

# IMU integration in SHARC buoy

---



Presented by:  
Yusuf Heylen

Prepared for:  
A/Prof. Amit Mishra  
Dept. of Electrical and Electronics Engineering  
University of Cape Town

Submitted to the Department of Electrical Engineering at the University of Cape Town  
in partial fulfilment of the academic requirements for a Bachelor of Science degree in  
Electrical and Computer Engineering

Monday 14<sup>th</sup> October, 2019



## Declaration

---

1. I know that plagiarism is wrong. Plagiarism is to use another's work and pretend that it is one's own.
2. I have used the IEEE convention for citation and referencing. Each contribution to, and quotation in, this report from the work(s) of other people has been attributed, and has been cited and referenced.
3. This report is my own work.
4. I have not allowed, and will not allow, anyone to copy my work with the intention of passing it off as their own work or part thereof.

Signature:   
Yusuf Heylen

Date: Monday 14<sup>th</sup> October, 2019

## Acknowledgments

---

First and foremost all praise is due unto God, for giving me the strength and courage to push through despite all the challenges that I have encountered along the way.

Secondly, I would like to thank A/Prof. Mishra for his patience with me, and his guidance throughout the year.

Jamie Jacobson for all his advice in helping me get footing when I was lost.

Kelsey Kaplan for sharing her knowledge on the *YOST IMU* and going out of her way to allow me to use it.

Stuart Lunt for letting me use the Stewart platform which he was working with and going out of his way to explain how it works to me.

Myrin Naidoo for his emotional support and encouragement when I was faltering. His encouragements injected me with renewed vigour to tackle this project.

To my Mother and Father, for all that they provide me to succeed.

Lastly, to the rest of my family and my friends for all the joy they bring in my life.

## Abstract

---

The Department of Electrical Engineering at the University of Cape Town is in the process of developing a buoy system for use in the Southern Ocean. This is on request from the Department of Oceanography at the University of Cape Town. The final buoy system is to be a low-cost, networked mesh of buoys. A requirement of these buoys is the ability to log key data required in oceanographic studies.

The spectral components of ocean waves are important parameters for use in this field of study. Current methods for acquiring this information comes from the deployment of buoys with inertial measurement units. The acceleration time series produced can be used to calculate these parameters. Buoys currently deployed by other researches in the polar regions use expensive inertial measurement units. The University of Cape Town cannot fund the cost of such inertial measurement units.

This project investigates whether a low-cost inertial measurement unit (MPU6050) is sufficient for deployment in these buoys to measure flexural-gravity waves. A sensor error model for the MPU6050 and a software framework to calculate the wave height displacement time series were developed. The wave height displacement time series calculated by the MPU6050 was compared to a one from a more expensive inertial measurement unit, as well as the expected values.

It was found that the data the MPU6050 yielded was inadequate to be used as the only means of vertical height data collection. Recommendations for improving the reliability of this data are provided.

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Background to the study . . . . .	1
1.2	Objectives of this study . . . . .	3
1.2.1	Problems to be investigated . . . . .	3
1.2.2	Purpose of the study . . . . .	3
1.3	Scope and Limitations . . . . .	4
1.4	Plan of development . . . . .	4
<b>2</b>	<b>Literature Review</b>	<b>6</b>
2.1	Relevant basic wave theory . . . . .	6
2.1.1	Wave spectra . . . . .	8
2.1.2	Wave characterization parameters . . . . .	9
2.1.3	Spectral moments . . . . .	10
2.2	Wave height recording techniques . . . . .	11
2.2.1	Satellite Positioning . . . . .	11
2.2.2	Inertial Measurement Units . . . . .	12

<b>3 Methodology</b>	<b>16</b>
3.1 Hardware used . . . . .	16
3.2 Calibration . . . . .	17
3.2.1 Sensor Error Model . . . . .	17
3.2.2 Calibration Procedure . . . . .	18
3.3 Wave height calculation . . . . .	20
3.3.1 Experimental Tests . . . . .	20
3.3.2 Time Series Analysis . . . . .	21
3.3.3 Calculating displacement time series . . . . .	24
3.3.4 Calculation of Wave characterization parameters . . . . .	25
<b>4 Results</b>	<b>27</b>
4.1 Calibration Results . . . . .	27
4.2 Experimental results . . . . .	28
<b>5 Discussion</b>	<b>30</b>
<b>6 Conclusions</b>	<b>32</b>
<b>7 Recommendations</b>	<b>33</b>
<b>A Realised implementations of algorithms</b>	<b>39</b>
A.1 Despiking implementation . . . . .	39

A.2	Detrending implementation . . . . .	40
A.3	Decimation implementation . . . . .	41
A.4	Spectral moment calculation . . . . .	42
A.5	Other wave characterization parameters calculation . . . . .	43

# List of Figures

1.1	Image of waves in pancake ice [5]	2
1.2	Image of SHARC Buoy designed topology	2
2.1	Image of the types of ocean waves, generation causes and restorative forces. [10]	6
2.2	Image of the formation of ocean waves via small wind eddies	7
2.3	Image of the regions of sea wave formation and its resulting motion [12]	7
2.4	Image of a buoy following a trochoidal motion in a gravity wave [14]	8
2.5	Graph of a Pierson-Moscowitz spectrum for a wind speed of $20ms^{-2}$	9
2.6	Visualisation of trilateration for calculation of sensor position [23]	12
2.7	Image of a capacitive accelerometer[26]	13
2.8	Figure (a) shows the non-orthogonality between the accelerometer's axes, while (b) shows the misalignment between the accelerometer's axis and that of the body frame axis [29]	13
2.9	Figure (a) shows bias error including the random drift and offset, while (b) shows the scaling error effect on the relation of input accelerometer data to that read at the output	14
3.1	Image of the orientations around $z$ axis used in the calibration procedure	18

3.2	Image of the 3D printed calibration cube used to securely fasten IMU in	19
3.3	Image of the department's stewart platform by [36]	20
3.4	Graph of the height and pitch signal sent to the stewart platform in test 3	21
3.5	Image of the effects of detrending and despiking a time series	23
4.1	Graphs of displacement time series for test 3	28

# List of Tables

4.1	Table of the SEM parameters calculated . . . . .	27
4.2	Table of the significant wave heights calculated for each test and IMU . .	28
4.3	Table of the Pearson correlation coefficient between IMUs for each test . .	29

# Chapter 1

## Introduction

### 1.1 Background to the study

The polar regions are at the forefront of research in Oceanographic studies. The changing climate is having a significant impact on the sea ice in these regions. In the Arctic, the sea ice has been in steady decline for the last few years. While more relevantly in the Antarctic, the sea ice has reached record peaks in the period of 2012 to 2014 [1]. It is known that sea ice plays a significant role in the dynamics of the Earth's climate, thus a greater understanding is needed to determine the dynamics of sea ice and its interaction with the surrounding environment. Especially as more powerful storms are expected in the Southern Ocean, with the potential to break up sea ice [2].

Surface waves can penetrate through the ice at the edge of the sea ice pack - a region known as the marginal ice zone. Here there is a significant interaction between the two. With sea ice affecting the dynamics of the ocean waves and the ocean waves affecting its formation. However, the current body of knowledge surrounding the interaction between surface waves and these ice floes in the Southern Ocean is limited [2]. A better understanding of the kinematics of the waves is required. Current methods for obtaining data include using remote airborne LIDAR and satellite imagery or field-based measurement techniques using buoys which are equipped with a Global Positioning System module (GPS) and/or an inertial measurement unit (IMU) [3]. Current buoys are equipped with expensive IMUs which can exceed costs of over a few thousand rands [4]. This prevents the mass deployment of such measurement devices in a mesh-like network.

The University of Cape Town's Department of Oceanography has contracted its department of Electrical Engineering to develop IMU equipped buoys for deployment on ice floes in

### 1.1. BACKGROUND TO THE STUDY

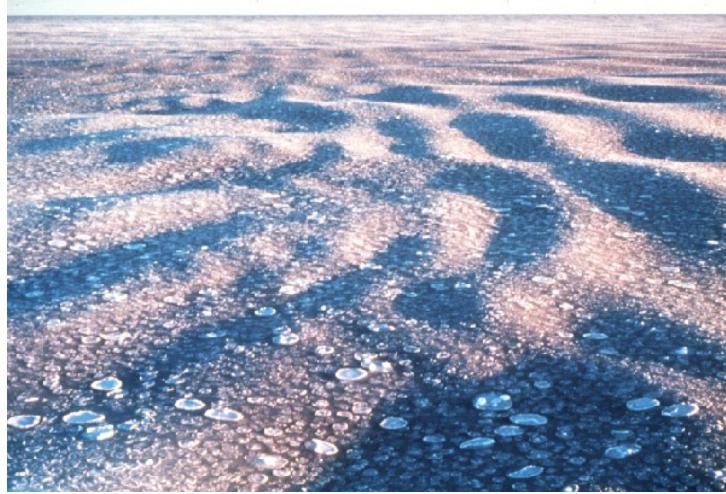
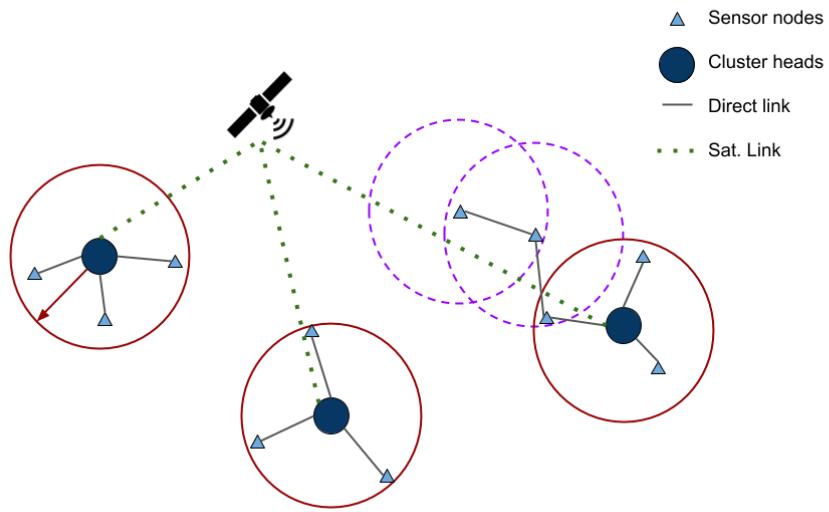


Figure 1.1: Image of waves in pancake ice [5]

the Southern Ocean. Due to funding constraints, this device ought to be cost-effective. The department of Electrical Engineering - hence to be known as “the department” - has designed a low-cost mesh-like network of buoys. The southern hemisphere antarctic research collaboration (SHARC) buoys consist of low-cost ‘daughter’ buoys that gather data and relay this to more expensive ‘mother’ buoys. This topology is seen in figure 1.2. A requirement for the ‘daughter’ buoys is that they are low-cost whilst able to gather useful data. It is from this that the necessitate of investigating low-cost inertial measurement units arose.

