Bachelor thesis TU Delft

Accuracy Comparison of Inertial Measurement Units (IMUs)

Ties Schwertasek[*]
Supervisors: Stephanie Tan, Hayley Hung[‡]
EEMCS, Delft University of Technology, The Netherlands

January 23, 2022

Abstract

The Midge is a sensor device developed by the Socially Perceptive Computing Lab (SPCL) at Delft University of Technology (TU Delft). This device is used to monitor human behaviour in social settings using several sensors. In this paper, the accuracy of the Inertial Measurement Unit (IMU) chip used in the Midge sensor package was evaluated. The IMU is responsible for sensing motion in multiple directions using an accelerometer, gyroscope and magnetometer. This experiment was performed by comparing several Midge devices with each other as well as comparing them with a modern smartphone. First will be explained how the control software was updated to run on modern hardware (computers) and the work that was done to reliably convert the generated binary data to readable data (parsing). Then the process of creating the test setup, performing the tests and analyzing the data will be explained.

Contents

---

[*] `t.g.schwertasek@student.tudelft.nl`
[‡] `s.tan-1@tudelft.nl, h.hung@tudelft.nl`

1 Introduction

A lot of devices nowadays contain Inertial Measurement Units (IMUs), these sensors allow measurement of movement of a device. They achieve this by using an accelerometer and a gyroscope to measure acceleration and rotation. Some IMUs also include a magnetometer to generate even more data [1]. When a device containing an IMU is carried by a person, it can therefore also measure their motion. This data is used for all kinds of purposes, from lifestyle gadgets tracking activity and sleep to safety applications such as fall or crash detection [2].

The reason for this paper, however, is the use of IMUs in the Midge. The Midge is a wearable sensor developed by the Socially Perceptive Computing Lab (SPCL) at Delft University of Technology (TU Delft). These devices are used for researching human interaction. The Midges provides them with even more data than only using existing techniques, like videotaping crowds. The Midge also has the advantage that it is with it's subject continuously, while it is possible for people to end up in a blind spot when only cameras are used.

To asses usability of data provided by an IMU for a specific use case, it is important to ascertain how accurate that data is. While small inaccuracies might not affect the recognition of repeating patterns like running or other sports, it might heavily affect detection of certain subtle one-off motions like the turning of a body towards a new conversation partner. When the sensor data gets to unreliable it might even become impossible to make this distinction. This paper will not go into the details of determining the actual meaning of the motions, instead just focusing on the sensors itself.

There is some previous research into the general topic of IMU accuracy, even specifically related to human motion tracking, for example [3]. The purpose of this paper however is to analyze the accuracy of the IMU found on the Midge. This will be done both by determining consistency between Midge devices and by comparing the Midge to an IMU found in a modern smartphone.

Previous research, including [3], focused on the more general topic of IMU accuracy, using sophisticated experiment setups, even including high-end robotic arms. This research focuses specifically on one type of IMU chip and will use controlled and repeatable experiments that are possible within the time given for this research and should be sufficient to analyze how this chip compares with the smartphone IMU.

This paper aims to answer the question **"How does the Socially Perceptive Computing Lab's Midge IMU compare to a widely used IMU (iPhone 13 Pro Max)?"**. It will do that by answering the following sub questions:

- How do the IMUs compare in measurement frequency?
- How do the IMUs compare in measurement accuracy?
- Is the accuracy of the Socially Perceptive Computing Lab's Midge sufficiently accurate for it's purpose?

2 Methodology

In order to retrieve data from the Midge devices, two steps have to be taken. The data collection processes on all Midge devices is controlled centrally by the hub. The data, stored on micro SD-cards, then has be offloaded onto a computer and parsed into a usable format by the parser. The working of these components is described in more detail below. Since other research also relied on this data gathering process to work, fixing the errors in this process was done together.

