COMP4951

A Computer Model of Electrocardiogram Signals

BY ANDREW AU

SUPERVISED BY SOCRATES DOKOS

Problem Formulation

There are different dynamic models for generating electrocardiogram signals that have been introduced over time.

Computer implementations of ECGs using a dynamic model is useful for both teaching and research purposes.

In teaching, dynamic models can be used to generate normal and abnormal ECG waveforms with parameter fitting, which means that you can make any ECG without real patient data or using an ECG machine.

For research a computer model can be used as a testing platform for new algorithms being used on ECG waveforms.

Problem Formulation

Current implementations of computer models are typically written in proprietary software like MATLAB and LabVIEW that are licensed and hard to distribute.

These implementations usually exist in a research context and do not perform parameter optimization to create various ECG waveforms.

My research topic is to address this issue and implement a computer model of ECGs that is open source, easy to distribute and that implements parameter optimization to create a variety of normal and abnormal ECGs.

Such an implementation can be used as a tool for learning or a tool for resource/testing algorithms in the future.

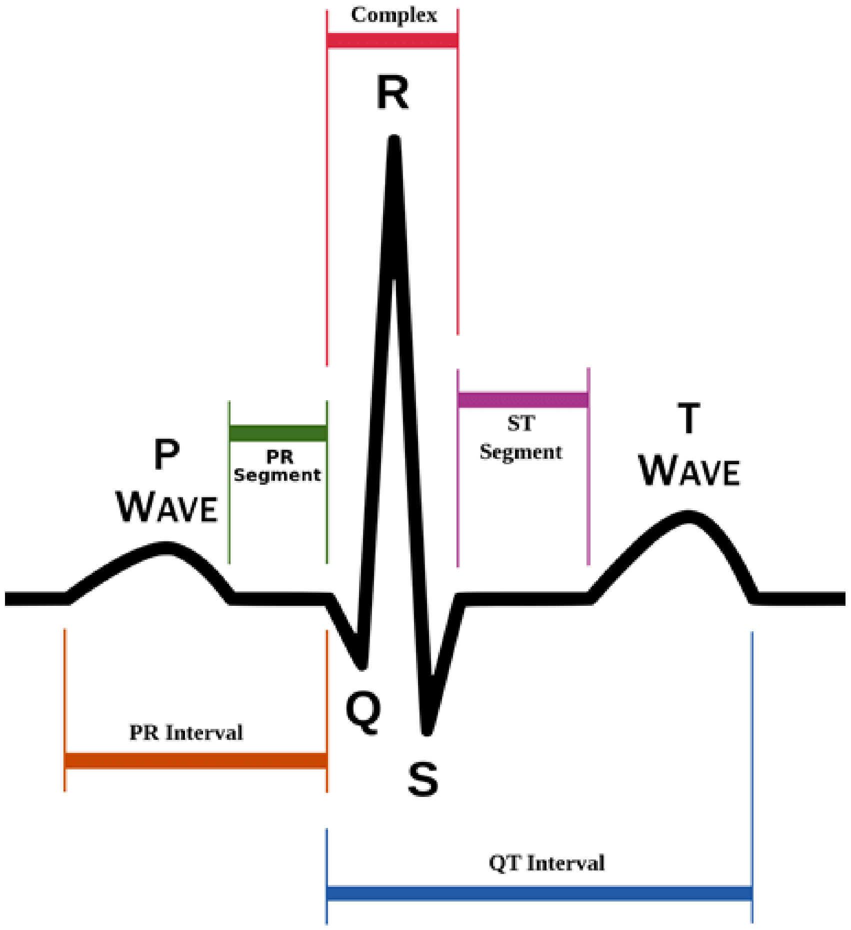
Background Information

An electrocardiogram (ECG) is a waveform generated by the electrical signal of the heart as it contracts and expands.

A single cycle of the ECG represents a single heartbeat.

An ECG is obtained by placing several electrodes onto a person’s body and limbs and recording the potential difference between the nodes with a machine.

The ECG is typically used in the medical field to diagnose heart disease and abnormalities.



Background Information

An ECG taken from a healthy heart is characterized by a series of peaks and troughs.

These peaks and troughs are labelled PQRST.

- P-Wave: Atrial depolarization (contraction of upper heart chambers)
- QRS complex: Ventricular depolarization, atrial repolarization (contraction of lower and relaxation of upper heart chambers)
- T-Wave: Ventricular repolarization (relaxation of lower heart chambers)

During the T-wave the heart prepares for the next heart cycle.