### GOAL: Hierarchical Mesh Network of Buoys



3

Figure 1.2: Image of SHARC Buoy designed topology

## 1.2 Objectives of this study

### 1.2.1 Problems to be investigated

This project seeks to investigate the use of a low-cost IMU on the department's existing buoys infrastructure for the calculating of relevant wave parameters. This first entails the analysis of the error of such an IMU, and the development of mechanisms to reduce this. It is to be determined whether such a low-cost IMU is a viable alternative to a more expensive one, which has not been done in existing literature [6], [7]. This will require the development of tools for the extraction of key information from IMU data produced in the form of the wave height displacement time series, the significant wave height as well as other spectral parameters.

### 1.2.2 Purpose of the study

Previous ice floe buoys that the department of Oceanography have deployed, have been limited in capabilities - being equipped only with GPS and temperature sensors. A purely GPS equipped sensor node is limited to being only able to measure the drift of the ice floes that the buoys are attached to. While GPS does have the ability to measure vertical height its accuracy is limited [8]. The effects of sea ice on ocean waves is of interest to the Department of Oceanography. Better understanding requires the knowledge of the vertical displacement of these buoys. Thus the current system is inadequate for the study of the complexities of the Southern Ocean.

Adding an IMU would aid in the research in this field. Current systems with IMUs deployed in polar conditions are expensive and thus it needs to be investigated whether a low-cost module can be used. However, the use of such a low-cost IMU would necessitate the mitigation of its error which requires the development and implementation of a sensor error model. In addition, the development of software to extract the wave height displacement and significant wave height, from the measured acceleration time series, needs to be done. As well as extracting key spectral parameters for the use of sending via satellite. Hence the objectives of this project can be stated as:

1. Test whether a low-cost IMU-based sensor is sufficient for addition to the existing buoy

2. Develop a sensor error model for the low-cost IMU
3. Develop software to extract the displacement time series from measurements
4. The development of software to acquire key parameters for Oceanographic studies

## 1.3 Scope and Limitations

The project entails the investigation of a low-cost wave height calculation system on an existing buoy system. It includes the comparison of a low-cost IMU sensor with an expensive one. From which the development of an algorithm to extract the significant wave height will be developed. This project does not look into the *implementation* of the IMU onto the buoy, but rather the use of the IMU in and of itself - whether it is accurate enough to be used for these scientific purposes and the development of error mitigation tools. As well as the development of tools from which to calculate ocean wave characterization parameters from the IMUs data.

The project is limited to the constraints given by the team working on the buoy in the department. These constraints limit which IMUs can be used. Other IMUs are not considered as these are the ones that the department has on hand. The project spans a 12 week semester which coincided with other course work. This limited the development of further error mitigation tools. Lastly, the testing equipment is limited to that which is available in the department.

## 1.4 Plan of development

The report starts with a brief introduction to ocean wave theory that is required for this report. Relevant mathematical models of ocean waves are looked at. The parameters that describe these ocean waves are then given. Following which, some approaches to calculating a buoy's vertical displacement are then reviewed. This includes the functioning of an inertial measurement unit, its the sources of error and calibration techniques are laid out.

The methodology describes the hardware that was used. The calibration of the MPU6050 is explained in detail with the sensor error model and calibration procedure given. Following this, the experimental process to calculate the wave height displacement, significant

#### 1.4. PLAN OF DEVELOPMENT

wave height and other wave characterization parameters are given. This includes the experimental setup.

Then, the results from calibrating and the experiment are provided and conclusions from these results are drawn.

These are discussed in the following chapter.

Lastly, recommendations are made for inertial measurement integration into SHARC buoy.

# Chapter 2

## Literature Review

### 2.1 Relevant basic wave theory

As described in [9] there are numerous mechanisms in which ocean waves are generated. These include: wind blowing over the surface of the ocean, seismically generated waves (for example from earthquakes,) and waves generated from gravitational attraction (to the moon for example.) Once waves have been generated, they propagate through the ocean with gravity acting as a restorative force. Figure 2.1 shows the different types of waves, their causes and the restorative forces acting upon them.

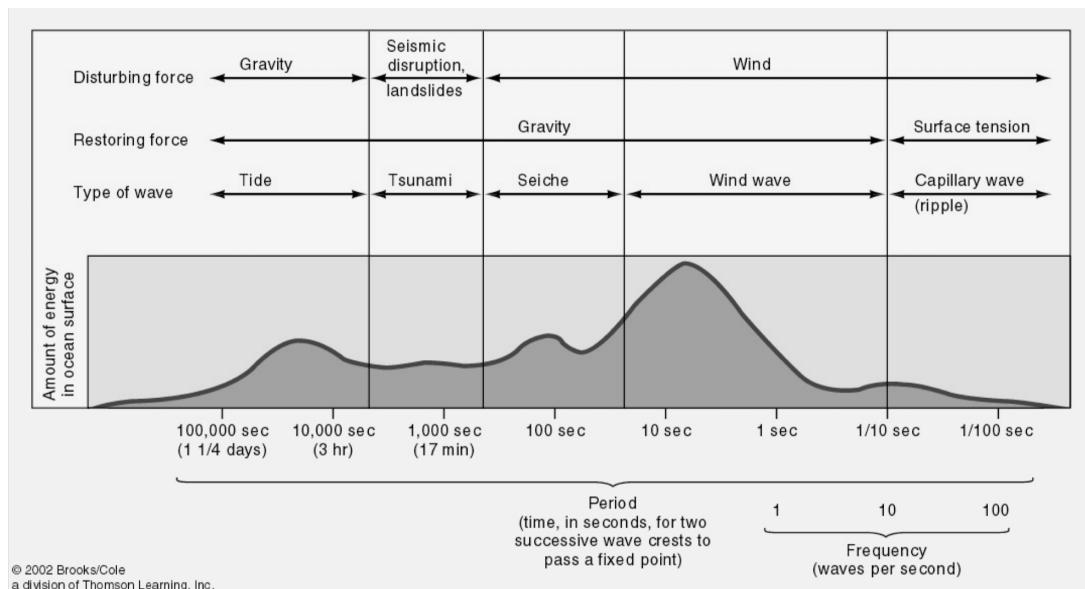


Figure 2.1: Image of the types of ocean waves, generation causes and restorative forces. [10]

Of primary interest in this project are the types of waves that are found near the marginal

## 2.1. RELEVANT BASIC WAVE THEORY

ice zone. The waves found here are mainly wind generated gravity-waves - this comes from the numerous storms that are found in the region [2]. As the wind blows over the surface of the ocean, the friction between the wind and waves leads to a corresponding drop in pressure near the surface of the water. The pressure difference results in the formation of circular wind eddies, which has the effect of generating a transverse wave form on the water's surface. This is seen in figure 2.2. A transfer of energy from the wind to the wave occurs as a result. Initially these ripples are small, but these ripples contribute towards a positive feedback loop in the transferral of energy from the wind to the waves and thus the growth of these waves. These processes are described in [11].

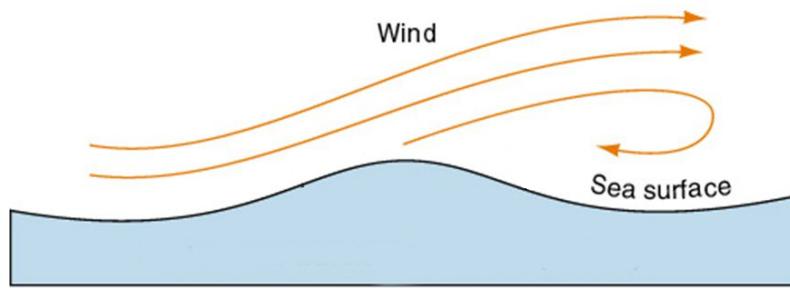


Figure 2.2: Image of the formation of ocean waves via small wind eddies

The phase when the ocean waves are experiencing growth is known as the transient state. This occurs in the “fetch” area, where the waves are exposed to the largest amount of wind. These wave forms are effectively “pushed” by the wind to create longitudinal motion. Figure 2.3 shows the longitudinal propagation of ocean waves as well as the region of their formation.

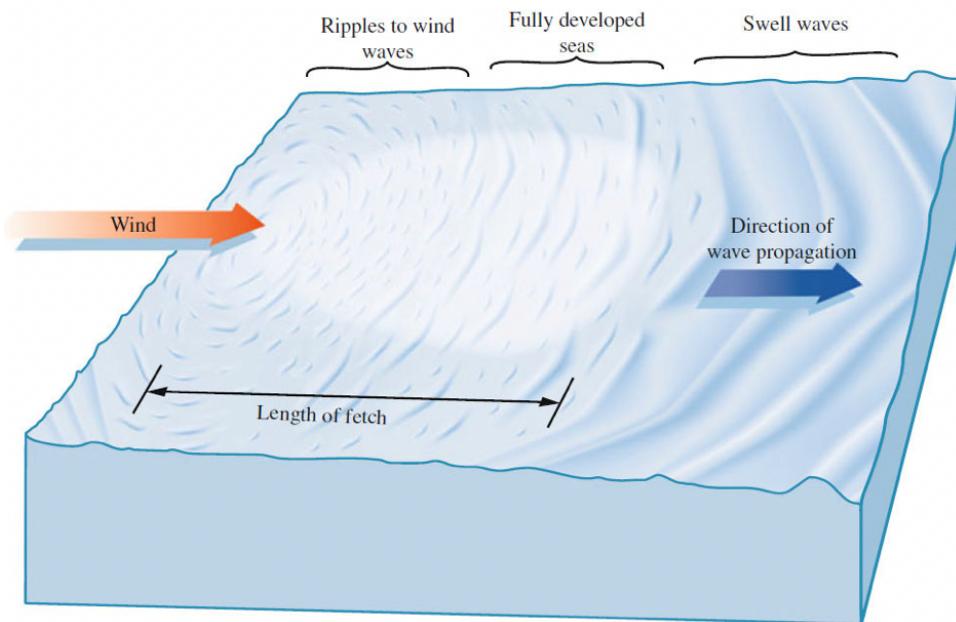


Figure 2.3: Image of the regions of sea wave formation and its resulting motion [12]

## 2.1. RELEVANT BASIC WAVE THEORY

Sea ice is known to contribute to the restorative forces of ocean waves [5]. This has the effect of changing the dynamics of the waves. These waves are called flexural-gravity waves, due to the combined action of the ice (flexural tension) and gravity acting as restorative forces. As one travels deeper into the sea ice pack, different wave dynamics will be found. This is as the density and size of ice floes increase deeper into the ice pack. The exact dynamics - as mentioned earlier - are not well understood however, it is known that ice has both scattering and amplitude altering effects on ocean waves. The larger the ice floes the greater the amplitude suppressing effects are [5].

