold detection algorithms could help reduce battery size and the time it takes for pacemaker programming. Several manufacturers have developed algorithms for gathering threshold data, which is used either on a beat-by-beat basis to ensure a passed response or intermittently to adjust output parameters. A summary of existing capture management algorithms is summarized below [5]:

- St. Jude/Pacemaker Autocapture
    - <u>Type:</u> determines evoked response (capture) beat-by-beat
    - <u>Backup pulse</u>: 4.5 V delivered immediately if no capture is detected
    - <u>Threshold search starts when:</u> two consecutive backup pulses detected
    - <u>Increase voltage:</u> 0.125 V steps until two consecutive captures are detected
    - <u>Safety margin:</u> 0.3 V added
- Boston Scientific Automatic Capture
    - <u>Type:</u> Beat-to-beat verification of myocardial capture based on ventricular evoked response
    - <u>Backup pulse</u>: Backup pulse of (measured threshold + 1.5 V) delivered 100 ms after initial stimulus if no capture is detected
    - <u>Threshold search starts when</u>: No capture for 2/4 beats
    - <u>Safety Margin:</u> 0.5 V above measured threshold (ventricular voltage)
- Medtronic Ventricular Capture Management
    - <u>Type:</u> Intermittent – activated every 15 mins for 42 days
    - <u>Measure Rheobase:</u> determined at 1 ms by amplitude decrement until loss of capture and then by amplitude increment until capture is confirmed.
    - <u>Measure Chronaxie:</u> determined by doubling the programmed amplitude and decreasing the pulse width (and sequentially increasing the amplitude to capture)
    - Recommended setting based off rheobase and chronaxie

These three methods find capture by increasing voltage, but every time the voltage is under the threshold, the pacemaker generates a backup pulse of higher voltage. It's a waste of energy to continuously initiate those backup pulses, especially if the threshold detection is on a beat-by-beat basis. St. Jude and Boston Scientific algorithms do not find the optimal stimulus duration, only the threshold for the stimulus amplitude. Optimizing both stimulus duration and amplitude will save energy. My goal is to develop an algorithm that improves/combines these three approaches by efficiently finding the chronaxie (the point of minimum pulsing energy) and dynamically adjusting pulse amplitude and duration.

IV. Purpose of Energy Saving Algorithm

The idea of this project stems from the desire to save as much energy as possible in a pacemaker. Pacemakers usually assume that the stimulation amplitude and duration operate around chronaxie (the point of minimum energy usage). I want to write an algorithm to find the minimum energy needed to stimulate a patient's myocardial tissue. This algorithm would be used periodically by a pacemaker within the patient to automate finding the chronaxie point of stimulation and setting the pulse duration and amplitude of an appropriate safety margin. By using this algorithm, the pacemaker would have a way to find the value of minimum energy usage, thus extending battery life.

V. Code Structure

This section covers the code structure of the git repository. Each module (file) has modular functions along with docstrings to describe inputs, outputs, and what it does. The following figure shows the flow of the repository code modules:

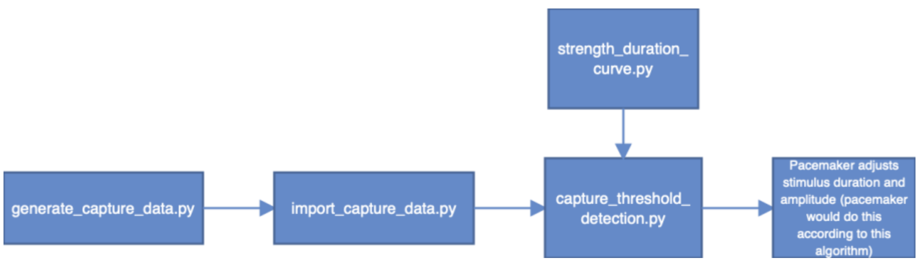


Figure 4. Flow Diagram - Repository Modules

requirements.txt

Includes necessary packages for the code to run properly. To run the code, create a virtual environment with python by typing 'python -m venv <VirtualEnvironmentName>' where <VirtualEnvironmentName> is the desired name of the virtual environment. For example python -m venv MyVenv makes a virtual environment with the name MyVenv. For Linux (MacOS), activate the virtual environment by typing 'source <VirtualEnvironmentName>/bin/activate' in the same folder in which you created the virtual environment. After activating the virtual environment, type pip install -r requirements.txt to install the packages. Refer to [6] for more detailed instructions on virtual environments.

