Independent Study on Droplet Motion, Calibration and Shape formation with Edge following

Praveen Gnanasekaran

May 14, 2017

Abstract

This document explains about the work I have done towards shape formation in droplets, problems faced, how they were overcome and experiments which worked and which did not work. It starts with explanation on range, bearing and heading measurements- how they are used, then explains calibrating the droplets for motion, an introduction to shape formation using the edge following method and future works and ideas to implement shape formation with localization.

1 Introduction

This report explains how to get started with using the API function calls for calibrating the droplets for motion, checking your calibration parameters, the range, bearing and heading values and how to use them and finally how to implement edge following in the droplets as a first step towards shape formation. The droplets by themselves are not calibrated for motion because of the way their motors are aligned and it is important to calibrate them manually to get them to move predictably. Once this is done, the range and bearing values might come in handy to move the droplet relative to another droplet's position in a swarm. The IR collision values can also be used to move the droplets in a certain path in cases where the range and bearing values might not be reliable. Finally this report talks about methods which have been tried but do not work or have given little success for edge following and research done towards this using a combination of IR collision values and RNB for shape formation.

2 Approach

2.1 Range, Bearing and Heading

The first step towards getting an estimate of the droplets' position in a swarm is using the Range and Bearing(RNB) measurements. For this experiment understanding how the RNB is calculated[3] was not necessary but knowing how to use API calls to use the RNB data was sufficient. For the RNB to work, there must at least two droplets involved. One is the transmitter (TX) which broadcasts its data to the receiver (RX) which can read this data and display it. The RNB values received by RX are always relative to the TX droplet.

Range The range between two droplets is the distance measured between them from their centers, as shown in Figure 1.

Figure 1: Range

Bearing The bearing of the receiving (RX) droplet is the measure of the number of degrees of where the TX droplet is located with respect to the RX droplet from the north line joining their centers[1]. Figure 2 shows the bearing angles measured from the top of the droplet. In Figure 3 below, the bearing of the RX relative to the TX in both cases is around -70 degrees.