Due to the complexity of modeling flexural-gravity waves and for this project, only a simplified model of wind-generated gravity-waves will be considered. This is a reasonable simplification as at the edge of the ice pack closest to the open ocean, the dynamics are similar to that of standard gravity-waves [5]. The gravity-waves found before the marginal ice zone are ordinary gravity-waves. These have a wave period of around 1s to 30s [9]. A gravity-wave in deep and medium depth waters follows a trochoidal pattern [13]. This comes from the combination of the transverse and longitudinal motion of the wave. This causes a particle on the wave to follow a circular orbit. As in [14], one can consider a buoy (or small ice floe) to be such a particle. Figure 2.4 shows this motion.

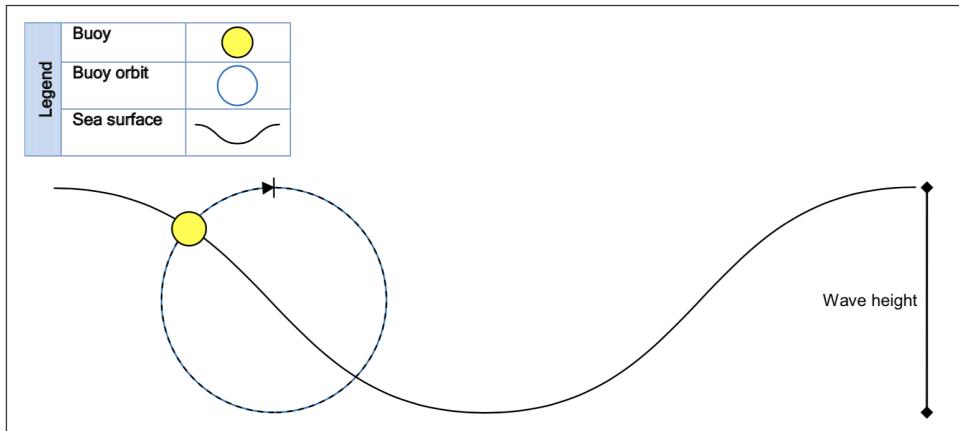


Figure 2.4: Image of a buoy following a trochoidal motion in a gravity wave [14]

### 2.1.1 Wave spectra

Ocean waves can be modelled as the sum of different sinusoidal waves, with varying amplitudes, phases, frequencies and directions [12]. The distribution of energy amongst these sinusoidal components is known as the energy spectrum. Various models of the spectra of gravity waves have been developed, the most noteworthy being that developed by [15]. This is known as the Pierson-Moscowitz spectrum. It assumes that the gravity-waves have reached an equilibrium state after being extensively blown on by wind. When

## 2.1. RELEVANT BASIC WAVE THEORY

ocean waves are in equilibrium they are known as being in the fully developed sea state. This model was expanded upon by researchers in the Joint North Sea Wave Observation Project which is able to account for non-fully developed seas [16]. The Pierson-Moscowitz power spectral density is given in equation 2.1.

$$S(\omega) = \frac{\alpha g^2}{\omega^5} \times e^{-\beta(\frac{\omega_p}{\omega})^4} \quad (2.1)$$

With  $\omega_p$  being the peak frequency,  $\alpha = 0.0081$  and  $\beta = 0.74$ . Figure 2.5 shows a plot of a Pierson-Moscowitz spectrum:

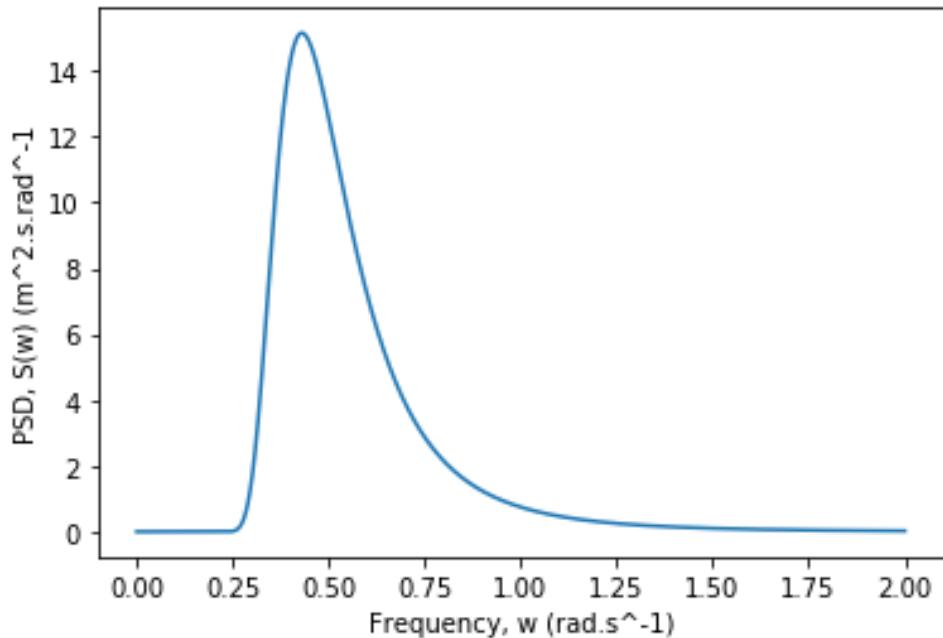


Figure 2.5: Graph of a Pierson-Moscowitz spectrum for a wind speed of  $20ms^{-2}$

### 2.1.2 Wave characterization parameters

#### Significant wave height

Of particular interest in oceanographic studies is the significant wave height of waves. There are three ways to calculate the significant wave height. The significant wave height can be given by the average of the highest one-third of waves measured in a given period, where the height is recorded as the trough to crest height [17]. The significant wave height calculated in this way is labelled as  $H_{1/3}$ . The significant wave height is also given by four times the standard deviation of the wave height displacement time series. The significant

wave height calculated in this manner will be labelled as  $H_t$  such that  $H_t = 4\sqrt{m_0}$  where  $m_0$  is the variance of the wave height displacement time series acquired during the wave acquisition period [18]. This holds only for waves with frequency components in a narrow range [19]. This assumption generally holds for medium and deep water waves [17] which are the waves generally experienced by ice floes in the southern ocean, as they experience high wind speeds which narrows the spectral band through equation 2.1.

### 2.1.3 Spectral moments

Another approach to calculate the significant wave height is to calculate it from the power spectrum. As mentioned in [17], the significant wave height can be calculated by finding the power in the wave by integrating the power spectrum. The significant wave height calculated using the power spectrum is given in equation 2.2. Where  $S(\omega)$  is the power spectrum. The power spectrum can be calculated by taking the Fourier transform of the autocorrelation of the real valued function describing the ocean wave displacement. If discrete measurements of the wave displacement is in hand, the power spectrum can be estimated via a periodogram [20].

$$H_s = 4 \times \sqrt{\int_0^\infty S(\omega) d\omega} \quad (2.2)$$

The variance of the wave height displacement time series  $m_0$ , is also the zeroth moment of the power spectral density. Where the  $n$ th moment is given in equation 2.3.

$$m_n = \int_0^\infty f^n S(f) df \quad (2.3)$$

These spectral moments (calculated via equation 2.3) allow for the calculation of key statistical results of the wave [6]. This include the wave energy period,  $T_e$  which is defined as the ratio of the first negative spectral moment to the zeroth spectral moment: i.e.  $T_e = \frac{m_{-1}}{m_0}$ . From this - and the spectral calculation of the significant wave height  $H_s$  (given in equation 2.2)- the omnidirectional wave power  $J$  can be found. This is defined in equation 2.4, where  $\rho$  is the density of the sea water [12] at the site. This quantity is the primary mechanism to assess the wave energy of a wave.

$$J = \frac{\rho g^2}{64\pi} H_s^2 T_e \quad (2.4)$$

In addition to these quantities the spectral moments can be used to determine the spectral bandwidth  $\epsilon_0$ , which is a measure of the relative spread of energy with the wave frequency. It is given by  $\epsilon_0 = \sqrt{\frac{m_0 m_{-2}}{m_{-1}^2} - 1}$ . While an estimate of the mean zero-crossing period of the waves  $T_z$  can be calculated via  $T_z \approx \sqrt{\frac{m_0}{m_2}}$  [12].

## 2.2 Wave height recording techniques

In order to calculate spectral moments and power spectrum, a recording of the wave displacement time series is needed. Methods for recording the wave displacement series can either be done directly or indirectly. Direct approaches include using satellite positioning. Indirect measurement can be done through measurement of the buoys acceleration through an inertial measurement unit, or through surface tension measurement (as done in [21]) which will not be looked at in this report due to its complex nature.

### 2.2.1 Satellite Positioning

The GPS system is a network of satellites that transmits accurate information about the satellites time and position. A sensor receives this information and calculates its position using trilateration. This process involves calculating the distance of the sensor to each of four satellites. From this it can determine its relative position, including its height [22]. This is seen in figure 2.6.

GPS has a 4m root mean square accuracy 95% percent of the time [8]. This accuracy is based on an ellipsoid model of the world, the World Geodetic System (WGS84.) However, in reality, the true vertical height of the earth can differ significantly from this which makes the use of GPS for vertical height measurement challenging.

The accuracy of GPS also strongly depends on the availability of satellites overhead the sensor's current location. This poses a further challenge for sensors in the polar regions as satellite availability is low. Hence it will not be considered for measurement of the wave height.

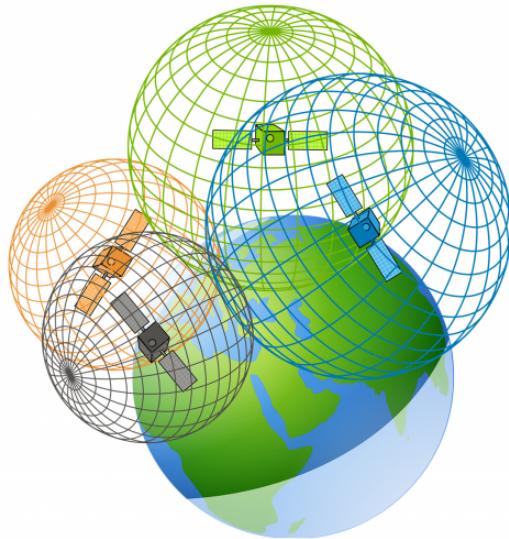


Figure 2.6: Visualisation of trilateration for calculation of sensor position [23]

### 2.2.2 Inertial Measurement Units

Inertial Measurement Units are comprised of an accelerometer, a gyroscope and sometimes a magnetometer [24]. Together these devices comprise an inertial measurement unit. It is named such as the data recorded allows for the computation of the device's position using a technique called dead-reckoning. Of particular interest is the accelerometer, it produces a time series of accelerations in a particular axis. Section 2.1.2 deals with the conversion of this time series into significant wave heights.