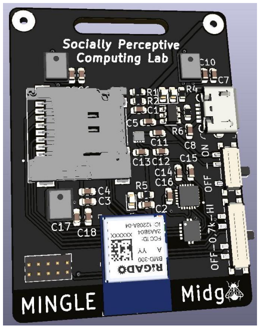

Figure 1: 3D render of Midge PCB [4]

2.1 The hub

The Midge sensor devices are controlled centrally by a piece of code called the hub. This hub allows controlling multiple Midge devices at once. The main task of this code is to send the current timestamp to all of the connected Midge devices and starting them all at once. The hub is also responsible for stopping the data collection process on all connected devices when the experiment is done.

The version of the hub code that was provided was written for Python 2, a programming language that was deprecated on January 1, 2020 [5]. Since this was already some time ago, it was not possible to get a fully compatible code environment set up. Some dependencies were not available in the specified versions anymore or did not run on the available hardware platforms.

To remedy this problem, we set out to update the hub code to run on Python 3. This was done by updating outdated code and upgrading outdated dependencies to supported versions. After some trial and error, we managed to successfully resolve incompatibilities between those Python versions.

Unfortunately that was not all that had to be done for the code to work. While it was running now on our machines and we were able to start data collection, it still contained a lot of bugs. Stopping data collection was not working properly and trying to access the interactive terminal returned several errors.

It became clear that a lot of work had been done by other members of the SPCL in a different code repository. This code fixed a lot of the bugs we had encountered before but was still written for Python 2. This was fortunately easily fixed by applying the same updates that had been made to the first repository. This eventually left us with a Python 3 compatible hub code base with all bugs resolved. All of this work on the code was unexpected and since we did not know about the second code repository at first. This troubleshooting unfortunately caused significant delays before performing our own experiments was possible.

2.2 The Parser

Data from the Midge devices is stored in a binary format on micro SD-card. This binary format consists of data frames for each measurement, including a time stamp and the raw data.

After getting some initial sample data from the devices, we noticed some inconsistent parsing behavior between different Midge devices. The time stamps for the data frames where very far from reasonable, causing the parser to throw errors.

It then came to light that some of the devices had received a firmware update, shortening the data frames from 32 bytes to 24 bytes by removing some useless padding. Since the parser expected 24 bytes, it did not work for our 32 byte data frames.

There was no record of which devices had which firmware and making sure all devices were updated to the same firmware was not feasible with the time left. Therefore data from each device was tested with different parsers, expecting different data frame lengths, to determine which firmware was installed on which device. This allowed the use of only the updated devices for the actual experiments.

2.3 Phone data

For data gathering on the phone an app called PhyPhox was used. This app allowed grouping sensors and recording them all at once. For this experiment the same sensors that were available in the Midge's IMU were selected: accelerometer, gyroscope and magnetometer. The app allowed to export the recorded data as CSV files so no parsing was required for this data.

3 Experiment

3.1 Requirements

In order to come up with a good experiment, requirements were formulated. The experiment had to:

- be repeatable
- generate data on as many sensors and axes as possible
- introduce as little noise and unexpected motion as possible

3.2 The setup

Initially, an experiment using a driving cart was designed. This cart would drive a fixed distance and would be propelled by a falling weight connected to it by a string. Using the force of gravity would ensure the same acceleration every time. This experiment would however only generate data on one axis and in one direction.

To improve this, an alternative experiment was proposed that would generate more data on more axes. This experiment uses a pendulum on which the devices would swing. This method generates movement in multiple directions and even some rotation of the devices. The added benefit of this method is that this motion can be related to human motion, since a human can be modelled as an inverted pendulum [6]. Inverted because feet on the ground are fixed and the body swings above, somewhat like a pendulum.

In testing it quickly became clear that this would not be sufficient. A string introduces too much unwanted additional motion in the form of side to side movement and slight bouncing, caused by the tension in the string. This would result in noise in the data and also making the experiment completely unrepeatable.

Using a solid rod instead of the string makes it a lot more rigid but posed some issues with mounting it to anything, as this was not as simple as using some tape.

The pendulum was constructed using 20mm aluminium extrusions to provide a very solid base tower. For the swinging rod the same extrusions were used, connected to the base tower by a ball bearing to ensure smooth swinging. All these components were mounted to each other using custom made 3D-printed brackets.

