# MSc Project: Oscillometric Blood Pressure


**Belinda Kneubühler 2504756K**

School of Engineering
College of Science and Engineering
University of Glasgow


Submitted in fulfilment of the requirements for the
Degree of MSc Computer Systems Engineering

Supervisor: Bernd Porr
Second Supervisor: Kiran Ramesh

August 2020

# Abstract

Abstract text goes here.

# Contents

# List of Tables

# List of Figures

# Acknowledgements

Acknowledgements text goes here.

# Declaration

# Chapter 1

# Introduction

## 1.1 Problem Description

## 1.2 Objectives

## 1.3 Outline

This report is split up into tree main parts. The first part, chapter 2 Background, discusses the thory of oscillometric blood preessure measurement. It focussed on research material and its implementaion in real-time processing.

The second part, chapter 4 Software, discusses the implementation of the application that was developed based on the needs described above and the findings made in the preceding chapter.

The third part, chapter 5 Results and Discussion, looks back at what was achieved in the project and gives an outlook on further recommended steps.

# Chapter 2

# Background

This chapter discusses the findings of researching existing and proposed algorithms for determining systolic and diastolic blood pressure through the oscillometric method. This report focusses on relatively simple algorithms that can be implemented in a real-time application, which is the goal of this project.

The first part of this chapter defines what blood pressure is and how it is measured manually. Afterwards

A brief section at the end of this chapter explores algorithms beyond the scope of this project that have been proposed in recent literature.

## 2.1 What is Blood Pressure?

Blood pressure is an important bio-medical measurement and often used for diagnostics in cardiovascular diseases. Most commonly, systolic and diastolic blood pressure are mentioned in pairs, but what do those numbers mean?

The heart is a pump that has two phases. When the ventricle is relaxed, the heart fills up and the diastolic blood pressure (DBP) is observed. When the heart contracts, it pushes the blood through the arterial system and the systolic blood pressure (SBP) is observed. A third characteristic is the difference between SBP and DBP, the pulse pressure (PP). Finally, the mean arterial pressure (MAP) is defined as the average pressure in the artery. Figure 2.1 shows how these pressure values are connected in one heartbeat. MAP is the area underneath the blue curve divided by the time of one pulse, indicated by the orange area. (2) Because the blood pressure is simplified as a sine wave the MAP is in the middle between SDP and DBP, usually the MAP is closer to DBP.

Blood pressure can be measured with invasive and non-invasive methods. The most accurate methods are invasive, but require professional expertise because a catheter has to be injected into the blood vessel. Subsequently, non-invasive methods are more common because they are

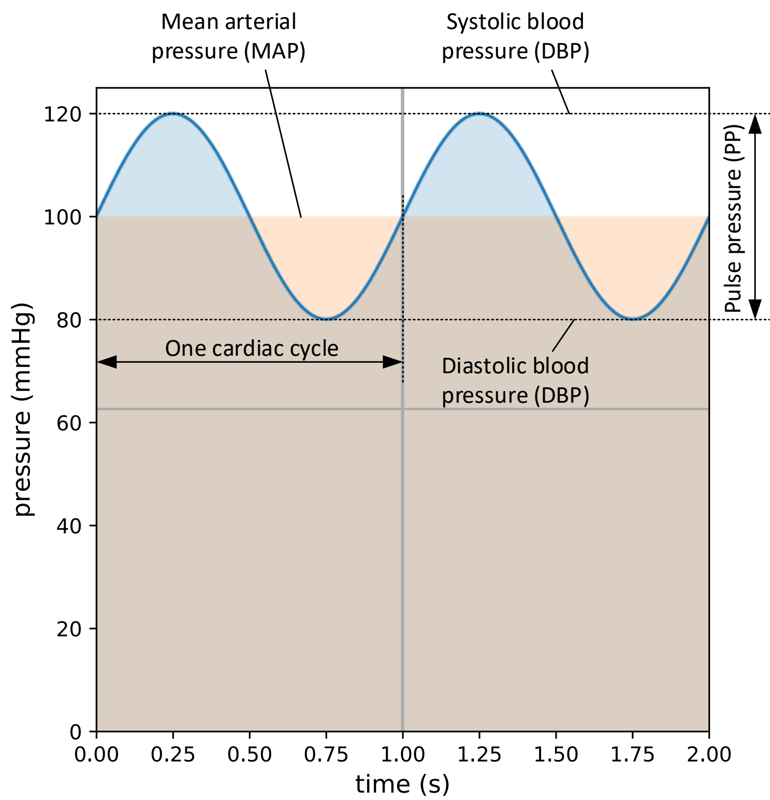


Figure 2.1: Characteristic blood pressure values, simplified as a sine wave. The values are: MAP: 100 mmHg, SBP: 120 mmHg, DBP: 80 mmHg and heart rate: 60 beats/s.

more convenient to use. With automated methods, taking blood pressure at home has become convenient and easy. However, the inaccuracies of these measurements are often ignored or even unknown.

## 2.2 Manual Blood Pressure Measurement

The manual method of measuring blood pressure relies on a sphygmomanometer and listening to the heart or Korotkoff sounds (named after their discoverer) with a stethoscope. This is called the *auscultatory method*. *Palpation* is another method where the pulse is felt at the wrist. It only allows detection of the SBP. The auscultatory method is described below because it provides recommendations that are true for all blood pressure measurements using a cuff.