#### Underlying Technology

The vast majority of modern inertial measurement units are comprised out of small microelectromechanical systems (MEMS,) these are named so as they are comprised of mechanical and electronic systems interacting to measure a specific phenomenon [25]. IMUs are comprised of these devices that measure acceleration and the angular rate of rotation - a gyroscope. A MEMS accelerometer is comprised of a small mass of capacitive plates on a spring. As the device experiences an acceleration, the mass moves in a particular direction changing the capacitance between the mass and the fixed plates. This change in capacitance corresponds to a force which the mass experiences. This is used to compute an acceleration in the corresponding axis on which the mass lies via Newton's second law. The device is seen in figure 2.7. There are three such devices - each lying on an orthogonal axis to each other.

## 2.2. WAVE HEIGHT RECORDING TECHNIQUES

Similarly, a gyroscope measures the Coriolis force that a mass experiences via a change of capacitance.

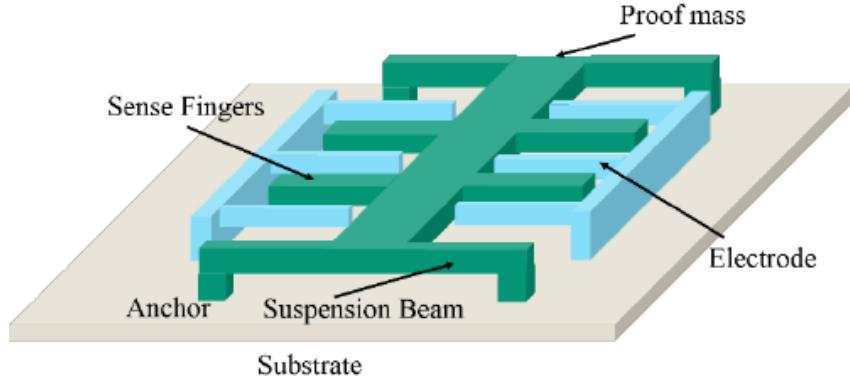


Figure 2.7: Image of a capacitive accelerometer[26]

### Error in IMUs

Accelerometers suffer from numerous amounts of errors. Some of the error comes during the manufacturing process, where the accelerometer axes may not be perfectly orthogonal to each other, this leads to error known as cross-axis sensitivity [27]. Another source of error comes from the misalignment between the accelerometer's axes and the body frame axes of the device. This leads to error in the measurement of the acceleration along those particular axes. However, body frame misalignment errors are hard to identify from cross-axis errors and thus are often ignored [28]. These two errors are seen in figure 2.8.

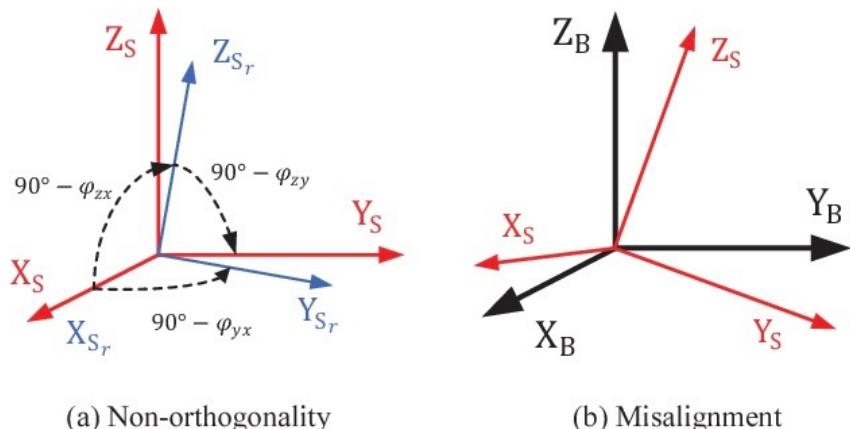


Figure 2.8: Figure (a) shows the non-orthogonality between the accelerometer's axes, while (b) shows the misalignment between the accelerometer's axis and that of the body frame axis [29]

The readings from an IMU are also affected by a scaling error and bias error [29]. The

## 2.2. WAVE HEIGHT RECORDING TECHNIQUES

scaling error is a misalignment of the projection of the input data to that read at the output. Whereas a bias error comprises a random drift error and an offset error. With both adding a fixed value between the input data and that at the output (this can be negative.) The random bias drift error is ignored during calibration as it cannot be accounted for due to its random nature. It depends on the designed sensitivity of the device. Figure 2.9 shows the effects of scaling and bias error between the input and output. These errors can be asymmetrical being different for positive and negative sensor readings [29].

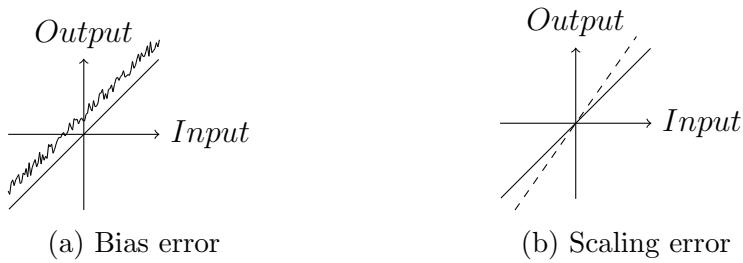


Figure 2.9: Figure (a) shows bias error including the random drift and offset, while (b) shows the scaling error effect on the relation of input accelerometer data to that read at the output

High-end IMUs are factory calibrated, the cost for this calibration is high. For example, the IMU used in [30], the *VN-100*, has a purchasing option to calibrate the device for usage in low and high temperature, adding over ZAR 45000 to the price tag.[4] On the other hand, lower cost IMUs are poorly calibrated - hindering their performance [28].

### Calibration techniques

Before calibration can begin a sensor error model (SEM) for the IMU needs to be chosen [31]. The SEM will take into account the various errors inherent in an IMU. The more errors taken into account the more complex the SEM will be.

There are numerous ways IMUs are calibrated [32]. They are often variations of the following:

1. Drop calibration - where the IMU is dropped from a set of predetermined heights. Here the IMU will only experience the acceleration due to gravity (and negligible drag force) from which the error can be calculated.
2. Turntable calibration - where the IMU is rotated through a set of different orientations. The gyroscopes measurements are compared to the set rotation rate of the turntable

## 2.2. WAVE HEIGHT RECORDING TECHNIQUES

while the accelerometer readings are compared to that due to gravity. [29]

3. Motion calibration - where the IMU is subjected to linear motion in a fixed direction. This can be done with a linear stage.

# Chapter 3

## Methodology

### 3.1 Hardware used

Two IMUs are to be considered. The first IMU is the low-cost MPU6050 (GY521) 6-axis by InvenSense, while the second is the higher quality *Yost Labs 3-Space™ Sensor* (YOST-IMU) which is around two-hundred times more expensive. The MPU6050 was manipulated using an Arduino UNO but will be eventually implemented on an STM32 Nucleo microcontroller.

#### MPU6050

Important parameters to note of the MPU6050 [33] include:

- Cross-axis sensitivity of  $\pm 2\%$
- Nonlinearity of 0.2% - which is a measure of the deviation from perfect linearity between the accelerometer output and input[34]
- Noise performance of  $400 \mu g/\sqrt{Hz}$  - a measure of how much noise is present in accelerometer readings of a set measurement bandwidth

#### YOST-IMU

Important parameters to note of the YOST-IMU [35] include:

- Noise performance of  $99 \mu\text{g}/\sqrt{\text{Hz}}$

The other parameters such as cross-axis sensitivity and nonlinearity were not given in the datasheet.

## 3.2 Calibration

Due to its low-cost nature, the MPU6050 is poorly calibrated under factory conditions. To deduce its fitness to be deployed, the sensor first has to be properly calibrated. As mentioned earlier, a sensor error model for the IMU needs to be chosen. Once one has been chosen, its relevant parameters need to be calculated. This is done subsequently.

### 3.2.1 Sensor Error Model

The SEM to be used in this project will be using a modified one as presented in [31] and [29], this model takes into account the bias offset error, scaling error, sensor-axis misalignment to body-axis error and non-orthogonality between the accelerometer axis. As in [31] this project will not be considering the sensor-axis misalignment to body-axis error. The SEM assumes that the bias offset error and scaling error are symmetrical. As the gyroscope is not necessary for the calculation of significant wave height, only the calibration of the accelerometer is looked at. Thus, considering the above, the SEM for the accelerometer can be given by:

$$\mathbf{v}_a = \mathbf{N} \mathbf{S} (\mathbf{u}_a - \mathbf{b}_o)$$

Where  $\mathbf{u}_a$  is a vector of the three raw accelerometer readings,  $\mathbf{b}_o$  is a vector of the bias offset errors along each axis.  $\mathbf{S}$  is a  $3 \times 3$  matrix of scaling factors such that:

$$\mathbf{S} = \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & s_z \end{bmatrix}$$

While  $\mathbf{N}$  is a  $3 \times 3$  matrix that corrects for the error due to the non-orthogonality of the accelerometer axis. Given by:

$$\mathbf{N} = \begin{bmatrix} 1 & 0 & 0 \\ \varphi_{xy} & 1 & 0 \\ \varphi_{zx} & \varphi_{zy} & 1 \end{bmatrix}$$

With  $\varphi_{yx}$ ,  $\varphi_{zx}$  and  $\varphi_{zy}$  being the small-angle errors between the theoretical orthogonal frame and the physically realised sensor axes.

In total nine calibration parameters need to be determined. Three for the bias offset error, three for the scaling factor error and three for the non-orthogonality error.

### 3.2.2 Calibration Procedure

The calibration procedure that follows is a similar technique used in [29] and [31]. It involves reading the accelerometer values in static conditions under different rotations. In [31] it is recommended that at least 24 different orientations ought to be used. The 8 orientations around the  $z$  axis are seen in figure 3.1. These positions are then repeated around the  $y$  and  $x$  axis. Note, the precise knowledge of each orientation is not needed - the ones presented are served as a guideline. As in [29], a 3D printed cube was made to fit the MPU6050 securely in. This is seen in figure 3.2. Before any readings take place, the accelerometer needs to be left idle for a while (between 5 and 15 min is ideal) to eliminate thermal effects on the readings.