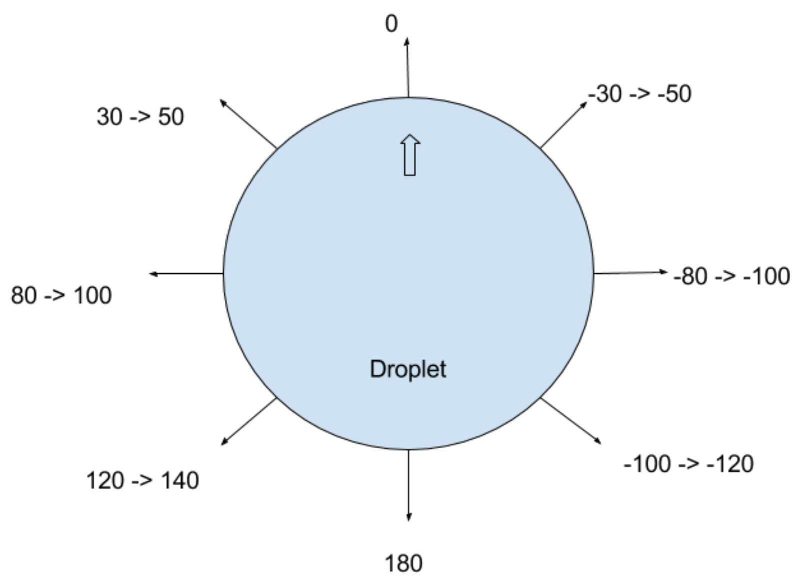

Figure 2: Bearing angles

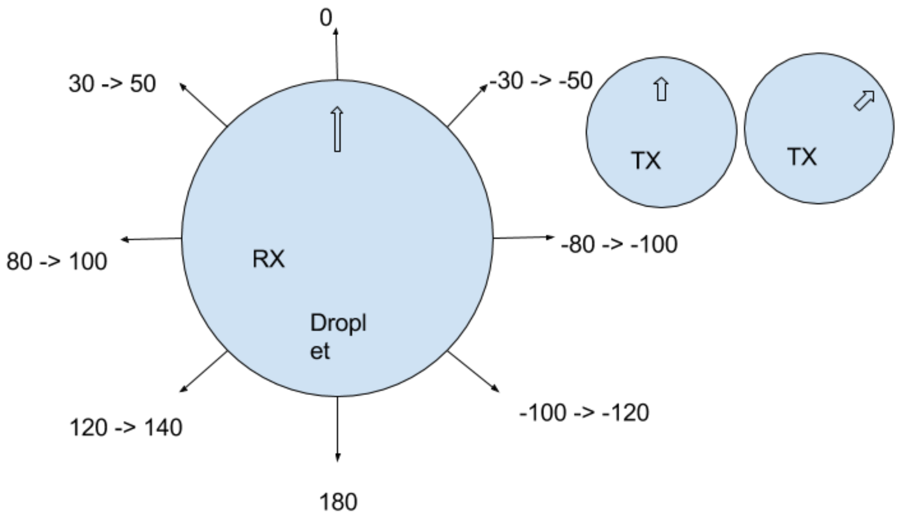

Figure 3: Bearing example

Heading The heading values between two droplets depend on the way the droplets' are oriented with respect to their north facing arrows, not where the droplet itself is placed. In Figure 5 above, the heading angles of both the RX droplets is around 70 degrees.

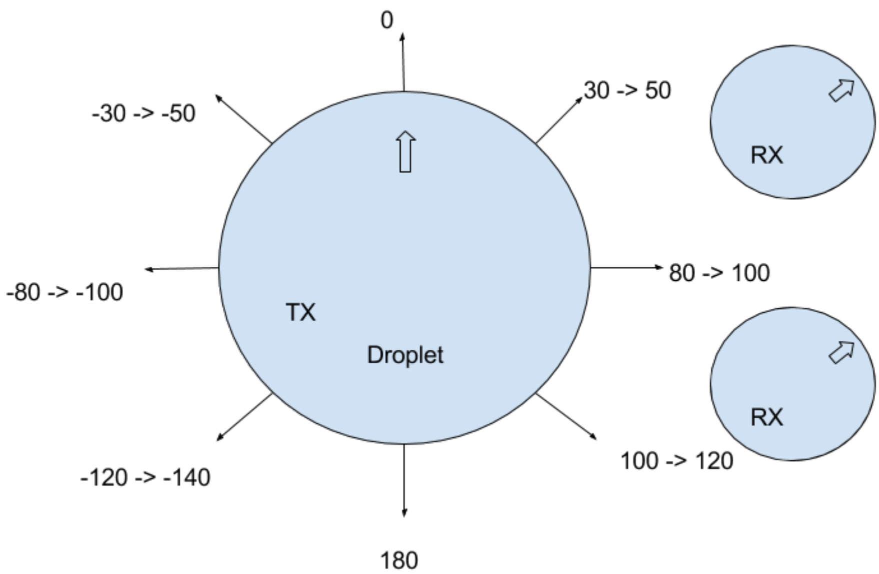

Figure 4: Heading example

A detailed description on how to use API calls to use the RNB data are attached in code snippets in appendix.

2.2 Calibrating Droplets for Motion

Uncalibrated droplets do not have their motors pointing towards the correct spin direction by default. To calibrate the droplets, it is important to understand how the droplets move. Using three (or two in the case of audio droplets) motors, the droplets vibrate sequentially to either move forward, turn clockwise or anti clockwise.