3.3 Running the experiment

To ensure there was no variance in movement between the devices, it was decided to run the experiment for all devices at once. This was done by fixing all the devices together to the pendulum.

Close attention was also paid to the alignment of the IMU chips in the different devices. This was done by using an x-ray picture of the inside components of the phone

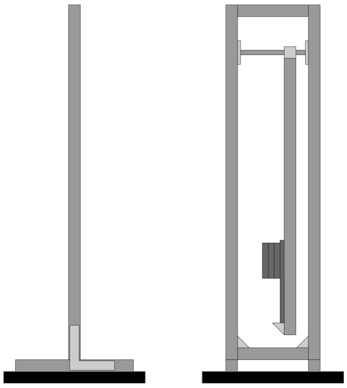

Figure 2: Diagram of the pendulum setup

Figure 3: Picture of the pendulum setup

Figure 4: The phone with the Midges fixed on to it

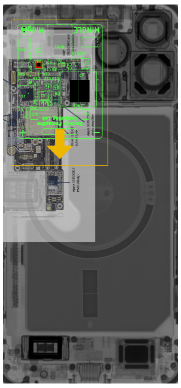

Figure 5: Positioning guide for mounting Midges on the phone

from [7] that could then be set as a background of the phone with a 1:1 scale. Overlaid on this picture was a picture of the phone's motherboard on which the IMU chip was then highlighted. The chip used in this series of iPhones was identified using iFixit's iPhone 13 Pro Teardown Guide [8] and Chip ID [9]. Another teardown [10] was used to find the chip position on the specific model that was used. On top of that, a diagram of the Midge's Printed Circuit Board (PCB) was placed with it's IMU chip lined up with the IMU of the phone. Lastly an outline of the Midge outer casing was drawn around the PCB diagram using manual measurements. Using this image, the Midges were positioned on exactly the right place on the phone. The position of the IMU chip on both devices only allowed the Midges to be mounted to the phone upside down, otherwise they would interfere with the phone case. One welcome side effect of that approach is that this lines up the data axes between the phone and Midges, since the IMU chip is mounted upside down on it's PCB.

The actual experiment consisted of manually moving the pendulum about 90° to either side, letting it go, and waiting until it stopped swinging completely. The data gathering was started slightly before moving the pendulum. It was then stopped some seconds after the pendulum stopped moving. This allows the stationary period before and after the movement to be clearly visible in the graphs. This experiment was conducted three times, using all devices, a phone and three Midges, each time. The experiments were named A, B, and C. Experiment A does not include phone data since the recording app on the phone crashed during the experiment. This data was included since it could still be relevant for comparison between the Midge devices. Three runs were considered enough, even with one experiment missing data, since it already showed a pattern of data from the devices.

3.4 Data processing

In order to successfully graph and compare all data, some modifications had to made. The units of the data from some of the sensors differed between Midge and phone. The phone's gyroscope data was changed from rad/sec to $°/sec$ to match the Midge devices. The accelerometer data from the Midge was converted from multiples of g (the value of gravity) to absolute m/s^2 values like used on the phone. This was done by multiplying the values in g by $g_n = 9.80665$, the standard acceleration of gravity on earth [11]. In both cases the conversion could also have been done the other way around, but that does not matter for the analysis.

With the units now aligned, patterns start to emerge when plotting the data. However, the data was not yet synchronized. This was foreseen between data from phone and Midge since they are manually started separately. It was a surprise that the data was also out of sync between Midges, since timestamp synchronization was part of the hub code's process. This was fixed by closely looking at the graphs and manually offsetting timestamps for each device. This was only possible because all devices encountered the exact same motion so extremities in the graphs could easily be used to align the data.