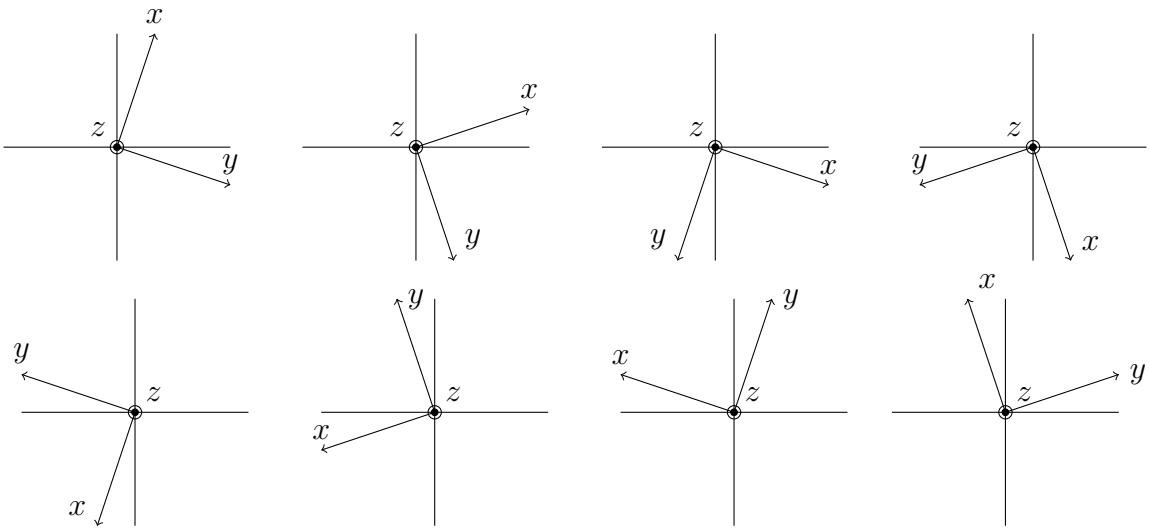


Figure 3.1: Image of the orientations around  $z$  axis used in the calibration procedure

An average of acceleration readings for each of the orientations is taken. Then the root mean square error - seen in equation 3.1 - is minimised using the Broyden-Fletcher-Goldfarb-Shannon (BFGS) algorithm, to get an estimate of each of the 9 calibration

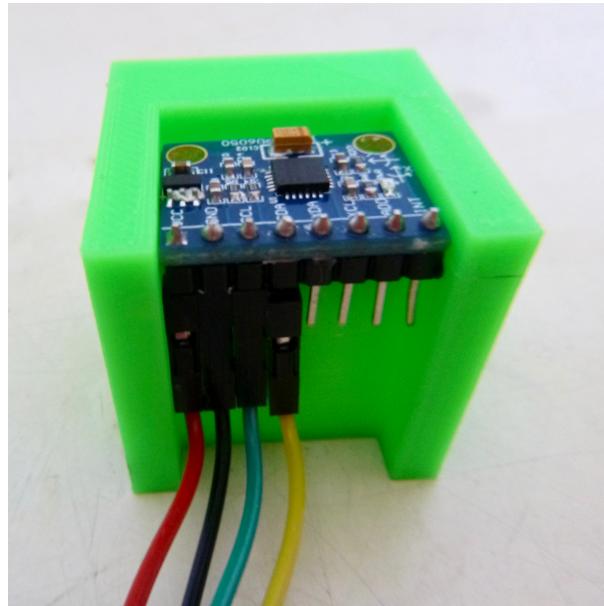


Figure 3.2: Image of the 3D printed calibration cube used to securely fasten IMU in parameters of the SEM. This was implemented in Python3 using the `scipy.optimize` library.

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (||\mathbf{NS}(\bar{\mathbf{u}}_a(i) - \mathbf{b}_a)|| - G)^2}{n}} \quad (3.1)$$

With  $G$  the magnitude of the gravity vector,  $i$  each  $i$ th orientation reading,  $n$  the total number of rotations used and  $\bar{\mathbf{u}}_a(i)$  the averaged acceleration vector reading for the  $i$ th rotation. Note, the root mean square error for the *uncalibrated* readings is as defined in 3.1 but with  $N$  and  $S$  equal to the identity matrix and  $b_a$  the zero vector.

In this project, ten thousand readings are taken for each rotation. The code developed and used are available on GitHub [here](#).

A detailed list of the calibration steps used are given below:

1. Place the MPU6050 securely in the calibration cube
2. Connect the MPU6050 to an Arduino UNO using I2C and wait 5 - 15 min for it to warm up
3. Load the calibration script and open up a serial monitor
4. Place the cube in a static position

5. Repeat the previous step for 24 (or more) different rotations
6. Take down the averaged acceleration vectors and save in a text file
7. Run the Python3 script
8. Record the SEM parameters

## 3.3 Wave height calculation

Once the MPU6050 has been calibrated, it can be used to calculate the significant wave height under controlled conditions to deduce it's fitness. First data needs to be collected in a controlled environment. The acceleration time series collected needs to be adequately conditioned before it can be used to calculate the significant wave height using the two methods laid out in section 2.1.2. Both IMUs were tested for comparison purposes. All the following methods were implemented in Python3 and are available here.

### 3.3.1 Experimental Tests

The department's Stewart platform designed by [36] was used to simulate gravity-waves as described in section 2.1. The platform can move in the  $x$ ,  $y$  and  $z$  directions as well as control its pitch, roll and yaw - making it ideal for this purpose. The platform is seen in figure 3.3.

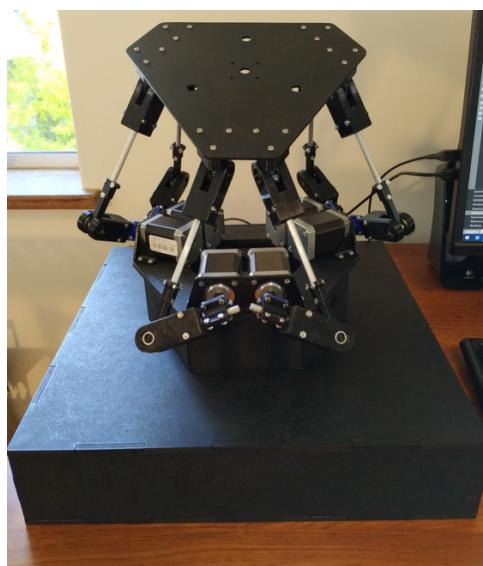


Figure 3.3: Image of the department's stewart platform by [36]

### 3.3. WAVE HEIGHT CALCULATION

Three tests were performed. All three sent a sine wave with A maximum and minimum height of 200mm and 110mm as the input to the  $z$  (height) value of the platform. These values were constrained by the limits of the Stewart platform. In the first test a 10s period sine wave for 90s was used, while in the second a 20s period wave for 100s was sent. The first two tests were chosen so that only the acceleration in the  $z$  direction ( $a_z$ ) were experienced by the IMUs. There are two reasons for this. The first is as [6] found that using only the  $a_z$  readings were better than correcting for the tilt of the ice floe - for tilts less than 50° from horizontal - and no valid ones were experienced in their infield deployments in the Antarctic. Secondly, by isolating the acceleration experienced in one axis, the sensitivity differences of the IMUs in one direction can be directly compared.

In the last test, a varying  $\pm 20^\circ$  pitch was sent into the Stewart in addition to the same parameters in the second test. This pitch followed a cosine wave with the same period. This was chosen as it is the derivative of the sine wave and thus simulates the deployment platform being perpendicular to the ocean wave. Figure 3.4 shows the height signal and pitch signal sent to the Stewart platform for this test.

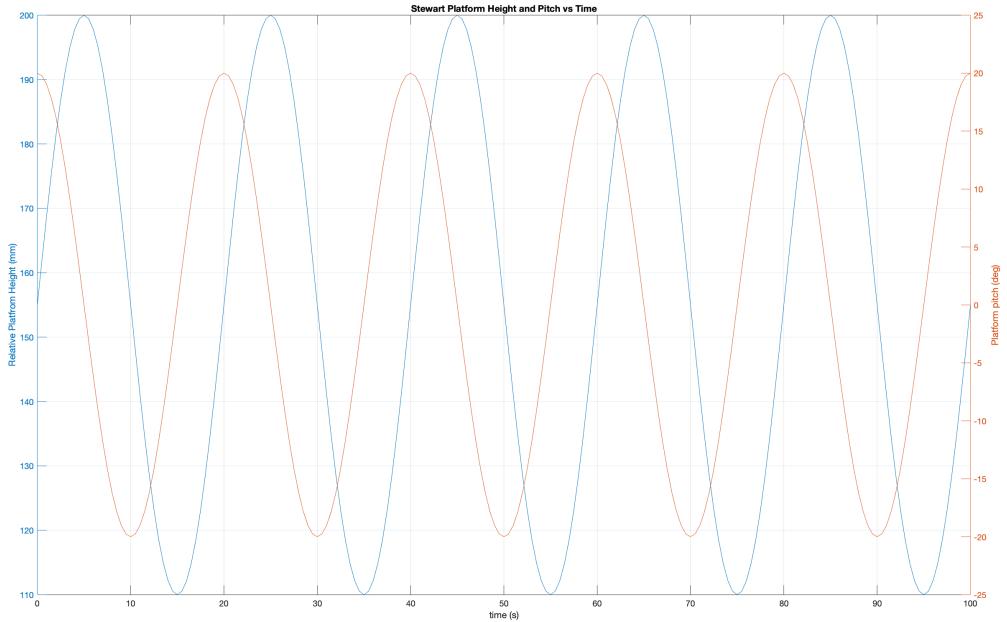


Figure 3.4: Graph of the height and pitch signal sent to the stewart platform in test 3

#### 3.3.2 Time Series Analysis

The acceleration time series of the two IMUs were used to calculate the displacement time series. Before this could be done, the signals had to be conditioned. These conditioning

steps are similar to those performed by [6]. Then the conditioned acceleration time series of each IMU were compared to each other for each test. The displacement time series were calculated and from this the significant wave heights via both methods. Each IMU's significant wave height was compared to each other, as well as with the significant wave height of the height signal sent to the Stewart platform.

### Despiking

The acceleration time series might have unlikely recordings (spikes) in it that need to be removed. A method of doing this is by removing any values that exceed a chosen standard deviation threshold. This method has an issue as the standard is itself calculated with the spikes included. This is avoided in [6] by selecting a very high standard deviation threshold of six, as Chebyshev's inequality states that 97.3% of data falls below six standard deviations of the mean.

However, this project uses the method recommended by [17]. Any spikes, that are three standard deviations above the mean are removed, over three iterations. This mitigates the issue of the standard deviation being affected by the spikes. This method was chosen over that in [6] as the laboratory tests described earlier follow uniform motions and hence such values are more unlikely. In addition due to the MPU6050 having more noise than the YOST IMU, a conservative approach is warranted. Values removed were replaced with linearly interpolated ones, by summing the values of the reading before and after and dividing it by two. This is seen in algorithm 1.