The sphygmomanometer is shown in figure 2.2. It consists of an inflatable cuff that is connected to a rubber pump and a scale that indicates the pressure in the cuff in mmHg. The pump is equipped with a valve that can be opened to release air.

Millimetres of mercury (mmHg) is a unit to describe pressure. 1 mmHg is defined as the pressure generated by a column of mercury of 1 mm height. Millimetres of Mercury is not part of the International System of Units (SI) but was defined by it before 2019 as 1 mmHg

= 133.322 Pa using standard gravity.(3) Since 2019, mmHg is no longer included in the SI brochure. However, it is still widely used in the medical field. (4)

Figure 2.2: Picture of a sphygmomanometer showing the cuff, inflation pump and mercury meter.

Intra-arteria or direct BP measurement is considered the 'gold standard', but most studies compare the developed algorithm with the auscultatory method, which underestimated SBP and overestimates DBP. (5) Nonetheless, the protocols to test automatic blood pressure monitors given by the British Hypertension Society (BHS) and their American counterpart, the Association for the Advancement of Medical Instrumentation (AAMI), both use the auscultatory method as a reference. (6; 7; 8)

### 2.2.1 Auscultatory Method

The cuff size must be appropriate for the patient's arm. The rubber bladder inside the cuff should cover more than 80 % but less than 100 % of the arm's circumference.

The cuff is completely deflated and tightly fit around the upper arm of the patient. The centre of the bladder should be over the brachial artery and the arm supported at heart level. The stethoscope is to be placed over the artery between the cuff and the patient's elbow and should not touch the cuff. This could influence the reading because it puts additional pressure on the artery and can interfere with hearing the sounds.(9; 10)

The valve at the pump is completely closed and the cuff is inflated to a pressure of about 30 mmHg above the expected systolic pressure. This causes the artery to flatten and not let any blood through. Subsequently, the valve is slightly opened to slowly deflate the cuff at a rate of approximately 3 mmHg/s. The artery will open and let small amounts of blood through. Systolic pressure is read when a pulse is first heard. Meanwhile, deflation continues. Diastolic pressure is read when the sounds disappear completely. Ultimately, blood can flow freely through the artery. After making sure that no further sounds can be heard by deflating for at least another 10 mmHg, the valve is opened completely to rapidly empty the cuff.(9; 10)

There are some discrepancies in literature on how to define the point of the diastolic blood pressure. Some references define it as when the Korotkoff sounds disappear completely. (9; 10) Others, predominantly older resources, define it as the point where the sounds are muffled.(2) Here, the former is assumed, because most current literature does and some literature suggests that the auscultatory method overestimates diastolic blood pressure. (11)

### 2.2.2 Calculation of MAP

Traditionally, MAP is calculated as an estimate by adding a third of the PP to the DBP. Or, how it is more commonly referred to, adding SBP and twice the DBP and dividing the result by 3 (equation 2.1). According to calculations by Bos *et al.*, this generally underestimates the MAP and they suggest to add 40% rather than a third of DBP. Furthermore, they suggest to using MAP measured by oscillometric devices rather than calculating it from values obtained through the auscultatory method. (12)

$$MAP = \frac{SBP - DBP}{3} + DBP = \frac{SBP + 2 \times DBP}{3} \tag{2.1}$$

According to Joe (13), nurses in intensive care units (ICU) today still use the manual calculation method on the automatically obtained SBP and DBP rather than rusting the measured MAP, which introduces large errors.

## 2.3 Automatic Blood Pressure Measurement

Most modern automatic devices use the oscillometric method to determine blood pressure. They are used both at home and in professional environments.

Automatic blood pressure monitors measure pressure in a cuff similar to the one used in the manual method and uses the small pressure changes (oscillations) extracted from the deflation curve to estimate blood pressure. Commercially available devices are usually equipped with an electric pump and the bladder contains a pressure sensor. Otherwise, they look similar to the cuff used for manual measurements. Some devices use wrist cuffs, but those are rarely recommended. (14) The most significant problem using wrist measurements is the influence of gravity on blood pressure and the accompanying risk of systematic errors when the measurement is not taken at heart level. (2) This problem is avoided by taking blood pressure at the upper arm.

**General Procedure** The procedure is also similar to the manual method. The pressure in the cuff is increased to a level above the expected systolic pressure, which leads to the artery being pushed close. While releasing the pressure slowly through the valve, blood starts flowing through the artery, resulting in small increases in pressure relative to the continued deflation of the cuff. As the cuff deflates further, these relative changes increase to a maximal value before they start decreasing again.(15; 16; 17) The shape, magnitude or envelope of these oscillations are used in various automatic blood pressure algorithms. Methods that do not use oscillations exist, but are not discussed here.

**Oscillation Extraction** There are two ways the oscillometric waveform (OMW) is extracted from the deflating pressure curve. The first one is filtering. A bandpass filter with a lower cut-off frequency between 0.1 to 0.5 Hz and an upper cut-off frequency between 0.1 to 0.5 Hz is recommended. (15) Implementations usually use first (18) to sixth order (6) Butterworth filters. They are known for their flat pass-band response and good frequency response.