generate_capture_data.py

Some "fake" data will be generated to simulate when the patient's myocardial tissue would be captured. This data will have 3 columns:

1. Stimulation Duration (constant)
2. Stimulation Amplitude
3. Whether or not the tissue is "captured" or depolarized

Stimulus Duration [ms]	Stimulation Amplitude [V]	Capture Status (0 = Not Captured 1 = Captured)
0.5	3	1
0.5	2.8	1
0.5	2.6	1
0.5	2.4	0
0.5	2.2	0
0.5	2	0

Figure 5 Example Captured Data

This data is generated from experimental data points. The purpose of generating this data is to simulate capture threshold data of myocardial tissue for a patient. The capture threshold detection algorithm will use this data to try and find capture points that match the experimental data points.

import_capture_data.py

Imports the data generated by generate_capture_data.py, parses each column into an array, and converts the values into floats. Each column in the data in the file will be converted into a list of float values that will be used in the capture threshold detection algorithm.

capture_threshold_detection.py

The algorithm iterates the stimulation amplitude [V] downward at a constant stimulation duration to find the “capture (threshold) points” on the strength-duration curve. It will iterate through voltage values until it finds the capture threshold (within 5% accuracy above the actual threshold value). Figure 7 goes into more detail about this process. After the algorithm finds the capture threshold value point, it will repeat that process for four different duration values. After the capture threshold value for each stimulus duration, the data points are stored in an array and used to optimize a strength-duration curve trendline. Rheobase and chronaxie are retrieved from that trendline equation. After the chronaxie value is found, the pacemaker stimulation duration and amplitude settings will be altered to save energy.

The following figure is a functional diagram of how my threshold capture algorithm works:

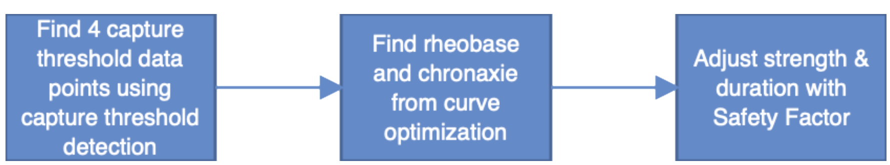


Figure 6. Flow Diagram - Adjusting Stimulation Strength & Duration

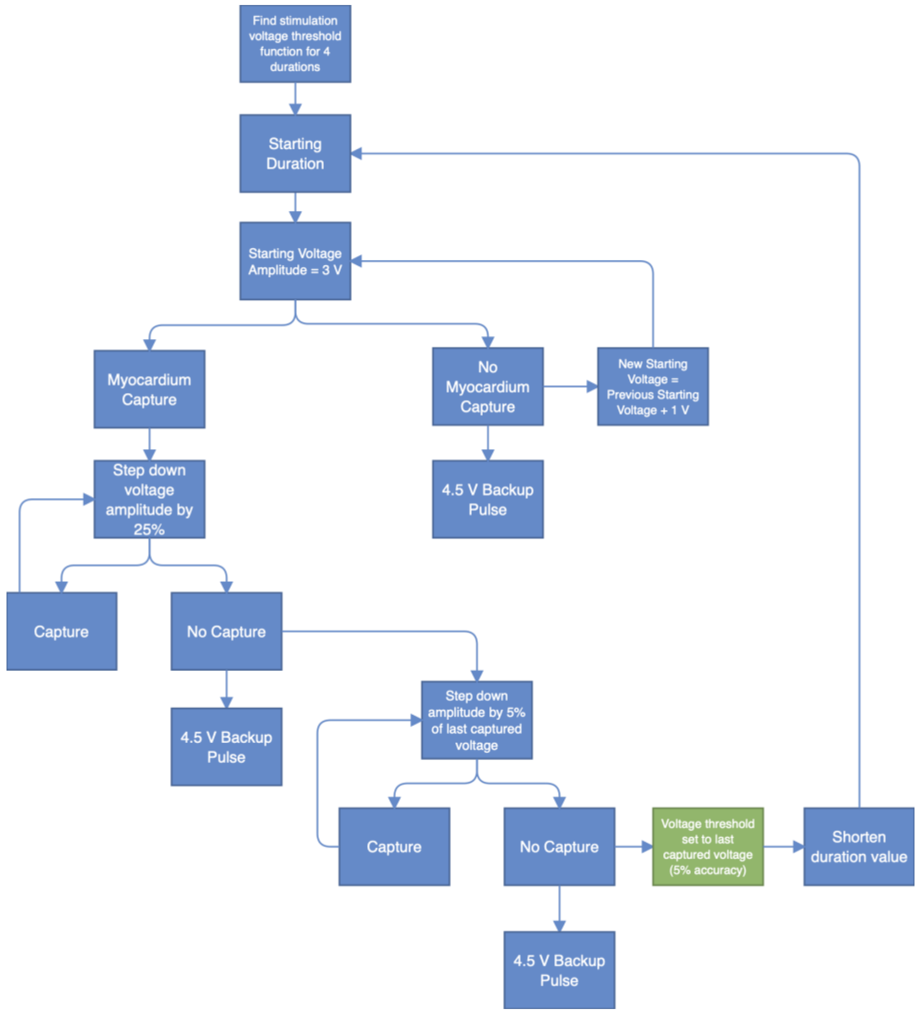


Figure 7. Functional Decomposition - Threshold Capture Detection Algorithm

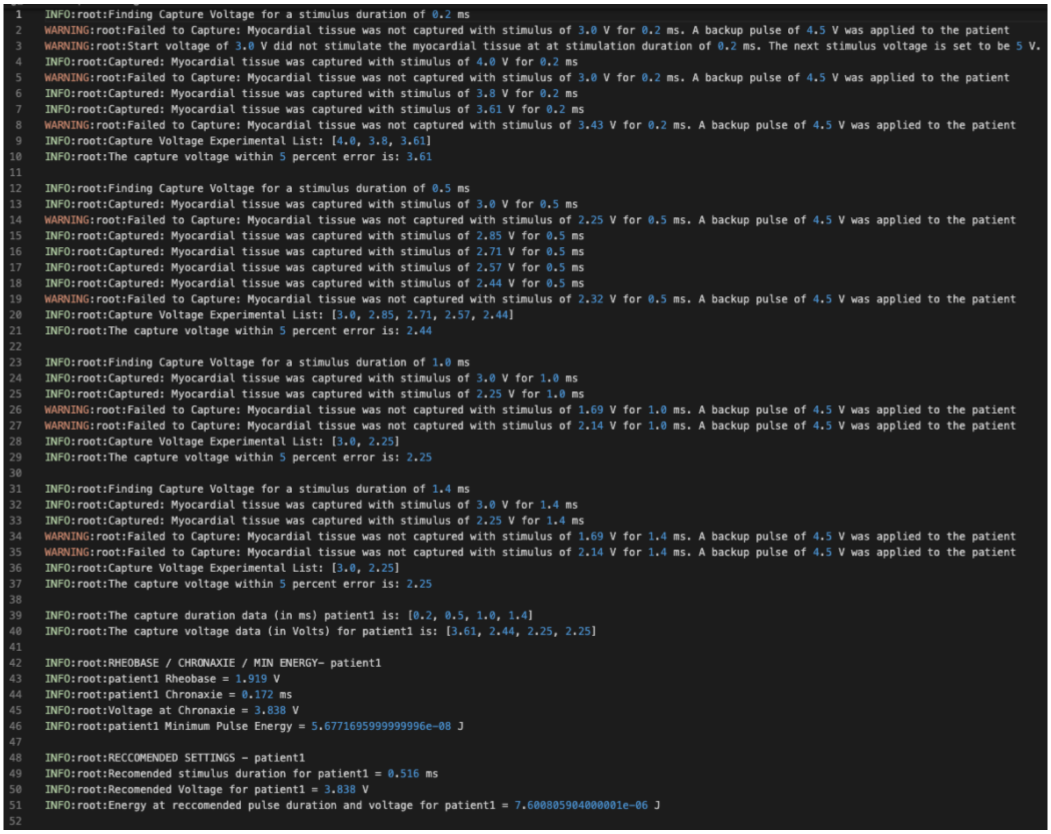
strength_duration_curve.py

This file contains functions to perform calculations on the threshold data gathered from the capture_threshold_detection.py. First, a trendline is found by optimizing the strength-duration curve formula: $V = V_r * (1 + t_c/t)$, where $V_r = rheobase$ and $t_c = chronaxie$. Rheobase and chronaxie are optimized to fit the capture threshold data. Plots of the experimental data, optimized curve, and energy are displayed for reference. Energy of the stimulation at chronaxie is also calculated for reference.