4 Results and Discussion

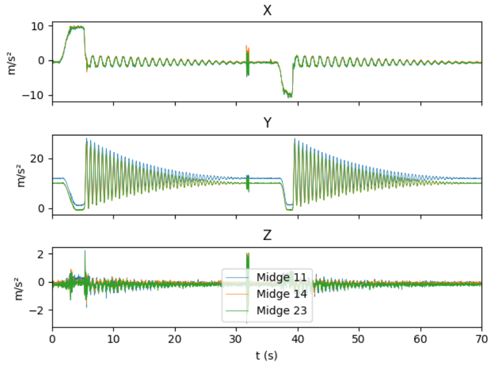

Figure 6: Experiment A accelerometer data

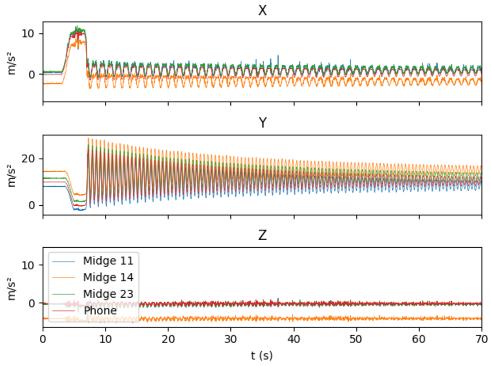

Figure 7: Experiment C accelerometer data

The complete data graphs for all sensors across all experiments can be found in Appendix A. Figure 6 and Figure 7 show some interesting examples of the observed sensor behaviour.

The first thing to note is that all sensors pick up the same motion patterns. Unfortunately, as seen in the graphs, the data values do have a significant offset between sensors. The difference is up to around 80% between all sensors. The Midges also differ up to around 50% from expected absolute values, where they are available. This difference from absolute values is not observed in the phone data.

The variances are even bigger when looking at the magnetometer data from the Midges in Figure 8 and Figure 9. This can be explained by the fact that these sensors

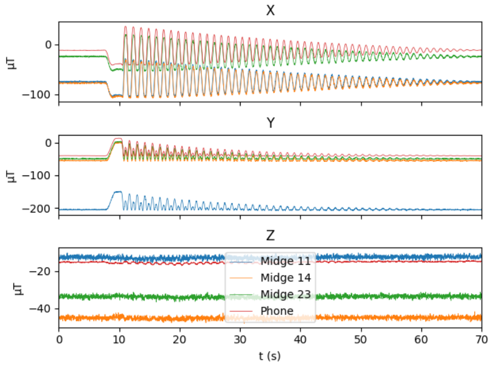

Figure 8: Experiment B magnetometer data

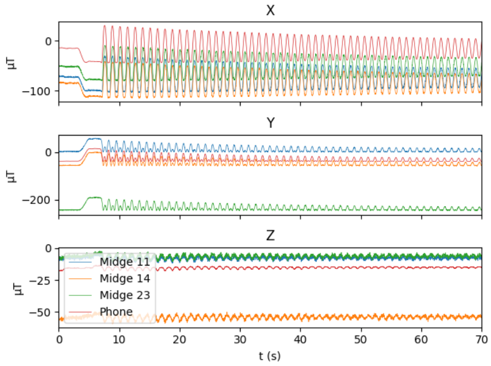

Figure 9: Experiment C magnetometer data

require calibration to work properly [12]. Since this is apparently not done on the Midge itself and there is no record of that happening with the Midges for other experiments, it is beyond the scope of this research to evaluate other values than the ones obtained from the Midge as is.

Another thing to note is that the Midge results show more noisy data than the phone signal. This is even observed when everything is stationary, before the experiment is started. This needs to be taken into consideration when analyzing this data.

Overall, the phone data seems the most reliable and precise, with the Midge devices providing unreliable data. Since the Midge devices do correctly pick up motion patterns the same as the phone, they might be usable for pattern recognition. The value offset makes the absolute values reported by these devices unreliable.

5 Conclusions and Future Work

While the data from the Midges can not be called useless, it can certainly not be called very reliable. When compared to the phone IMU, it has a lower sample rate. The Midge also has more noisy data and less accuracy. It is therefor clear that the phone IMU is the much better one. When taking into account the observed inaccuracies, it should however still be possible to recognize movement patterns in the data, still providing some useful information for the SPCL. They are however not accurate enough to blindly use the absolute values they report.

More experiments can be done on the Midges to test motion on other axes or at other speeds. It might also be interesting to try an experiment where more exact expected values are known. This would allow absolute comparison to exact values instead of just mostly relative values between different devices.