The second approach is to use de-trending. It requires to know the beginning of each pulse to be able to fit a line of continuously deflating pressure to the identified points. Its advantage is, that it additionally reproduces an estimated deflation curve. However, it requires additional data, for example, ECG for pulse detection.(15)

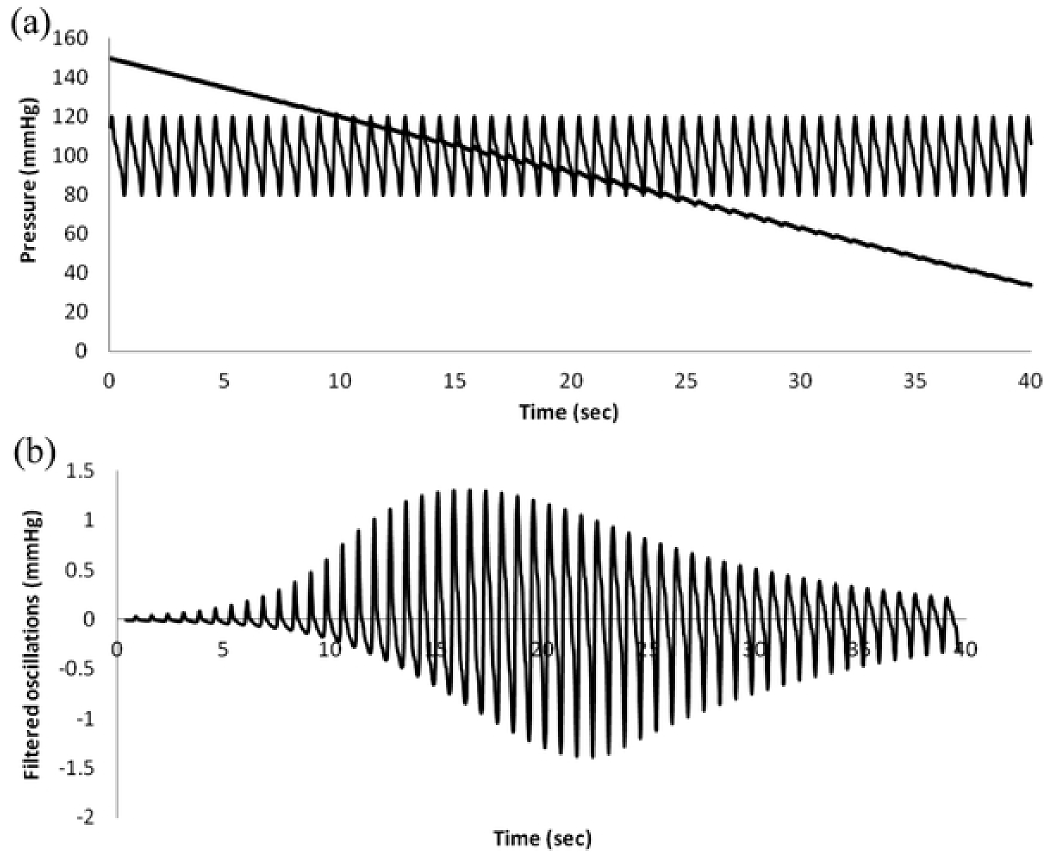


Figure 2.3: A simulated blood pressure extraction. (a) shows the defalting pressure in the cuff and (b) the high-pass filtered oscillations. Source: (1) (CC)

Ultimately, from a deflation curve as in the simulated signal in figure 2.3 on the top, the oscillations are extracted. The bottom plot in figure 2.3 shows oscillations extracted with a thrid order Butterworth bandpass filter with the cut-off frequencies at 0.5 Hz and 5 Hz. (1)

**Envelope** The envelope of the OMW is used by basic automatic algorithms. Formation of the envelope is achieved in different ways. The simplest way is to register only local maxima. Similar to that, and most common, is the subtraction of the following through from a local peak or interpolating the curve of local maxima and local minima to subtract them from each other. Some algorithms additionally fit a curve on the obtained oscillometric waveform envelope (OMWE) in an attempt to remove artefacts. (19)

### 2.3.1 Maximum Amplitude Algorithm

The maximum amplitude algorithm (MAA) is based on the assumption that the oscillations are maximal when the pressure in the cuff equals arterial pressure. Accordingly, the recorded pressure at which oscillations are maximal is considered a valid estimate of the MAP (1; 16; 20)(21), as long as the compression chamber is kept small (22). To avoid introducing errors, the cuff should always be tightly fit, because air volume in the cuff causes maximum oscillations to be above the true MAP. Hence, Ursio and Cristalli suggest to use the lowest pressure of the plateau of maximal oscillations as the value to assess MAP.(17)

A recent publication by Chandrasekhar *et al.* (11) employs a more complex model than the one originally introduced by Mauck *et al.*(22). They conclude that the MAA results in a weighted average of systolic and diastolic BP. According to their model, MAA underestimates MAP for higher blood pressure.

**Fixed-Ratio Algorithm** The fixed-ratio algorithm builds on the MAA and is based on the assumption that the systolic and diastolic blood pressure occur at specific fractions of the maximum oscillations before and after its occurrence. Determining SBP and DBP after applying the MAA was tested by Geddes *et al.* in 1982 (20). Geddes recorded Korotkoff sounds while measuring oscillations <to find a ratio of oscillation amplitudes for SBP and DBP. They defined the ratio for the systolic pressure to be 0.5 and the ratio for the diastolic pulse as 0.8. These ratios were found empirically and Geddes acknowledges that the systolic pressure is overestimated and the ratio for the diastolic pulse is not constant for a range of different diastolic pressures.