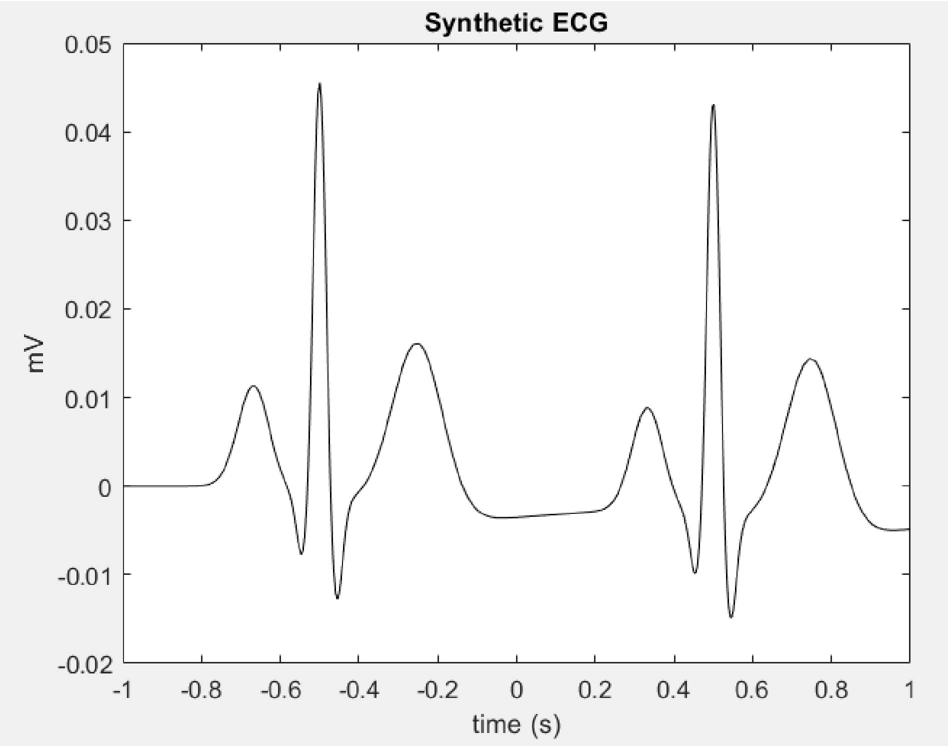
Background Information

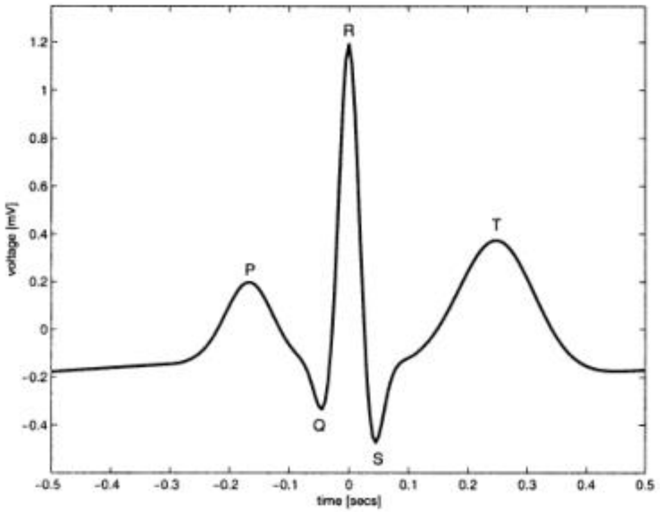
ECGs are taken from a patient through the use of an ECG machine.
However, acquiring a patient and an ECG machine can be difficult, in which case simulators are useful.

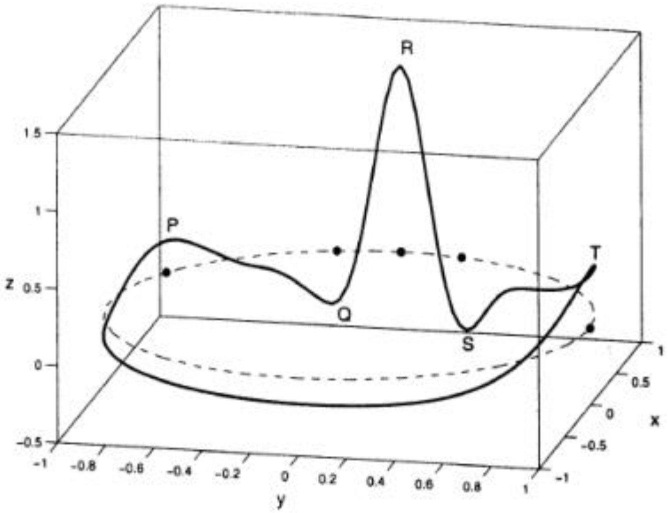
Simulators can be used to create any normal and abnormal ECG without having to use real patient data.