6 Responsible Research

While the research done for this particular paper is purely technical and doesn't directly affect any people, the use case of these sensor boards is to analyze human motions and interactions. Motion analysis is already proving very useful for things like sports tracking and even safety applications like fall or crash detection. This has even saved lives [13].

The area of interaction tracking may pose some ethical questions. While currently someone must wear a sensor badge voluntarily, it's not entirely unthinkable that the same kind of analysis can be performed by a smartphone, possibly without user consent.

This would allow even more detailed interaction registration, even in more crowded settings.

References

[1] N. Ahmad, R. A. R. Ghazilla, and N. M. Khairi, "Reviews on various inertial measurement unit (imu) sensor applications," *International Journal of Signal Processing Systems*, vol. 1, 2013. [Online]. Available: http://www.ijsps.com/uploadfile/2013/1128/20131128022014877.pdf.

[2] G. Shin, M. H. Jarrahi, A. Karami, N. Gafinowitz, A. Byun, and X. Lu, "Wearable activity trackers, accuracy, adoption, acceptance and health impact: A systematic literature review," Sep. 2018. DOI: 10.13140/RG.2.2.22188.10888.

[3] L. Ricci, F. Taffoni, and D. Formica, "On the orientation error of imu: Investigating static and dynamic accuracy targeting human motion," *PLoS ONE*, vol. 11(9), 2016. DOI: https://doi.org/10.1371/journal.pone.0161940.

[4] SPCL. "Midge github repository." (), [Online]. Available: https://github.com/TUDelft-SPC-Lab/midge-code.

[5] "Sunsetting python 2." (), [Online]. Available: https://www.python.org/doc/sunset-python-2/.

[6] D. A. Winter, A. E. Patla, F. Prince, M. Ishac, and K. Gielo-Perczak, "Stiffness control of balance in quiet standing," *Journal of Neurophysiology*, vol. 80, no. 3, pp. 1211–1221, 1998, PMID: 9744933. DOI: 10.1152/jn.1998.80.3.1211. eprint: https://doi.org/10.1152/jn.1998.80.3.1211. [Online]. Available: https://doi.org/10.1152/jn.1998.80.3.1211.

[7] S. Goldheart. "Iphone 13 pro and pro max teardown wallpapers." (), [Online]. Available: https://www.ifixit.com/News/52903/iphone-13-pro-and-pro-max-teardown-wallpapers.

[8] "Iphone 13 pro and pro max teardown wallpapers." (), [Online]. Available: https://www.ifixit.com/Teardown/iPhone+13+Pro+Teardown/144928.

[9] "Iphone 13 pro full chip id." (), [Online]. Available: https://www.ifixit.com/Guide/iPhone+13+Pro+Full+Chip+ID/144993.

[10] H. K. Vishwakarma and A. Razdan. "Apple iphone 13 pro max teardown report." (), [Online]. Available: https://unitedlex.com/insights/apple-iphone-13-pro-max-teardown-report.

[11] "Codata value: Standard acceleration of gravity." (2018), [Online]. Available: https://physics.nist.gov/cgi-bin/cuu/Value?gn (visited on 01/23/2022).

[12] M. . Kok and T. B. Schon, "Magnetometer calibration using inertial sensors," *IEEE Sensors Journal*, vol. 16, no. 14, pp. 5679–5689, 2016. DOI: 10.1109/jsen.2016.2569160.

[13] V. . H. . Orellana. "My apple watch saved my life: 5 people share their stories." (Sep. 10, 2020), [Online]. Available: https://www.cnet.com/tech/mobile/apple-watch-lifesaving-health-features-read-5-peoples-stories/ (visited on 01/23/2022).