Later studies tried to find accurate ratios, mostly experimentally with the ratio for SBP usually being determined between 0.45 and 0.73 and the ratio for DBP a bit higher between 0.69 and 0.83.(15; 16) Mathematical models confirmed that a generalised ratio cannot be found. Parameters like the age of the subject and in this regard, the arterial stiffness as well as pulse pressure influence the ratios. (17) A higher PP results in a smaller ratio for SBP and larger ratio for DBP. A stiff arterial wall causes volume changes to happen slower. While this has a small effect on the systolic ratio, the diastolic ratio decreases with the stiffness of the artery.(1)

### 2.3.2 Derivative Algorithm

The derivative algorithm uses the OMWE and plots it against the deflation pressure. The points of the maximal and minimal slope are determined. The pressure point where the derivative of the OMWE is maximal is assumed to be the diastolic BP and where the derivative is minimal is the systolic BP respectively. (6; 15)

Even though mathematical models have confirmed this method to produce valid estimates of both SBP and DBP without the need for empirically obtained ratios, it is extremely vulnerable to noise and therefore not used in practical applications. (1; 11)

Jazbinsek *et al.* evaluated the fixed-ratio and derivative algorithms. They used a device equipped with ECG and a microphone to validate their results. Various ways to form the envelope were used, including detrending, filtering and even using the low-frequency part of the audio signal of a microphone that recorded Korotkoff sounds. However, to evaluate the fixed-ratio method, they used a commercial device to find average ratios and used the values from the same device as a reference for the evaluation. Their evaluation of the derivative algorithm showed many local extrema as expected. The algorithm was only able to perform by applying additional constraints that bias the measurements significantly. For example, the extrema has to be distanced from the MAP value more than 15 mmHg. (6; 23; 24)

### 2.3.3 Other Algorithms

Since the above-mentioned algorithms use only the envelope of the oscillations to determine BP, they discard all the information that lies within the shape of a single pulse. Naturally, scientists have tried to find ways to use this information to improve BP algorithms.

There are a variety of different ideas that have been tested in small scale experiments of mostly with less than 30 test subjects. While they all claim to improve BP measurements compared to the fixed-ratio algorithm, they often lack significant evidence and mathematical validation.

**Non-Fixed Ratio** Sapinski (5) proposed a standard algorithm. This algorithm uses ratios to determine the SBP and DBP from MAP as above but calculates the ratios based on the oscillations. The integral of the maximal oscillation is divided by its time period and gives the amplitude of the pulse wave at systolic pressure. The amplitude at diastolic pressure is defined as the difference between the MAP amplitude and the SBP amplitude. This algorithm is based on theoretical assumptions formed form comparisons to the direct method, e.g. that the added oscillation pulse amplitudes at systolic and diastolic pressures equal the amplitude at mean pressure. Sapinski mentions that the average value of systolic oscillation is 40 % of the MAP amplitude and 60 % for the diastolic oscillation, respectively. This part is regularly quoted in other literature, without mentioning that this is not a proposed ratio. Sapinski concludes, that the proposed algorithm does not fulfil the criteria of the AAMI standard compared to the auscultatory method, but does compared to the invasive method. No further literature has been found that validate or contradict these findings.

**Pulse Morphology** Specific characteristics of single pulses can be examined to determine blood pressure. Following are four of the most commonly used indices. The definitions are used from Mafi *et al.*(25)

- Stiffness index (SI): The height of the subject ($h$) divided by the time difference between systolic and diastolic peak ($\Delta T$) (equation 2.2) is an indicator for arterial stiffness, but requires to know the person's height.

$$SI = \frac{h}{\Delta T} \tag{2.2}$$

- Augmentation index (AI): The difference of systolic ($A_S$) and diastolic peak ($A_D$), divided by the systolic peak, expressed in percentage of pulse pressure (equation 2.3).

$$AI = \frac{A_S - A_D}{A_S} \times 100\% \tag{2.3}$$

- Reflection index (RI): The of systolic ($A_S$) divided by the diastolic peak ($A_D$), expressed in percentage (equation 2.4).

$$RI = \frac{A_S}{A_S} \times 100\% \tag{2.4}$$

- $\Delta T/T$ Ratio: The time difference between systolic and diastolic peak ($\Delta T$) is decided by the duration of the pulse ($T$). Both $T$ and $\Delta T$ increase with age.

Mafi *et al.*(25) plotted the indices in time and used absolute maxima or minima to determine MAP. There is a significant spike where MAP is expected. Using local maxima and minima next to the found MAP to estimate SBP and DBP has less significance. Problematic with this approach is that the reference values were measured by an Omron device with an unknown implementation of oscillometric blood pressure. Because the device did not display the measured MAP, the MAP reference was calculated using equation 2.1, which is known be wrong using accurate values of SBP and DBP and likely delivers arbitrary values when using values from an automatic device.

Another implementation from Mafi *et al.*(26) is taking the '1st' derivative of each pulse and taking the maximal value of that to form another curve that looks similar to the one used in MAA. This curve is then treated equally to the fixed-ratio algorithm to determine MAP, SBP and DBP. The ratios are determined experimentally using reference measurements. Additionally, the MAP is again calculated using the faulty equation 2.1 and the 18 test subjects aged between 24 and 68 are healthy and have no history of cardiovascular disease. The authors argue that the shape is less sensitive to noise than the amplitude and therefore, their approach is more robust than MAA.