patient.log file

Displays logging information about the capture threshold process, and results for a given set of data for a patient. This information includes:

- When a stimulus voltage was captured
- When a stimulus voltage was not captured
- Capture threshold data for duration and voltage
- Rheobase and chronaxie
- Recommended stimulation duration & amplitude settings



```
INFO:root:Finding Capture Voltage for a stimulus duration of 0.2 ms
WARNING:root:Failed to Capture: Myocardial tissue was not captured with stimulus of 3.0 V for 0.2 ms. A backup pulse of 4.5 V was applied to the patient
WARNING:root:Start voltage of 3.0 V did not stimulate the myocardial tissue at at stimulation duration of 0.2 ms. The next stimulus voltage is set to be 5 V.
INFO:root:Captured: Myocardial tissue was captured with stimulus of 4.0 V for 0.2 ms
WARNING:root:Failed to Capture: Myocardial tissue was not captured with stimulus of 3.0 V for 0.2 ms. A backup pulse of 4.5 V was applied to the patient
INFO:root:Captured: Myocardial tissue was captured with stimulus of 3.8 V for 0.2 ms
INFO:root:Captured: Myocardial tissue was captured with stimulus of 3.61 V for 0.2 ms
WARNING:root:Failed to Capture: Myocardial tissue was not captured with stimulus of 3.43 V for 0.2 ms. A backup pulse of 4.5 V was applied to the patient
INFO:root:Capture Voltage Experimental List: [4.0, 3.8, 3.61]
INFO:root:The capture voltage within 5 percent error is: 3.61

INFO:root:Finding Capture Voltage for a stimulus duration of 0.5 ms
INFO:root:Captured: Myocardial tissue was captured with stimulus of 3.0 V for 0.5 ms
WARNING:root:Failed to Capture: Myocardial tissue was not captured with stimulus of 2.25 V for 0.5 ms. A backup pulse of 4.5 V was applied to the patient
INFO:root:Captured: Myocardial tissue was captured with stimulus of 2.85 V for 0.5 ms
INFO:root:Captured: Myocardial tissue was captured with stimulus of 2.71 V for 0.5 ms
INFO:root:Captured: Myocardial tissue was captured with stimulus of 2.57 V for 0.5 ms
INFO:root:Captured: Myocardial tissue was captured with stimulus of 2.44 V for 0.5 ms
WARNING:root:Failed to Capture: Myocardial tissue was not captured with stimulus of 2.32 V for 0.5 ms. A backup pulse of 4.5 V was applied to the patient
INFO:root:Capture Voltage Experimental List: [3.0, 2.85, 2.71, 2.57, 2.44]
INFO:root:The capture voltage within 5 percent error is: 2.44

INFO:root:Finding Capture Voltage for a stimulus duration of 1.0 ms
INFO:root:Captured: Myocardial tissue was captured with stimulus of 3.0 V for 1.0 ms
INFO:root:Captured: Myocardial tissue was captured with stimulus of 2.25 V for 1.0 ms
WARNING:root:Failed to Capture: Myocardial tissue was not captured with stimulus of 1.69 V for 1.0 ms. A backup pulse of 4.5 V was applied to the patient
WARNING:root:Failed to Capture: Myocardial tissue was not captured with stimulus of 2.14 V for 1.0 ms. A backup pulse of 4.5 V was applied to the patient
INFO:root:Capture Voltage Experimental List: [3.0, 2.25]
INFO:root:The capture voltage within 5 percent error is: 2.25

INFO:root:Finding Capture Voltage for a stimulus duration of 1.4 ms
INFO:root:Captured: Myocardial tissue was captured with stimulus of 3.0 V for 1.4 ms
INFO:root:Captured: Myocardial tissue was captured with stimulus of 2.25 V for 1.4 ms
WARNING:root:Failed to Capture: Myocardial tissue was not captured with stimulus of 1.69 V for 1.4 ms. A backup pulse of 4.5 V was applied to the patient
WARNING:root:Failed to Capture: Myocardial tissue was not captured with stimulus of 2.14 V for 1.4 ms. A backup pulse of 4.5 V was applied to the patient
INFO:root:Capture Voltage Experimental List: [3.0, 2.25]
INFO:root:The capture voltage within 5 percent error is: 2.25

INFO:root:The capture duration data (in ms) patient1 is: [0.2, 0.5, 1.0, 1.4]
INFO:root:The capture voltage data (in Volts) for patient1 is: [3.61, 2.44, 2.25, 2.25]

INFO:root:RHEOBASE / CHRONAXIE / MIN ENERGY- patient1
INFO:root:patient1 Rheobase = 1.919 V
INFO:root:patient1 Chronaxie = 0.172 ms
INFO:root:Voltage at Chronaxie = 3.838 V
INFO:root:patient1 Minimum Pulse Energy = 5.6771695999999996e-08 J

INFO:root:RECCOMENDED SETTINGS - patient1
INFO:root:Recomended stimulus duration for patient1 = 0.516 ms
INFO:root:Recomended Voltage for patient1 = 3.838 V
INFO:root:Energy at reccomended pulse duration and voltage for patient1 = 7.600805904000001e-06 J
```