---

**Algorithm 1:** Remove unlikely values in series

---

**Input:** An array  $A$  of one dimensional acceleration readings

**Result:** The array  $A$  with unlikely occurrences replaced

```

for  $i \leftarrow 1$  to 3 do
    foreach  $a_n \in A$  do
        if  $\neg(\bar{a} - 3\sigma_A < a_n < \bar{a} + 3\sigma_A)$  then
             $a_n \leftarrow \frac{a_{n+1} + a_{n-1}}{2}$ 
        end
    end
end
```

---

This algorithm is implemented in Python 3 in appendix A.1.

## Detrending

Underlying trends in the time series would affect the calculations of the displacement of the sensor. Thus before integrating the series, trends were removed. As recommended by [20], the signal was detrended via algorithm 2. This is equivalent to applying a single-pole, high-pass filter with time constant  $\tau = \frac{1/f_s}{(1-k)}$ . The value of  $k$ , chosen such that  $\tau$  is at least 100s.

---

**Algorithm 2:** Detrend series

---

**Input:** An array  $A$  of one dimensional acceleration readings,  $f_s$  the sampling frequency  
**Result:** An array  $A^*$  of detrended values

```

 $s_{n-1} \leftarrow 0$ 
 $k \leftarrow 1 - \frac{1/f_s}{100}$ 
foreach  $a_n \in A$  do
     $s_n \leftarrow (a_n - \bar{a}) + k \times s_{n-1}$ 
     $A_n^* \leftarrow (a_n - \bar{a}) - (1 - k) \times s_n$ 
end
```

---

This algorithm is implemented in Python 3 in appendix A.2.

Unlike [6] after the signal was detrended, it was not despiked again. This is as the more stringent despiking method ensures that unlikely values that might have been missed due to a trend are removed. Figure 3.5 shows the final effects of detrending and despiking an acceleration time series.

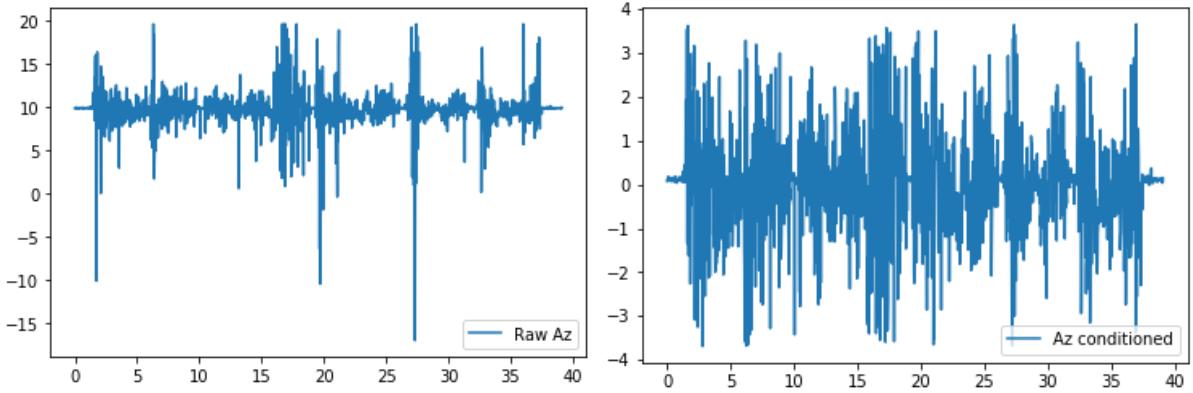


Figure 3.5: Image of the effects of detrending and despiking a time series

## Filtering

Once the signal is conditioned a low pass filter is applied to eliminate any high-frequency noise. This was done using an analogue first-order Butterworth filter with a cut-off frequency at  $8\text{Hz}$ , as done in [6].

## Oversampling and Decimation

By oversampling the desired (acceleration) signal and decimating it down to the required frequency, the signal to noise ratio can be improved [37]. It is recommended in [20] to oversample at  $640\text{Hz}$  and decimate down to  $2\text{Hz}$ . Prior to decimation a low-pass filter needs to be applied (this is equivalent to averaging the signal). A downside with using this method is that it greatly increases the computational load [37]. To minimize this [6] applied a two stage decimation. First the signal is averaged with a second order low-pass Butterworth filter with a cut-off frequency of  $1\text{Hz}$ , the signal is then decimated down to  $8\text{Hz}$  by a factor of 80. Following which, the signal is passed through another second order Butterworth low-pass filter with a cut-off frequency of  $0.5\text{Hz}$  and then decimated down to  $2\text{Hz}$  by a factor of 4.

This functionality was not implemented at the time of availability of the Stewart platform. Thus the signal filtered at  $8\text{Hz}$  was used to calculate the displacement time series as well as for comparison between the two inertial measurement units. However this functionality is available in the software framework, the python 3 implementation is seen in appendix A.3. This does require the installation of the `scipy.signal` python library.

### 3.3.3 Calculating displacement time series

To calculate the significant wave height from the acceleration time series first the displacement time series has to be calculated. Acceleration is the double-time derivative of displacement  $a(t) = \frac{d^2s}{dt^2}$

Thus to recover the displacement of the accelerometer, the acceleration time series needs to be integrated twice. However, integration amplifies errors in the time series and thus double integration of a measured acceleration time series quickly diverges from the true displacement value, so careful care needs to be taken to minimize the errors produced.

Taking the discrete Fourier transform (DFT) of a time series and multiplying it by  $\frac{1}{j\omega}$ , is

equivalent to integrating the series in the time domain [38]. Hence, by multiplying the DFT twice by  $\frac{1}{j\omega} \times \frac{1}{j\omega} = \frac{-1}{\omega^2}$  and taking the inverse Fourier transform, the double integral of a time series can be obtained.

This is essentially the property that is exploited in [6] and [30]. Due to the double integration increasing low frequency noise, a high-pass filter is needed to suppress this. To do this a half-cosine taper function is used as a windowing function, which suppresses these low frequency components, while preventing a sudden cut-off. This, together with multiplying desired components by  $\frac{-1}{\omega^2} = -\frac{1}{(2\pi f)^2}$  gives the frequency weighting function in equation 3.2, where  $f_n$  is the Nyquist frequency and  $f_1 = 0.02Hz$  and  $f_2 = 0.03Hz$  - so chosen as they are suitable for waves with a period between  $4s - 20s$ , this is described in [20].

$$H(f) = \begin{cases} 0 & \text{for } 0 < f < f_1 \\ \frac{1}{2}[1 - \cos(\pi \frac{f-f_1}{f_2-f_1})][-\frac{1}{(2\pi f)^2}] & \text{for } f_1 \leq f \leq f_2 \\ -\frac{1}{(2\pi f)^2} & \text{for } f_2 \leq f \leq f_n \end{cases} \quad (3.2)$$

Once each frequency bin has been multiplied by its respective weight, the result is inverse Fourier transformed to get the height time series i.e.  $h(t_i) = IFFT[H(f_i)FFT(a_z)]$ .

### 3.3.4 Calculation of Wave characterization parameters

#### Calculation of significant wave height

Using the two methods laid out in section 2.1.2, the significant wave height was calculated for each test and IMU. The theoretical wave height (that sent as a signal to the Stewart platform,) was also used to calculate a benchmark significant wave height. These values were then compared.

From the displacement time series, the significant wave height can be calculated. This is done by calculating the standard deviation of the height time series and multiplying it by 4 (as the standard deviation is equal to the square root of the variance)  $H_t = 4std(\eta)$ . The power spectral density (PSD) was not calculated by taking the Fourier transform its autocorrelation, rather the power spectral density was estimated with the periodogram

as in [6]. The significant wave height via the spectrum is

$$H_s = 4 \times \sqrt{\Delta f \sum_{i=1}^{1024} P(i)} \quad (3.3)$$

Where  $P(i)$  is the periodogram estimate of the PSD and  $\Delta f$  is the frequency difference of each spectral component.

### **Calculation of spectral moments and other parameters**

The framework to calculate the  $n$ th degree moment was developed. While these values are not necessary for the usage of comparison, the spectral moments are important parameters for the characterization of an ocean wave. This was explained in section 2.1.2. The software to do so was implemented in python 3, it requires the `scipy.signal` library and is seen in appendix A.4. The method used to calculate these are as those presented in [20].

From the spectral moments the wave energy period  $T_e$ , the spectral bandwidth  $\epsilon_0$  and the estimate of the mean zero-crossing period of the waves  $T_z$  can be calculated. This was laid out in section 2.1.2. Likewise these parameters where not necessary for comparison purposes. The code to calculate these parameters are given in appendix A.5.

### **Comparison**

Finally, the Pearson cross-correlation coefficient defined as  $\rho = \frac{cov(A,B)}{\sigma_A \sigma_B}$  was calculated for all the tests, with  $A$  and  $B$  being the filtered-conditioned acceleration readings of each IMU respectively.

# Chapter 4

## Results

### 4.1 Calibration Results

The results for the calibration procedure are given here. The data set obtained and used is available online. The Root Mean Square Error of the accelerometer readings before calibration is  $0.366233 \text{ ms}^{-2}$  while after calibration it is  $0.002992 \text{ ms}^{-2}$ . Table 4.1 gives the results for the SEM parameters. Where **s** is each  $s_x$ ,  $s_y$  and  $s_z$  parameter respectively, likewise for the **N** parameters and the **b** offsets.

<b>s</b>	$\varphi_{xy,zx,zy}$	<b>b</b>
0.8455	-0.1722	-0.0845
0.7968	0.0227	-0.1915
0.80645	0.2088	-0.1847

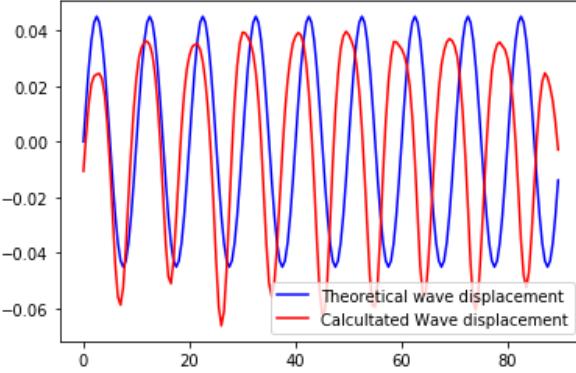
Table 4.1: Table of the SEM parameters calculated

## 4.2 Experimental results

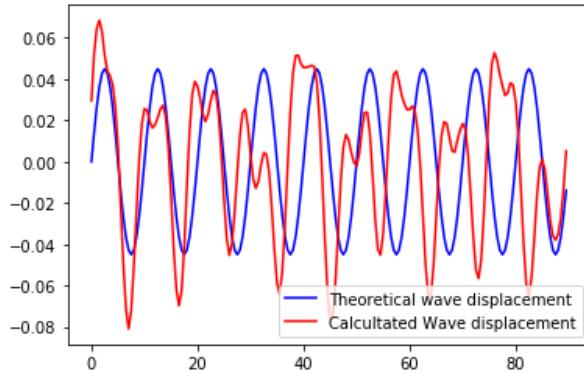
The significant wave heights calculated via each method and for each test are shown in table 4.1. Where  $H_t$  is the significant wave height calculated from the computed displacement time series and  $H_s$  is that from the power spectral density of the displacement. The computed displacement time series for the third test is given in figure 4.1.