**Model-Based Algorithms** These algorithms use a predicted OMWE and fit it to the observed one to determine the characteristics. While they do consider factors that the standard algorithms discard, such as arterial stiffness and pulse pressure, they are vulnerable to artefacts that are not considered by the model. They could be used to validate algorithms. (1)

**Neural Networks** The features extracted from above, and more, are often used in neural networks (NN). However, it has been shown, that the indices can reliably only be used to determine the MAP. Moreover, some of the features are highly dependant on external factors. For example, the exact deflation rate influences the duration of the OMWE (proposed by Lee *et al.*(27)) and the constraint of 2 to 3 mmHg/s given by Lim *et al.*(18), who implemented a NN using the proposed features, seems hardly enough. Similarly, the used filter by Lim *et al.*, a first-order bandpass Butterworth filter with cut-off frequencies of 0.5 Hz and 5 Hz, is likely to remove valuable information from the pulses.

## 2.4 Summary

Write summary of the algorithms with their advantages and disadvantages

# Chapter 3

# Hardware

This chapter describes the hardware used in this project.

# Chapter 4

# Software

This chapter is a description of the developed software. It provides a brief overview of the important architecture and algorithm implementations. The projects GitHub page provides the full source code, setup instructions and Doxygen documentation.(28) The project is licensed under the GNU General Public License v2.0.

## 4.1 Overview

The application is split into the user interface (UI) and data processing. Each part runs in their thread and they are kept separated from each other. The data processing class is called `Processing` and the user interface class `Window`.

The user interface is built with Qt, an open-source widget toolkit for creating graphical user interfaces. Additionally, the subset Qt Widgets for Technical Applications (Qwt) is used to display plots of the acquired data.

The `Processing` class handles data acquisition and processing, sending notifications to the observer with instructions on what action to perform or what values to update.

Communication between the classes is done with callbacks in an observer pattern from the `Processing` to the `Window` class. The processing class is an observable subject that the window class registers to. The window has a reference to the processing class to pass on user input.

### 4.1.1 Class Diagram

The 4.1 The Class Diagram. Needs work shows a simplified class diagram of the application. Following is a more in-depth discussion of its elements.

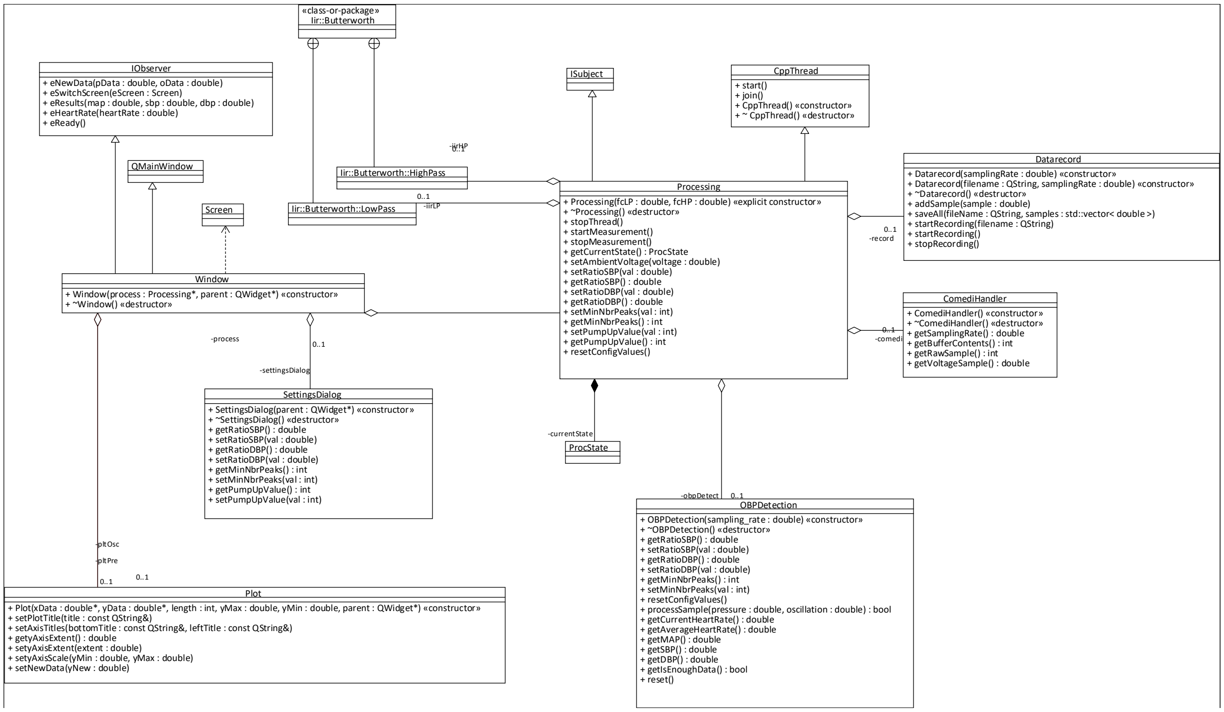


Figure 4.1: The Class Diagram. Needs work.