This means that we can simulate different heart conditions as well.

Simulators are implemented with mathematical models, of which there are different solutions.







Background Information

One dynamic model is the one proposed by McSharry et al. in 2003.

This model generates a trajectory in 3D space which moves around a unit circle on the xy plane.

Events PQRST are placed along the unit circle.

As the trajectory approaches an event it moves in the positive/negative position from the z-axis.

As the trajectory moves away from an event it settles back into a resting position.

Background Information

The dynamic equations of motion are given by a set of three ordinary differential equations:

$$\dot{x} = \alpha x - \omega y$$
$$\dot{y} = \alpha y + \omega x$$
$$\dot{z} = -\sum_{i \in \{P,Q,R,S,T\}} a_i \Delta\theta_i \exp\left(-\frac{\Delta\theta_i^2}{2b_i^2}\right) - (z - z_0)$$

Where:

$\alpha = 1 - \text{sqrt}(x^2+y^2)$

$\Delta\theta_i = (\theta - \theta_i) \bmod 2\pi$

$\theta = \text{atan2}(y, x)$

ω is angular velocity as it moves in the xy plane

The values of θ_i, a_i and b_i are a set of fixed values that are calculated to fit an ECG.

Literature Survey

Year	Author	Findings
2003	McSharry et al.	A Dynamic model for generating electrocardiogram signals using a system of ordinary differential equations. Can be used as a testing platform for algorithms.
2004	P McSharry, GD Clifford	Open source software for generating electrocardiogram signals implemented in C, Java, MATLAB and Octave.
2005	P McSharry, GD. Clifford	Method to Filter ECGs and Evaluate Clinical Parameter Distortion using Realistic ECG Model Parameter Fitting.
2005	R Sameni, MB Shamsollahi, C Jutten, M Babaie-Zadeh	Filtering Noisy ECG Signals Using the Kalman Filter Based on a Modified Dynamic ECG Model (based on McSharry model). Also introduces a method of parameter selection by taking the average of a phase wrapped graph (3D ECG plot).
2010	O Sayadi, MB Shamsollahi, GD Clifford	Synthetic ECG generation and Bayesian filtering using a Gaussian wave-based dynamical model.

Literature Survey

Year	Author	Findings
2012	P Kovacs	ECG Generator based on Geometrical features using polynomial approximation and rational functions.
2014	J Kubicek, M Penhaker, R Kahankova	Design of a synthetic ECG signal based on the Fourier series.
2018	I Andras, L Michaeli, D Grimaldi	Model for generating simple synthetic ECG signals using trigonometry and linear functions/derivation of Gaussian pulse.

Project Scope

The aim of this project is to implement the basic computer model of an ECG proposed by McSharry et al. with parameter fitting.

The implementation will include a user interface for inputting variables and data for parameter fitting.

It will **NOT** implement features of ECGs such as *RSA*, *Mayer Waves* and *noise* from real ECGs.

The implementation will be written in Python and SciPy, a library for math functions.

Timeline for Term 1

Week 1-2	Week 3-4	Week 5-6	Week 7-10
Understanding the problem	Implement model in MATLAB	Seminar Presentation	Implement model in Python and End of Term Report
Related reading on ECGs	Experience with using ODE Solvers and math required for basic model	Prepare materials for presentation	Implement basic model in Python using SciPy. Write end of term report.

Timeline for Term 2

Week 1	Week 2-4	Week 5-8	Week 9-10
Learn components to implement user interface	Implement user interface	Implement Parameter fitting	Testing/Parameter fitting cont.
Look up detail on how to implement user interface	Implement user interface for reading data/inputting variables	Implement some form of parameter fitting	Continue implementing parameter fitting/other requirements or start on testing

Timeline for Term 3

Week 1-4	Week 5-6	Week 7	Week 8-10
Continue work from Term 2 Complete implementation from Term 2 and finish implementation.	Testing Test features of implementation of parameter fitting.	Packaging/ distribution Prepare everything for open source distribution	Final Report Write final report

Thank you for listening

END OF PRESENTATION.