A Results

A.1 Experiment A

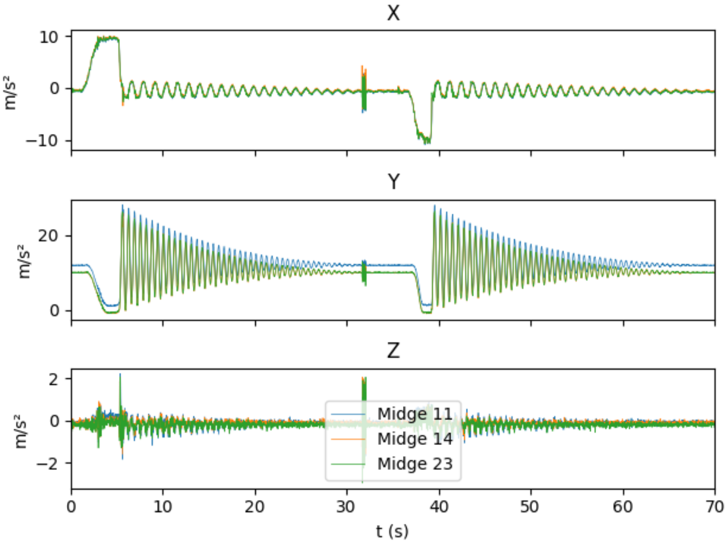

Figure 10: Accelerometer

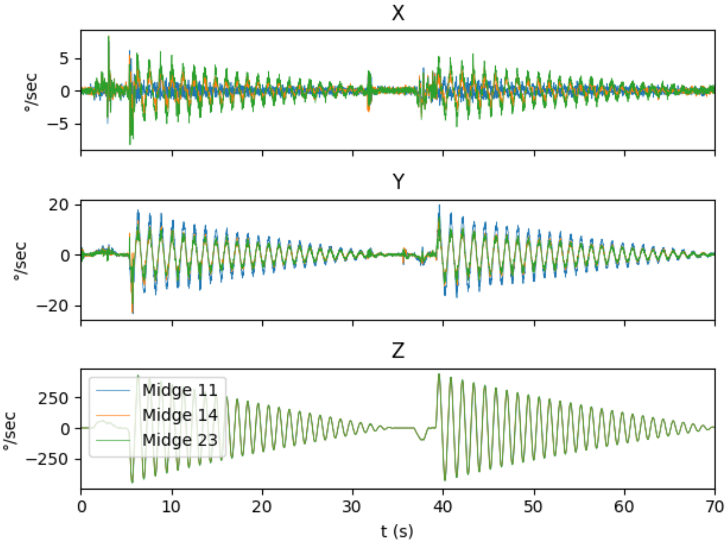

Figure 11: Gyroscope

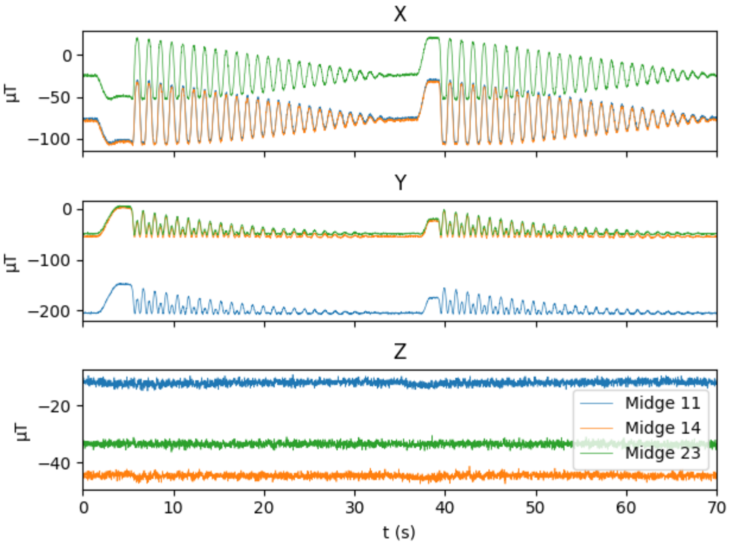

Figure 12: Magnetometer