## 4.2 Data Processing

In the processing class, the data is acquired and processed. The acquisition is handled by a ComediHandler object, that abstracts handling of the underlying hardware. The data is then filtered and the observers are notified so they can display the new data.

The logic in the processing object is handled through a state machine. The state deflate is where the data is analysed to extract the blood pressure characteristics from the oscillations. This is done in a separate class, called OBPDetection.

### 4.2.1 State Machine

It is implemented in the Processing class.

### 4.2.2 Processing Class

The processing class inherits from the CppThread class and the ISubject class. CppThread is a wrapper to the std::thread class that was written by Bernd Porr to avoid static methods and makes the class a runnable thread. It has an instance of ComediHandler to acquire and two filter instances to pre-process the data. The raw, unfiltered data is sored in a vector that can be handed to the Datarecord instance to save it as a file. Meanwhile, the filtered data is sent to the GUI to display and handed to the OPDetection instance that performs the algorithm. While the data acquisition and filtering are running whenever the thread is running, the state machine described

Figure 4.2: UI

above decides when data is passed to the OBPDetection or stored to a file.

#### Thread Safety

The processing class has public getter and setter methods, that allow objects, with access to it, to read and change configuration variables. These variables are defined as atomic, to make access to them thread-safe. Similarly, the booleans set by starting and stopping the measurement and stopping the thread are atomic.

### 4.2.3 ISubject

ISubject is a simple interface that lets the implementing class easily notify its observers about certain events. Each notification is realised as a private notify-X method. The public methods are 'attach' and 'detach'. Through these, a class that implements the above described IObserver (section **??**) can be attached to the subject. This stores a reference to the object in a list. When one of the notify methods is called, the notification is sent to all the observers in the list. The observer can be removed through the detach method.

Just like the IObserver class, the ISubject class can not be instantiated directly but has to be used to inherit from.

### 4.2.4 Data Acquisition

### 4.2.5 Filtering

### 4.2.6 Blood Pressure Detection

### 4.2.7 Storing the Recorded Data

## 4.3 User Interface

The user interface is shown in figure 4.2 UI. It is split up into two parts. The left side accepts user input and gives instructions to the user what they have to do to take their blood pressure. The right side shows the data being acquired in real-time. There are two plots. The upper plot shows pressure data filtered with a low-pass filter of 10 Hz, the lower plot shows the data additionally low-pass filtered at 0.5 Hz, which results in a bandpass filter. This is the oscillogram that is the main input for the algorithm to determine the user's blood pressure.

As mentioned above, the graphical user interface (GUI) is built with Qt. The QtDesigner was used to design a first draft of the GUI, but because this does not allow integration into applications that are built outside of the Qt development environment, the whole GUI was completely built-in C++ code from this draft.

### 4.3.1 Guided Blood Pressure Measurement

The left side of the GUI guides the user through taking blood pressure. The process is split up into five pages that are shown in figure 4.3. All pages have a dial that shows the current pressure in mmHg. The first page has information on how to prepare for the measurement. Pressing the button at the bottom of the page starts the measurement process. The button is only enabled if the processing part of the application is ready.

Figure 4.3: The screens that guide the user through taking their blood pressure in the order that they are shown.

The second page instructs the user to pump up the pressure in the cuff to a certain value. The default value is 180 mmHg. Once that value is reached, the GUI automatically switches to the next page.

This page tells the user to release the pressure in the cuff again. This should be done slowly at a rate of approximately 3 mmHg/s. There is currently no other feedback than the dial for how fast the pressure is being released. At the bottom of the page, the current heart rate, calculated from the latest oscillation peaks, is shown.

Once enough data is collected, the fourth page is shown. It requires the user to deflate the cuff completely to show the results. If the algorithm fails to collect enough data within a specified time (currently configured as 5 minutes) the measurement stops and goes back to the start page.

When the pressure in the cuff reaches nearly zero, the final results page is shown. It displays MAP, SBP, DBP and the average heart rate during the measurement. All data is shown as whole numbers without decimals.

### 4.3.2 Menu

A menubar provides access to an information pane as well as a settings pane. Both open up as dialogue windows and disable user input on the main window. Data acquisition is kept running and is displayed in the plots.

#### Information

The information pane shows the application version number, shows the licence and provides a link to the project GitHub page. Figure 4.4 shows the dialogue.

Figure 4.4: The information dialogue.

#### Settings

The settings dialogue (shown in figure 4.5) lets the user change some application configurations. The user is not recommended to change these if they are not aware of the consequences. The settings are stored persistently but only take effect after restarting the application. The range of accepted values is limited, to keep the algorithm working. Because not all values have been tested, the user is warned that the application might not perform reliably anymore. The values are stored once the user presses the ’OK’ button. If the ’Cancel’ button is clicked, the settings application closed without saving the values.

The values can be reset by pushing the corresponding button. These changes take effect immediately.

Figure 4.5: The settings dialogue.

The handling and storing of these values are done through the QSettings class of the Qt library. It allows to save and load values by specifying a string key. If there is no key found with the provided key, the default value is selected. The default value is what is hardcoded in the Processing class.

### 4.3.3 The Window Class

The `Window` class inherits from the `QMainWindow` class and the `IObserver` class. The `QMainWindow` class makes the class an executable Qt window, with Qt taking care of updating the GUI and generating events for button clicks and so on. The `IObserver` makes the class able to be registered as an observer for a subject that will send notifications to the observer. More on the `IObserver` below.