Figure 8. Patient log that tracks activity of the pacemaker capture threshold detection algorithm

VI. Experimental Strength-Duration Curve Data

Data Set 1

- Pulse Duration [ms]
  - duration = [0.1, 0.2, 0.3, 0.4, 0.5, 1, 1.4]
- Original Voltage Amplitude [V]
  - voltage = [5, 3.5, 2.8, 2.6, 2.4, 2.2, 2.2]
- Original Rheobase = 1.86 V
- Original Chronaxie = 0.17 ms

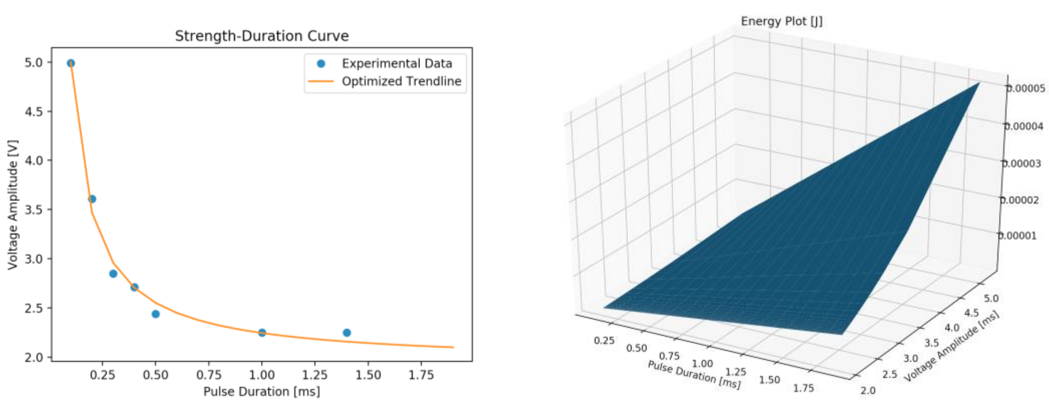


Figure 9. Strength-Duration Curve & Energy Plot of Data Set 1. Notice how the energy is lowest at chronaxie

Data Set 2 [6]

- Pulse Duration [ms]
    - duration = [0.3, 0.5, 0.8, 1, 1.5]
- Original Voltage Amplitude [V]
    - voltage = [2.25, 1.55, 1.25, 1.15, 0.92]
    - Original Rheobase = 0.63 V
    - Original Chronaxie = 0.77 ms