	MPU6050		YOST IMU		Theoretical
	$H_t$ (mm)	$H_s$ (mm)	$H_t$ (mm)	$H_s$ (mm)	$H_t$ (mm)
Test 1	21.846109	21.846109	47.598709	47.598709	127.279221
Test 2	57.351404	57.351404	81.159944	81.159943	127.279221
Test 3	142.070419	142.070419	117.401572	117.401572	127.279221

Table 4.2: Table of the significant wave heights calculated for each test and IMU



(a) Graph of the YOST-IMU calculated displacement(m) time(s) series versus theoretical displacement



(b) Graph of the MPU6050 calculated displacement(m) time(s) series versus theoretical displacement

Figure 4.1: Graphs of displacement time series for test 3

## 4.2. EXPERIMENTAL RESULTS

Table 4.3 shows the Pearson correlation coefficient calculated between the MPU6050 and the YOST IMU for their filtered and conditioned acceleration readings (in the  $z$  axis.) Where  $0 \leq \rho \leq 1$ , such that 1 implies a perfect linear relationship between the two and 0 non.

Experiment	$\rho(\eta_{MPU6050}, \eta_{YOST})$
Test 1	0.472023
Test 2	0.555654
Test 3	0.772139

Table 4.3: Table of the Pearson correlation coefficient between IMUs for each test

# Chapter 5

## Discussion

### Calibration

From the calibration results, it is clear that developing a sensor error model for the MPU6050 significantly reduces the error in its reading. With the root mean square error dropping by a factor of over one-hundred. This is a significant improvement. It is not an unrealistic value as in [31] a factor of ten was achieved for a much higher quality inertial measurement unit.

### Experiments

While conducting the experiments it was noted that the Stewart platform suffers from a noticeable amount of noise in the stability of the platform. This is as it uses electric motors instead of hydraulic ones. It also had a slight tilt (around 5°) from horizontal where one of the corners had previously been damaged. Another issue experienced was that the sensor error model was not fully developed at the time of experimentation - with only the offset values being applied - this occurred as the platform was being used by another final year student for a EEE4022 project and thus the platform was only able to be acquired at this point in the development.

The experimental results are interesting to analyse. It is noted that for each successive test, both the IMUs produce significant wave height results that are closer to the true value. Likewise, the Pearson's correlation coefficient increased for each test. It is suspected that this is the case for two reasons.

Firstly, as - in the first test - the frequency (of the height) changing was higher than that of the subsequent tests. While this frequency ( $0.1Hz$ ) and the following ones ( $0.05Hz$ ) lie within expected ones, the faster frequency would increase noise produced by the Stewart platform.

The accuracy increase between the second and third test is greater than that of the first to the second. It is strongly suspected that this is since the first two tests only performed a change of height in  $z$  axis. As mentioned in 2.1.2, the motion of a buoy on a gravity-wave can be modelled as circular. The third test modelled this motion more accurately - which would affect the measured spectrum and hence the conditioning (which is designed for gravity-waves) would yield better results. This is seen as in the third test where the YOST IMU's significant wave height and displacement results were close to that of the theoretical value, discounting any noise produced by the platform. Thus it is reasonable to only consider the third test for comparison purposes.

On the other hand, the MPU6050's significant wave heights calculated in the third test were slightly more off-target than that of the YOST IMU's (an absolute difference of 14.79 versus one of 9.87.) The Pearson's correlation coefficient for the third test was reasonable at a value of 0.77. However, the displacement time series is not as accurate as the YOST-IMU's, with small dips occurring where the top of the wave crest ought to be. Accounting for the noise of the platform and more applying the robust sensor error model it can be expected that the significant wave height calculated would be closer to that of the theoretical model and the Pearson's correlation coefficient higher - but this cannot be expected to be by much.

# **Chapter 6**

## **Conclusions**

While the output of the low-cost MPU6050 is not as accurate as the more expensive inertial measurement unit (YOST-IMU), it can be considered a reasonable approximation. This approximation being better if its errors are accounted for using the sensor error model provided in this project. However, whether this accuracy is good enough for use in scientific deployments is uncertain. The data it provides is not reliable enough to be considered a truly accurate representation of the physical ocean wave process underlying the data. On the other hand, the results from processing the YOST-IMUs data are much closer to that of the underlying phenomena.

# Chapter 7

## Recommendations

Due to the less accurate nature of the MPU6050, it is strongly recommended that if it is to be deployed in the Antarctic then there should be another mechanism by which to calculate the buoys displacement for comparison. This could be done using a pressure sensor which can be related to elevation as described in [21]. Furthermore, any MPU6050 that is to be deployed should be calibrated according to the method described in this project to minimize the error as much as possible. The accuracy of the MPU6050 should also further be improved by implementing a Kalman filter, as well as oversampling at a much higher frequency (640Hz is recommended by [20]) and decimating down - the second's functionality is included in the software. Further mechanisms could include deploying the buoys in a relatively tight cluster and taking the average of the readings, this approach would have to first be studied. If all of these cannot be implemented a more accurate inertial measurement unit should be used, such as the VN-100 which has been extensively tested and deployed in the field [7].

A limitation of this report was that no cold lab testing was done - the Stewart platform not being designed to operate in cold temperatures. By building a new wave simulation device (such as the Ferris wheel device in [14]) that can operate in cold lab conditions the cold environment effects on the IMU can be deduced. Such a device would have minimal mechanical vibration in comparison to the Stewart platform.

A sensor error model of the MPU6050's gyroscope could be developed to use it to correct for platform tilt. This is not necessarily strongly advised as extensive testing would have to be done as in [6] from what tilt angles it is necessary to correct for. The advantage of not using the gyroscope is that by disabling it the MPU6050 uses less power. Thus, it would have to be determined whether using the gyroscope to compensate for platform tilt would increase the accuracy of acceleration readings and whether this increase is worth

the extra power consumption.

Lastly, during the project, it was noted that the signal processing required on the acceleration time series is extensive, this would increase further if oversampling and a Kalman filter were to be implemented. It is suggested that the approach used in [30], whereby signal processing is offloaded to another microprocessor - which enters sleep mode when not needed. The same spectral parameters used in the literature ([30],[6]) should be sent via satellite, it is not feasible to send the acceleration time series.

# Bibliography

- [1] National Aeronautics and Space Administration, "Antarctic Sea Ice Reaches New Record Maximum", *National Aeronautics and Space Administration*. [Online]. Available: [urlhttps://www.nasa.gov/content/goddard/antarctic-sea-ice-reaches-new-record-maximum](https://www.nasa.gov/content/goddard/antarctic-sea-ice-reaches-new-record-maximum). [Accessed: Oct. 13, 2019].
- [2] A. Kohout, M. Williams, S. Dean and M. Meylan, "Storm-induced sea-ice breakup and the implications for ice extent", *Nature*, vol. 509, no. 7502, pp. 604-607, 2014. doi: 10.1038/nature13262
- [3] P. Sutherland and J. Gascard, "Airborne remote sensing of ocean wave directional wavenumber spectra in the marginal ice zone", *Geophysical Research Letters*, vol. 43, no. 10, pp. 5151-5159, 2016. doi: 10.1002/2016gl067713.
- [4] "VN-100 Rugged", Vectornav.com. [Online]. Available: <https://www.vectornav.com/purchase/product/vn-100-rugged>. [Accessed: 03-Oct-2019].
- [5] K. Christensen and G. Brostrom, "Waves in sea ice," Norwegian Meteorological Institute, Norway, Technical report, 2008.
- [6] A. Kohout, B. Penrose, S. Penrose and M. Williams, "A device for measuring wave-induced motion of ice floes in the Antarctic marginal ice zone", *Annals of Glaciology*, vol. 56, no. 69, pp. 415-424, 2015. doi: 10.3189/2015aog69a600.
- [7] J. Rabault, G. Sutherland, B. Ward, K. Christensen, T. Halsne and A. Jensen, "Measurements of Waves in Landfast Ice Using Inertial Motion Units", *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 11, pp. 6399-6408, 2016. doi: 10.1109/tgrs.2016.2584182.
- [8] U.S. Department of Defense, "GLOBAL POSITIONING SYSTEM STANDARD POSITIONING SERVICE PERFORMANCE STANDARD". Washington, D.C: Defense Pentagon, 2008. [Online]. Available: <https://www.gps.gov/technical/ps/2008-SPS-performance-standard.pdf>. [Accessed: 03-Oct-2019].

## BIBLIOGRAPHY

- [9] W. H. Munk, “Origin and Generation of Waves,” Coastal Engineering Proceedings, vol. 1, 1950.
- [10] R. Mitterer, course materials for GEOSCIENCES 3401, OCEANOGRAPHY, Spring 2006. Online: <https://personal.utdallas.edu/~mitterer/Oceanography/pdfs/OCEChapt09.pdf>, The University of Texas at Dallas. Downloaded on [09 October 2019].
- [11] L. H. Holthuijsen, “Waves in Oceanic and Coastal Waters,” p. 405, 2007.
- [12] A. Pecher, J. Kofoed and M. Folley, Handbook of ocean wave energy. Springer Open, 2017, pp. 43-60.
- [13] O. Phillips, “The dynamics of the upper ocean”, 2nd ed. Cambridge: Cambridge Univ. Press, 1980.
- [14] J. Kuperus, “Wave Monitoring using Wireless Sensor Nodes”, BSc(Hons) thesis, University of Melbourne, Melbourne, Victoria, 2009
- [15] Pierson, W., Moskowitz, L.: A proposed spectral form for fully developed wind seas based on the similarity theory of S.A. Kitaigorodskii. *J. Geophys. Res.* 69, 5181–5203 (1964)
- [16] Hasselmann, K., et al.: Measurement of wind-wave growth and swell decay during the joint North Sea wave project (JONSWAP). *Dtsch. Hydrogr. (1973)*
- [17] M.D. Earle, “Nondirectional and directional wave data analysis procedures”, National Data Buoy Center, National Oceanic and Atmospheric Administration, US Department of Commerce, Washington, DC, NDBC Technical Document 96-01, Jan 1996
- [18] National Buoy Data Center, “How are significant wave height, dominant period, average period, and wave steepness calculated?”, *National Oceanic and Atmospheric Administration*, June 2018. [Online]. Available: <https://www.ndbc.noaa.gov/wavecalc.shtml> [Accessed: 08-Oct-2019].
- [19] Prof. A.H. Techet, course material “13.42 Design Principles for Ocean Vehicles”, Spring 2005. MIT OpenCourseWare: <https://web.mit.edu/13.42/www/handouts/reading-wavespectra.pdf>, Massachusetts Institute of Technology. Downloaded on [08 October 2019].
- [20] M. Tucker and E. Pitt, “Waves in ocean engineering”. Amsterdam: Elsevier, 2001.