All GUI elements are set up in the constructor of `Window`, including the settings and info plane. However, they will only be shown if necessary. E.g. what page of the user instructions is shown is defined by the setting of the screen enum (currentScreen).

The Window object is instantiated with a reference to a Processing object in the constructor. This is necessary so the user inputs can be relayed and the Processing thread can be stopped when the window is closed and the application exited.

### 4.3.4 The IObserver Class

The `IObserver` class defines the functions that the observable class (the subject) uses to notify the observer. All functions are virtual but implemented as empty functions. This way, the observing class can choose to implement and therefore listen to the notifications that it wants to and ignore the ones that it does not.

The IObserver class can not be instantiated directly but has to be used to inherit from. This is achieved by having a protected constructor.

**The following methods can be implemented:**

- Ready:
  Informs the observer that the subject is ready.
- New Data:
  Sends a new data pair to the observer. Each data pair consists of pressure data and oscillation data. Both are doubles.
- Switch Screen:
  Tells the observer which screen to display.
- Results:
  Sends the results to the observer. The results are three doubles for MAP, SBP and DBP.
- Heart Rate:
  Sends a new heart rate value to the observer. The value is a double.

These methods are called from a class that inherits from ISubject, which is explained in detail in section **??** below.

#### Thread safety

Because the methods from the `IObserver` class are called from an object, which is likely to be running in another thread than the window, it is important to implement all functions in a thread-safe manner. Qt offers a good way to do this by sending queued events.

One example of how to enable the start button, in a not thread-safe way is by accessing it directly, as follows:

```
btnStart->setDisabled(false);
```

Instead, the function `QMetaObject::invokeMethod` is called to send an event to the GUI thread. This is done by specifying the object, which function of the object to call, what arguments to set, and the type of event to send. The `Qt::QueuedConnection` will send an event into a queue that will be handled when the thread is executed. The function to call is given as a string argument. Considering the example above, this results in the following statement:

```
bool bOk = QMetaObject::invokeMethod(btnStart, "setDisabled",
 ↪  Qt::QueuedConnection,  Q_ARG(bool, false));
assert(bOk);
```

The return value confirms if the connection could be made, i.e. the given string is a valid function to call for the given object. This is tested through an assert during development. The assert will fail if the boolean is not true. It is important not to have the assert around the whole function call, because it will be removed in the release build.

The plots have their underlying data set changed every time a new sample is available because a new sample is always added on the right side and an old one is discarded on the left side. This is made by changing the raw data in memory and therefore, the Qt process is not informed about that. To solve this issue, the plots are regularly updated manually. This happens in a timer event with an interval of 50 ms, which is enough for the human eye to see a continuous movement. To avoid thread safety problems, the plot objects are accessed after acquiring a mutex.

## 4.4 Third Party Software

- Qt
- CppThread a wrapper to the std::thread class written by Bernd Porr to avoid static methods.
- *iir1* An implementation of the infinite impulse response (IIR) filters for sample-by-sample, real-time processing written in C++ by Bernd Porr. Provided as a library.
- *plog* Portable, simple and extensible C++ logging library.

# Chapter 5

# Results and Discussion

This chapter highlights important results of the project.

- valve needs to open continuously to have a steady stream of 3mmHg/s opening introduces noise

- many algorithms only work for clean data

- MAP is used for diagnostics

- why are microphones not used to determine blood pressure with the karakoff sounds?

- PTT (beyond oscillometry)

..

## 5.1 General Problems with Algorithms

most algorithms expect perfect data to be 'perfect'

natural variablility of blood pressure during measurement left and right arm difference 10mmHg normal

mmHg is no longer a SI unit since 2019

# Chapter 6

# Conclusion

This chapter discusses the outcome of the project and proposes the next steps to take.

The goal of this project was to be able to conduct experiments with test subjects and to record datasets. Unfortunately, due to the ongoing situation when this project was conducted, this was not possible. Even though all possible safety measurements were considered and approved by the University's risk assessment department, the approval process was rendered impossible to go through within the given time frame. Apart from not being able to record any datasets, the algorithm might suffer from systematic errors due to subjective charasteristics in the developers blood pressure oscillations.

# Bibliography


[1] C. F. Babbs, “Oscillometric measurement of systolic and diastolic blood pressures validated in a physiologic mathematical model,” *BioMedical Engineering Online*, vol. 11, 2012.

[2] W. F. Boron and E. L. Boulpaep, *Medical Physiology, 2e Updated Edition E-Book*. Elsevier health sciences, 2012.

[3] “The international system of units (si),” *Organisation Intergouvernementale de la Convention du Mètre*, 2006.

[4] D. Pražák, V. Sedlák, E. Sınır, and F. Pluháček, “Changing the status of mmhg,” *Accreditation and Quality Assurance*, vol. 25, no. 1, pp. 81–82, 2020. [Online]. Available: https://doi.org/10.1007/s00769-019-01414-7

[5] A. Sapinski, “Standard algorithm for blood pressure measurement by sphygmo-oscillographic method,” *Medical and biological engineering and computing*, vol. 34, no. 1, pp. 82–83, 1996.

[6] V. Jazbinsek, J. Luznik, S. Mieke, and Z. Trontelj, “Influence of different presentations of oscillometric data on automatic determination of systolic and diastolic pressures,” vol. 38. Annals of Biomedical Engineering, 2010.