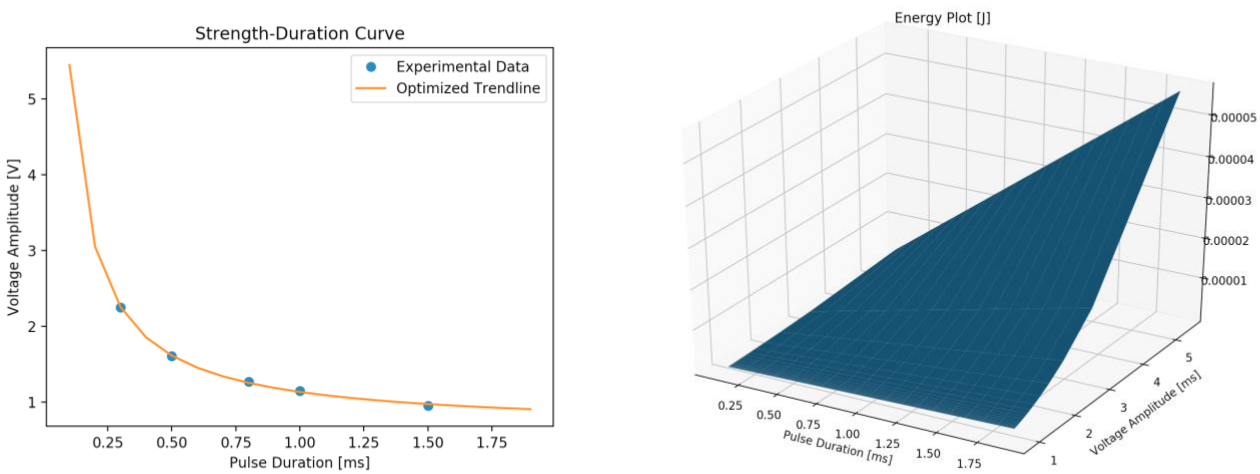


Figure 10. Strength-Duration Curve & Energy Plot of Data Set 1. Notice how the energy is lowest at chronaxie

VII. <u>Methods for Testing the Algorithm</u>

In the scope of this project, I will do four tests on a few sets of sample data for the capture threshold detection algorithm:

1. Capture Threshold Detection Accuracy:
    a. Find capture threshold voltages from capture_threshold_detection.py
    b. Compare with the original voltages to verify that the algorithm finds the capture threshold voltages within 5% of the true capture threshold voltage
    c. Pass = algorithm does find the capture threshold voltage within 5% above the actual capture threshold voltage
    d. Fail = algorithm does not find the capture threshold voltage within 5% above the actual capture threshold voltage
2. Rheobase and Chronaxie Accuracy:
    a. From the capture threshold voltage data from capture_threshold_detection.py, calculate the rheobase and chronaxie
    b. Calculate the percent difference of the rheobase & chronaxie from the algorithm vs the original rheobase & chronaxie
    c. Pass = algorithm does find the rheobase and chronaxie within 5% above the actual rheobase and chronaxie
    d. Fail = algorithm does not find the rheobase and chronaxie within 5% above the actual rheobase and chronaxie
3. How does changing the amount of data points that the algorithm gathers change the accuracy of chronaxie and rheobase
    a. Reduce the amount of duration values that the algorithm iterates voltage over
    b. Compare rheobase and chronaxie of the original data and the data with reduced data points

4. Investigate how adding voltage noise in the original data will affect the percent difference in rheobase and chronaxie found by the algorithm.

   The goal of this study is to investigate how voltage noise (due to other biological activity, such as other muscles or respiration) will affect the accuracy of the capture threshold detection algorithm. Increasing amplitudes of voltage noise was added to the original data. I took the following steps for each amplitude of added voltage noise to perform this study:

   a. Find rheobase and chronaxie of the original stimulus amplitude vs duration data
   b. Add noise voltage to the original stimulus amplitude data
   c. Create capture data files (in the format of Fig 5) for the noisy stimulus voltage data
   d. Uses the capture threshold algorithm to find the capture threshold voltages
   e. Calculate the chronaxie and rheobase for the noisy capture voltages found by the capture threshold algorithm
   f. Find the percent difference between the rheobase and chronaxie of the original data and the noisy data
   g. Plot % difference of rheobase and chronaxie vs. noise amplitude

1. <u>Capture Threshold Detection Accuracy:</u>

Data Set 1

- Experimental Voltage Amplitude [V]
  - voltage = [5, 3.5, 2.8, 2.6, 2.4, 2.2, 2.2]
- Voltage Amplitude [V] found using the algorithm
  - voltage = [4.99, 3.61, 2.85, 2.71, 2.44, 2.25, 2.25]
- % Difference between original voltage amplitudes and voltage amplitudes measured by the capture threshold algorithm
  - [0.2%, 3.05%, 1.75%, 4.06%, 1.64%, 2.22%, 2.22%]
  - All Pass