A.2 Experiment B

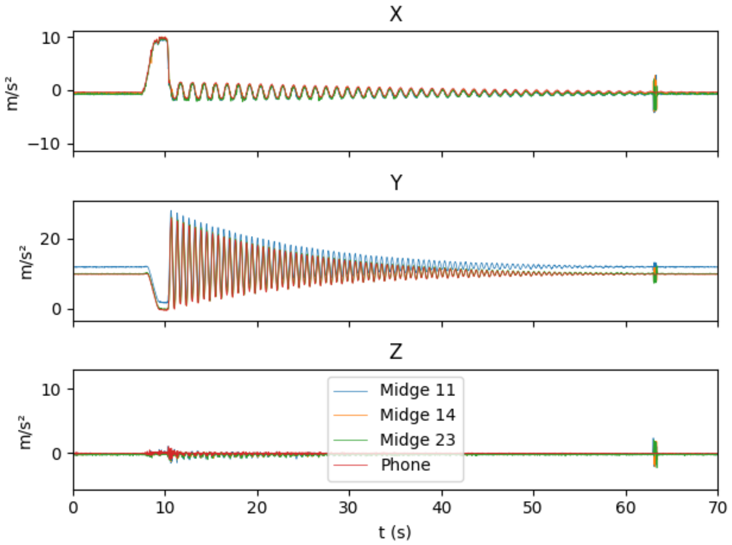

Figure 13: Accelerometer

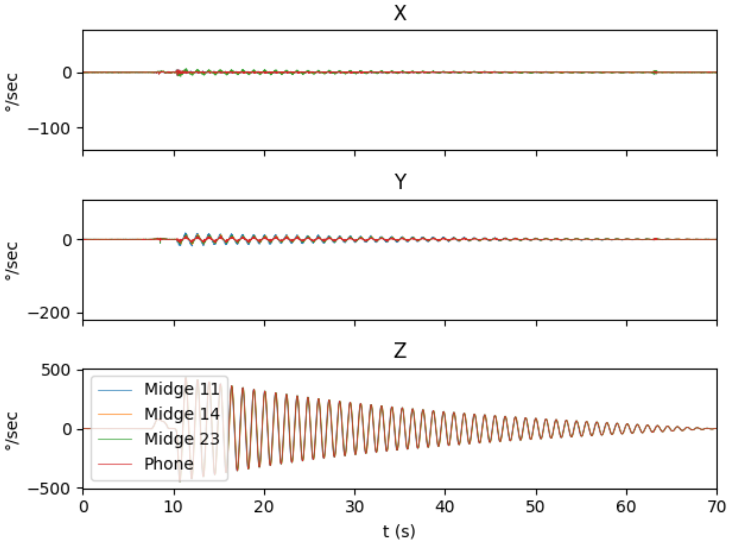

Figure 14: Gyroscope

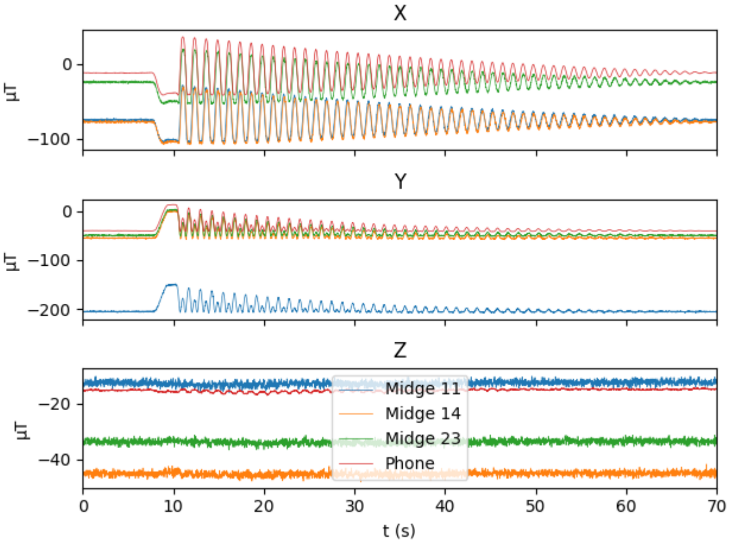

Figure 15: Magnetometer