[7] E. O’Brien, J. Petrie, W. Littler, M. de Swiet, P. L. Padfield, K. O’Malley, M. Jamieson, D. Altman, M. Bland, and N. Atkins, “The british hypertension society protocol for the evaluation of automated and semi-automated blood pressure measuring devices with special reference to ambulatory systems,” *Journal of Hypertension*, vol. 11, p. S43–S63, 1993.

[8] “Non-invasive sphygmomanometers — part 2: Clinical investigation of automated measurement type,” Association for the Advancement of Medical Instrumentation, Sphygmomanometer Committee, American National Standards Institute, Inc., Tech. Rep., 2013.

[9] G. Lloyd, “Clinical guidelines for measuring/ monitoring blood pressure,” Mersey Care NHS Foundation Trust, Tech. Rep., 2018.

[10] R. A. Reeves, "Does this patient have hypertension?: How to measure blood pressure," *JAMA: The Journal of the American Medical Association*, vol. 273, pp. 1211–1218, 4 1995.

[11] A. Chandrasekhar, M. Yavarimanesh, J. O. Hahn, S. H. Sung, C. H. Chen, H. M. Cheng, and R. Mukkamala, "Formulas to explain popular oscillometric blood pressure estimation algorithms," *Frontiers in Physiology*, vol. 10, 2019.

[12] W. J. W. Bos, E. Verrij, H. H. Vincent, B. E. Westerhof, G. Parati, and G. A. V. Montfrans, "How to assess mean blood pressure properly at the brachial artery level," *Journal of Hypertension*, vol. 25, 2007.

[13] E. Joe. Blood pressure measurements in the icu: Trust only the map in oscillometric devices! [Online]. Available: https://www.youtube.com/watch?v=TG0NcqKeRWE

[14] Bp monitors. British and Irish Hypertension Society. [Online]. Available: https://bihsoc.org/bp-monitors/

[15] M. Forouzanfar, H. R. Dajani, V. Z. Groza, M. Bolic, S. Rajan, and I. Batkin, "Oscillometric blood pressure estimation: Past, present, and future," *IEEE Reviews in Biomedical Engineering*, vol. 8, 2015.

[16] G. Drzewiecki, R. Hood, and H. Apple, "Theory of the oscillometric maximum and the systolic and diastolic detection ratios," *Annals of Biomedical Engineering*, vol. 22, 1994.

[17] M. Ursino and C. Cristalli, "A mathematical study of some biomechanical factors affecting the oscillometric blood pressure measurement," *IEEE Transactions on Biomedical Engineering*, vol. 43, no. 8, pp. 761–778, 1996.

[18] P. K. Lim, S. C. Ng, W. A. Jassim, S. J. Redmond, M. Zilany, A. Avolio, E. Lim, M. P. Tan, and N. H. Lovell, "Improved measurement of blood pressure by extraction of characteristic features from the cuff oscillometric waveform," *Sensors (Switzerland)*, vol. 15, 2015.

[19] M. Forouzanfar, "A modeling approach for coefficient-free oscillometric blood pressure estimation," 2014.

[20] L. A. Geddes, M. Voelz, C. Combs, D. Reiner, and C. F. Babbs, "Characterization of the oscillometric method for measuring indirect blood pressure," *Annals of Biomedical Engineering*, vol. 10, 1982.

[21] M. Ramsey, "Noninvasive automatic determination of mean arterial pressure," *Medical and Biological Engineering and Computing*, vol. 17, no. 1, pp. 11–18, 1979. [Online]. Available: https://doi.org/10.1007/BF02440948

[22] G. W. Mauck, C. R. Smith, L. A. Geddes, and J. D. Bourl, "The meaning of the point of maximum oscillations in cuff pressure in the indirect measurement of blood pressure-part ii," *Journal of Biomechanical Engineering*, vol. 102, pp. 28–33, 1980.

[23] V. Jazbinsek, J. Luznik, and Z. Trontelj, "Non-invasive blood pressure measurements: separation of the arterial pressure oscillometric waveform from the deflation using digital filtering," 01 2005.

[24] V. Jazbinsek, "New algorithm for automatic determination of systolic and diastolic blood pressures in oscillometric measurements." 2016 39th International Convention on Information and Communication Technology, Electronics and Microelectronics, MIPRO 2016 - Proceedings, 2016.

[25] M. Mafi, S. Rajan, M. Bolic, V. Z. Groza, and H. R. Dajani, "Blood pressure estimation using oscillometric pulse morphology," in *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, Aug 2011, pp. 2492–2496.

[26] M. Mafi, S. Rajan, M. Bolic, V. Z. Groza, and H. R. Dajani, "Blood pressure estimation using maximum slope of oscillometric pulses," *Conference proceedings (IEEE Engineering in Medicine and Biology Society. Conf.)*, vol. 2012, p. 3239, 2012.

[27] S. Lee, J.-H. Chang, S. W. Nam, C. Lim, S. Rajan, H. R. Dajani, and V. Z. Groza, "Oscillometric blood pressure estimation based on maximum amplitude algorithm employing gaussian mixture regression," *IEEE Transactions on Instrumentation and Measurement*, vol. 62, no. 12, pp. 3387–3389, 2013.

[28] B. Kneubühler. (2020) Oscillometric blood pressure measurement. [Online]. Available: https://github.com/itsBelinda/obp