Data Set 2

- Experimental Voltage Amplitude [V]
  - voltage = [2.25, 1.55, 1.25, 1.15, 0.92]
- Voltage Amplitude [V] found using the algorithm
  - voltage = [2.25, 1.61, 1.27, 1.15, 0.95]
- % Difference between original voltage amplitudes and voltage amplitudes measured by the capture threshold algorithm
  - [0.0%, 3.73%, 1.57%, 0.0%, 3.16%]
  - All Pass

2. <u>Rheobase & Chronaxie Accuracy:</u>

Data Set 1

- Original Rheobase & Chronaxie
  - Rheobase = 1.86 V
  - Chronaxie = 0.17 ms
- Rheobase & Chronaxie calculated from the data found from the algorithm
  - Rheobase = 1.94 V

- Chronaxie = 0.16 ms
- % Difference between the original and calculated rheobase and chronaxie values
    - Rheobase difference = 4.3 % (Pass)
    - Chronaxie difference = 5.88 % (Fail)

Data Set 2

- Original Rheobase & Chronaxie
    - Rheobase = 0.63 V
    - Chronaxie = 0.77 ms
- Rheobase & Chronaxie calculated from the data found from the algorithm
    - Rheobase = 0.65 V
    - Chronaxie = 0.73 ms
- % Difference between the original and calculated rheobase and chronaxie values
    - Rheobase difference = 3.17 % (Pass)
    - Chronaxie difference = 5.19 % (Fail)

3. <u>Rheobase & Chronaxie Accuracy with Reduced Data Points (Performed on Data Set 1):</u>

- Original Rheobase & Chronaxie
    - Rheobase = 1.86 V
    - Chronaxie = 0.17 ms
- 7 Capture threshold data points measured from the algorithm for data set 1
    - Duration Values [ms]: 0.2, 0.5, 1, 1.4
    - Rheobase difference = 4.3 %
    - Chronaxie difference = 5.88 %
- 5 Capture threshold data points measured from the algorithm for data set 1
    - Duration Values [ms]: 0.2, 0.3, 0.5, 1, 1.4
    - Rheobase difference = 2.69 %
    - Chronaxie difference = 0 %
- 4 Capture threshold data points measured from the algorithm for data set 1
    - Duration Values [ms]: 0.2, 0.5, 1, 1.4
    - Rheobase difference = 3.23 %
    - Chronaxie difference = 0 %
- 3 Capture threshold data points measured from the algorithm for data set 1
    - Duration Values [ms]: 0.2, 0.5, 1
    - Rheobase difference = 1.61 %
    - Chronaxie difference = 11.76 %
- 2 Capture threshold data points measured from the algorithm for data set 1
    - Duration Values [ms]: 0.2, 1
    - Rheobase difference = 2.69 %
    - Chronaxie difference = 5.88 %

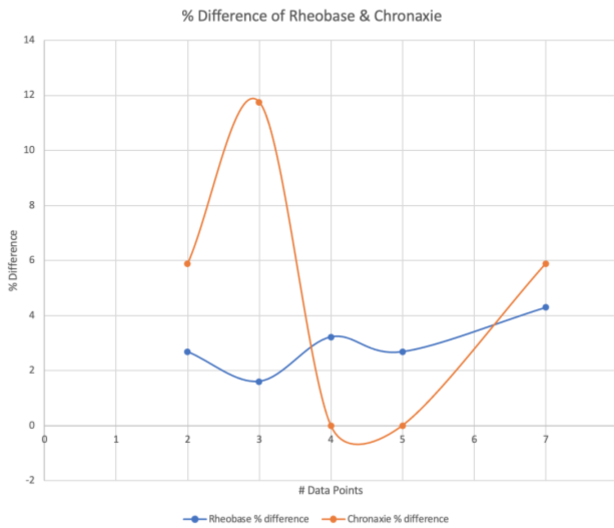


Figure 11. % Difference in rheobase & chronaxie with different # of capture threshold points taken by the algorithm