A.3 Experiment C

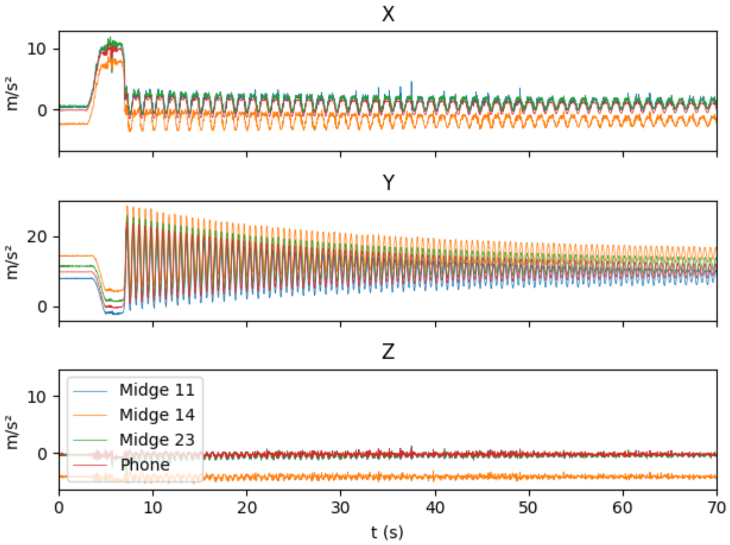

Figure 16: Accelerometer

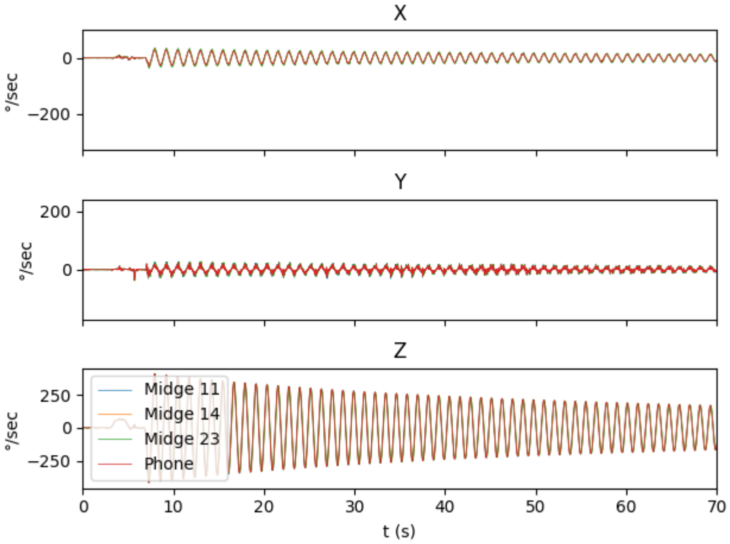

Figure 17: Gyroscope

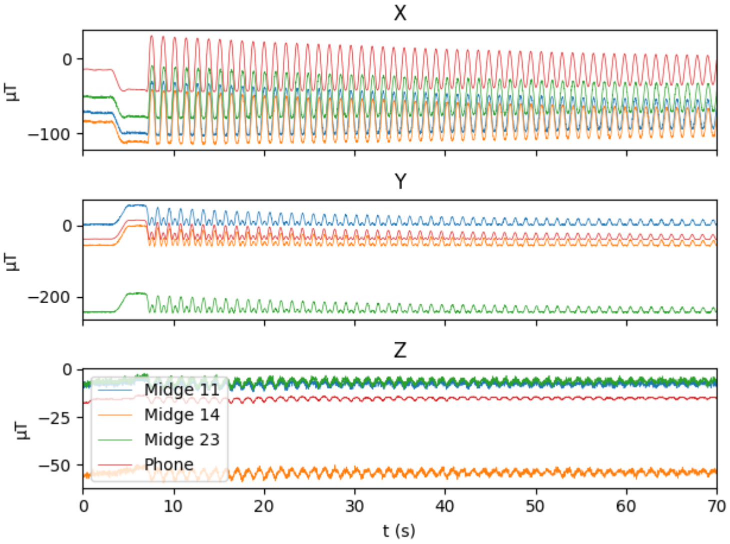

Figure 18: Magnetometer