- [21] A. Marchenko, E. Morozov and S. Muzylov, "Measurements of sea-ice flexural stiffness by pressure characteristics of flexural-gravity waves", *Annals of Glaciology*, vol. 54, no. 64, pp. 51-60, 2013. doi: 10.3189/2013aog64a075.
- [22] G. Blewitt, "Basics of the GPS Technique: Observation Equations", 1999. [Online]. Available: <http://www.nbmng.unr.edu/staff/pdfs/Blewitt%20Basics%20of%20gps.pdf>. [Accessed: 03-Oct-2019].
- [23] GISGeography, Intersection of GPS distance spheres. 2016.[Online]. Available: <https://gisgeography.com/wp-content/uploads/2016/10/GPS-Trilateration.png>. [Accessed: 03-Oct-2019].
- [24] J. Voldman, course materials for 6.777J / 2.372J "Design and Fabrication of Microelectromechanical Devices", Spring 2007. MIT OpenCourseWare: <http://ocw.mit.edu/>, Massachusetts Institute of Technology. Downloaded on [03 October 2019].
- [25] S. Senturia, Microsystem design, 1st ed. Boston: Kluwer Academic, 2002, pp. 3-7.
- [26] S. Sinha, S. Shakya, R. Mukhiya, R. Gopal, and B. Pant. "Design and simulation of mems differential capacitive accelerometer". 2014. doi: 10.13140/2.1.1074.8809
- [27] I. Frosio, F. Pedersini and N. Borghese, "Autocalibration of MEMS Accelerometers", *IEEE Transactions on Instrumentation and Measurement*, vol. 58, no. 6, pp. 2034-2041, 2008. doi: 10.1109/tim.2008.2006137.
- [28] D. Tedaldi, A. Pretto and E. Menegatti, "A robust and easy to implement method for IMU calibration without external equipments", 2014 IEEE International Conference on Robotics and Automation (ICRA), 2014. doi: 10.1109/icra.2014.6907297.
- [29] J. Lv, A. Ravankar, Y. Kobayashi and T. Emara, "A method of low-cost IMU calibration and alignment", 2016 IEEE/SICE International Symposium on System Integration (SII), 2016. doi: 10.1109/sii.2016.7844027.
- [30] J. Rabault, O. Gundersen, G. Sutherland, A. Jensen, A. Marchenko and Ø. Breivik, "AN OPEN SOURCE, VERSATILE, AFFORDABLE WAVES IN ICE INSTRUMENT FOR SCIENTIFIC MEASUREMENTS IN THE POLAR REGIONS", 2019. Available: <https://arxiv.org/abs/1901.02410v4>.
- [31] J. Rohac, M. Sipos and J. Simanek, "Calibration of low-cost triaxial inertial sensors", *IEEE Instrumentation & Measurement Magazine*, vol. 18, no. 6, pp. 32-38, 2015. doi: 10.1109/mim.2015.7335836.
- [32] A. Archer, "Investigation of Three-Axis Accelerometer Calibration Techniques", Honors Thesis for BSE (Bachelor of Science in Engineering), University of Iowa,

2017. Available: [https://ir.uiowa.edu/honors\\_theses/29](https://ir.uiowa.edu/honors_theses/29).
- [33] InvenSense, MPU-6000 and MPU-6050 Product Specification, PS-MPU-6000A-00 datasheet, 08/19/2013
- [34] “Accelerometer Specifications - Quick Definitions”, Analog Devices. [Online]. Available: <https://www.analog.com/en/products/landing-pages/001/accelerometer-specifications-definitions.html>. [Accessed: 04- Oct- 2019].
- [35] Yost Labs, Yost Labs 3-Space SensorTM Watertight USB/RS232, TSS-USBWT-S, Technical Brief, 2017
- [36] L. Clark, “The Stewart Platform - A Proposed Ocean Surface Simulation Platform”, BSc(Hons) thesis, University of Cape Town, Cape Town, Western Cape, 2018
- [37] Silicon Labs, “*IMPROVING ADC RESOLUTION BY OVERSAMPLING AND AVERAGING*”, Silicon Laboratories, AN118, July 2013. [Online]. Available: <https://www.silabs.com/documents/public/application-notes/an118.pdf>. [Accessed: 14 October 2019]
- [38] A. Brandt and R. Brincker, “Integrating time signals in frequency domain – Comparison with time domain integration”, Measurement, vol. 58, pp. 511-519, 2014. doi: 10.1016/j.measurement.2014.09.004.
- [39] V. Squire, “A fresh look at how ocean waves and sea ice interact”, Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, vol. 376, no. 2129, p. 20170342, 2018. doi: 10.1098/rsta.2017.0342.
- [40] L. Bender, N. Guinasso, J. Walpert and S. Howden, “A Comparison of Methods for Determining Significant Wave Heights—Applied to a 3-m Discus Buoy during Hurricane Katrina”, Journal of Atmospheric and Oceanic Technology, vol. 27, no. 6, pp. 1012-1028, 2010. doi: 10.1175/2010jtecho724.1.
- [41] M. Pedley, “Tilt Sensing Using a Three-Axis Accelerometer”, Freescale Semiconductor, Technical Report. AN3461, March 2013.
- [42] H. Sverdrup and W. Munk, “Wind sea and swell: Theory of relations for forecasting, Publ. 601, US Navy Hydrogr,” Off., Washington, DC, 1947.

# Appendix A

## Realised implementations of algorithms

### A.1 Despiking implementation

The python 3 implementation of algorithm 1 is given here:

```
1 def despike(arr):
2     """
3         Remove spikes in data that is above/below three standard deviations from the
4             mean
5         @param: arr array to remove spikes from
6         @returns: arr = array with removed spikes and (linear) interpolated values
7     """
8     for i in range(0,3):
9         spikes = 0
10        mean = sum(arr)/len(arr)
11
12        #Calculate the Standard deviation
13        s = 0
14        for j in range(0, len(arr)):
15            s += (arr[i] - mean)**2
16
17        stdDiv = (s/len(arr))**0.5
18        i = 0
19        spikeIndexArr = []
20        for reading in arr:
21            if (reading > (mean + stdDiv*3) or reading < (mean - 3*stdDiv)):
22                spikes += 1
23                spikeIndexArr.append(i)
24                i += 1
25            #Interpolate
26            for i in spikeIndexArr:
27                arr[i] = (arr[i-1] + arr[i+1])/2
28
29    return arr
```

## A.2 Detrending implementation

The python 3 implementation of algorithm 2 is given here:

```

1 def detrend(arr, fs):
2     """
3         Remove trends in time series. Equivalant to a high pass filter
4         with time constant =  $(1/fs)/(1 - k)$ 
5         @param: arr = array to remove trends from
6         @param: fs = sampling frequency
7         @returns: detArr = detrended array
8     """
9     mean = sum(arr)/len(arr)
10    detArr = []
11    sn_prev = 0
12    k = 1 - (1/fs)/100
13    for yn in arr:
14        sn = (yn-mean) + k*sn_prev
15        detArr.append((yn-mean) - (1-k)*sn)
16        sn_prev = sn
17    return detArr

```

## A.3 Decimation implementation

The python 3 implementation of the unused decimation scheme is given here:

```

1 import scipy.signal as sig
2
3 def analogLPF(arr, fn, fc = 8, n=1):
4     """
5         Apply an nth order analog low pass filter on an array
6         @param: arr = signal to filter
7         @param: fn = nyquist frequency
8         @param: fc = cut-off frequency
9         @param: n = order of filter
10        @returns: filtered signal
11    """
12    w = fc/fn
13    b, a = sig.butter(n, w, 'low', True)
14    return sig.lfilter(b,a, arr)
15
16 def decimate(arrAz):
17     """
18         Decimate the signal to 2Hz
19         @param: arrAz - the filtered signal sampled at 640Hz to decimate
20         @returns: decimated array
21     """
22     #Create a no gain discrete transfer function
23     dti = sig.TransferFunction(1, 1, dt=(1/640))
24
25     #Apply 1Hz analog LPF and decimate to 80Hz
26     arrAz = analogLPF(arrAz, 640/2, 1, 2)
27     s80 = sig.decimate(arrAz, 80, ftype=dti)
28
29     #Apply 0.5Hz analog LPF and decimate to 2Hz
30     s80 = analogLPF(s80, 80/2, 0.5, 2)
31     s2 = sig.decimate(s80, 4, ftype=dti)
32     return s2

```

## A.4 Spectral moment calculation

The python 3 implementation of the spectral moment calculation is given here:

```

1 import scipy.signal as sig
2
3 def spectralMoment(heights, fs, n):
4     """
5         Calculate the nth moment from an array of displacement values.
6         @param: heights - the array of displacement values
7         @param: fs - the sampling frequency of heights
8         @param: n - the order of the moment to return
9         @returns: m_n - the nth spectral moment
10    """
11    #Estimate the PSD
12    f, pxx = sig.periodogram(heights, fs)
13    df = fs/len(heights)
14    s = 0
15    for i in range(1, 1025):
16        s+= (f[i]**n)*pxx[i]
17    m_n = df*s
18    return m_n

```

## A.5 Other wave characterization parameters calculation

The python 3 implementation of the calculations of other wave characterization parameters is given here:

```

1 def waveEnergyPeriod(m_neg_one, m_0):
2     """
3         Calculate the wave energy period of a wave (Te)
4         @param: m_neg_one - the first negative spectral moment
5         @param: m_0 - the zeroth spectral moment
6         @returns: The wave energy period
7     """
8     return m_neg_one/m_0
9
10 def spectralBandwidth(m_0, m_neg_one, m_neg_two):
11     """
12         Calculate the spectral bandwidth epsilon_0
13         @param: m_0 - the zeroth spectral moment
14         @param: m_neg_one - the first negative spectral moment
15         @param: m_neg_two - the second negative spectral moment
16         @returns: The spectral bandwidth
17     """
18     return ((m_0 * m_neg_two)/(m_neg_one**2) - 1)**0.5
19
20 def meanZeroCrossingPeriod(m_0, m_2):
21     """
22         Calculate the estimate of the mean zero-crossing period of the waves
23         @param: m_0 - the zeroth spectral moment
24         @param: m_2 - the second spectral moment
25         @returns: The estimate of the mean zero-crossing period
26     """
27     return (m_0/m_2)**0.5

```