4. Noise Study

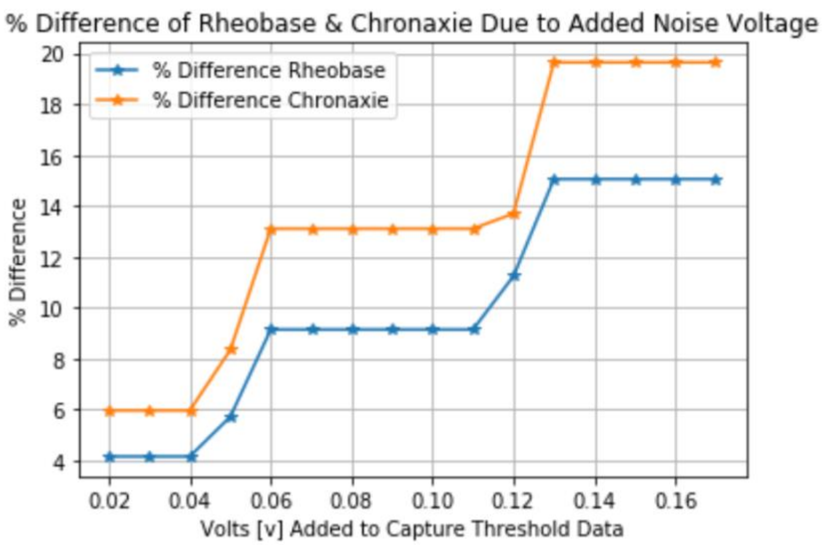


Figure 12. % Difference of Rheobase & Chronaxie Due to Added Noise Voltage

VIII. Conclusions

The algorithm does find all the capture threshold voltages within 5% of the actual threshold voltage in both data sets. The rheobase points in data set 1 and 2 were 4.3 % and 3.17 %, respectively. The chronaxie point in data set 1 and 2 were 5.88 % and 5.19 %, respectively. These percentages are a little higher than I would have liked to see. In a future iteration of this algorithm, I would try changing the step-down size of the voltages to be 20% and 3% (reference Fig 5.) to decrease the percent differences. The cost of more accurate measurements is that more data points have to be collected and it will take longer for the algorithm to find the chronaxie point.

The results of Fig 11 are not what I expected. I expected that these curves would have a more smooth and negative sloping curve. My reasoning for the shape of the curve is the following. Because of the nature of the capture threshold detection algorithm, the accuracy of the rheobase and chronaxie depends on how closely the algorithm-detected capture threshold is to the actual capture threshold. So, this curve would vary based on which duration values the algorithm uses to detect capture threshold values. From the results shown in Fig 11, I would have the algorithm capture threshold points for four stimulus duration values. Since this data is dependent on how closely the algorithm finds the actual capture values, two or three data points could still be valid choices, but more investigation and tests are needed.

The results from the study of how adding voltage noise to the original capture threshold values are interesting. It seems that there is a threshold amplitude of noise that the algorithm can take before losing accuracy. The reason goes back to how the algorithm finds the captured voltage within 5% of the actual capture voltage. When the noise voltage added to the original data surpasses 5% of the original data, the % error of rheobase and chronaxie starts to increase.

The tests performed in this paper have a small sample size, so more data is needed to make more accurate conclusions. There are also many more tests that could be done with the algorithm. Some possible future tests include (but are not limited to):

- measure the amount of energy saved compared to other manufacturer's algorithms
- measure energy consumption of this algorithm
- find optimal step-down voltage sizes
- find optimal duration values for the algorithm to iterate over for each patient

IX. Sources

1. The Strength-Duration Curve and Its Importance in Pacing Efficiency: A Study of 325 Pacing Leads in 229 Patients
2. https://bhrs.com/wp-content/uploads/2020/02/BHRS-CIED-FU-Standards-FEB-2020-FINAL-1.pdf
3. [Effect of programmed safety margin on function time of modern dual chamber pacemakers]
4. Pacing technology: advances in pacing threshold management
5. Electrophysiological Disorders of the Heart E-Book: Expert Consult
6. https://github.com/dward2/BME547/blob/master/Lectures/virtual_environments.md
7. His bundle has a shorter chronaxie than does the adjacent ventricular myocardium: Implications for pacemaker programming