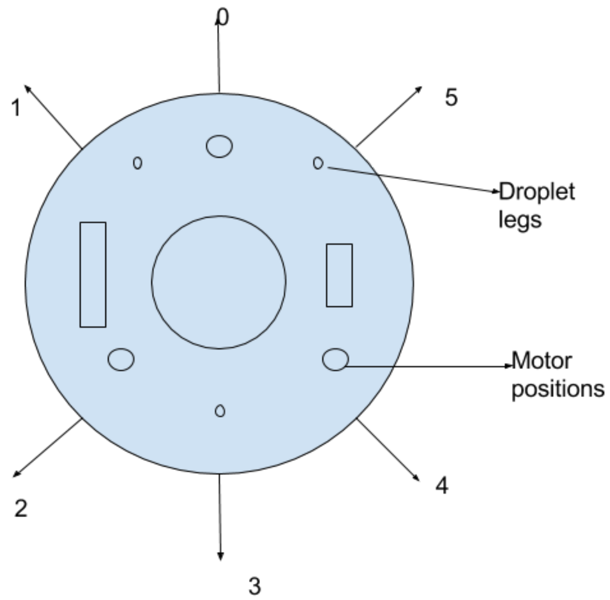

Figure 5: Flipped droplet

For example, for the droplet to move forward (in direction 0), the motor closest to direction 0 should not rotate and the motors near directions 4 and 2 should rotate anticlockwise and clockwise respectively. This results in a stick slip motion of the droplet legs near directions 1,3 and 5 in **-**, **NULL** and + directions respectively. This can better be understood by looking at the motion of the legs opposite to the motor direction.

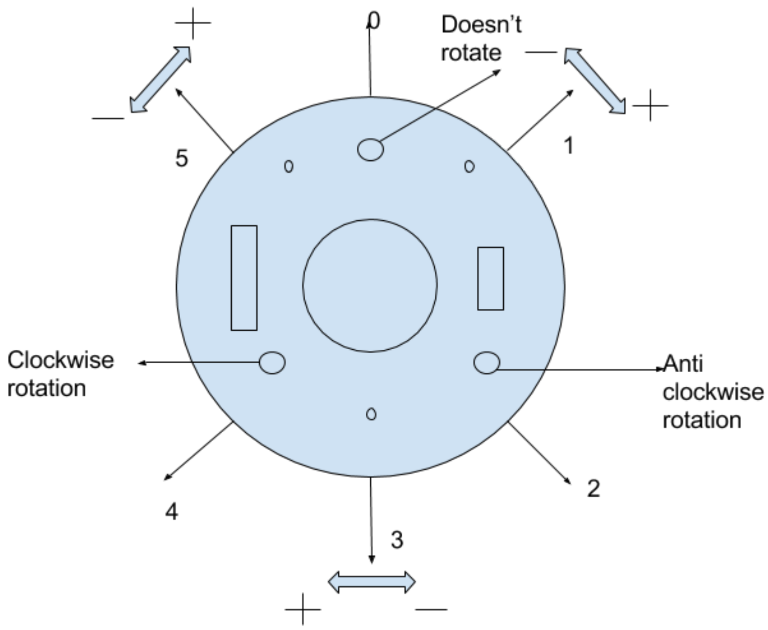

Figure 6: Calibrating a droplet for forward motion

Figure 5 shows a droplet which is flipped indicating its legs and motor positions inside the droplet. Hence for the same case as before, for forward motion, the leg near direction 3 does not move and the legs near directions 5 and 1 should move clockwise and anti clockwise respectively. This type of motion is called stick slip motion. Detailed explanation on how to calibrate these droplets using API calls are explained in the attached code snippets.

2.3 New API function calls related to calibration

The API function call **read motor settings** is used to check the motor setting values of another droplet. This is added so that the droplet does not have to be plugged in to check its motor settings calibration values serially on the terminal. Using this new API call, one droplet can receive the motor setting values of any targeted droplet and check if it is calibrated. The function **check if motor calibrated** can be used at the beginning of any program to see if the current droplet is calibrated or not. These function calls can be used to check if a particular droplet has been calibrated **wirelessly** instead of having to plug it in. This eases the process of calibrating the droplets. Some of the other functions that were implemented include **follow edge with rnb**, **move to closest** and **check ir coll max**, which are more related with shape formation whose implementations will be discussed in the forthcoming subsection.

2.4 Shape Formation: Edge Following

The paper on kilobots[4], Programmable self-assembly in a thousand robot swarm suggested that the first step towards shape formation is edge following. Edge following is the motion of the droplet towards a way point in the swarm. Edge following as the name indicates makes the droplet follow an edge until it reaches its desired position in the swarm. As a first step, I implemented the edge following with the raw IR collision values received from each of the droplets' individual IR sensor pairs. Using this function call, a droplet can follow any white surface (like another droplet or even a wall) in the clockwise or anticlockwise direction for a certain duration of time or till it covers a certain distance, say 90 degrees. This works if the droplet is close enough to the white surface that the IR collision values are able to detect the surface. If the droplet was far away from the swarm, the **follow edge with rnb** function is called. This function uses RNB data to get close enough to a droplet and then uses edge following discussed previously to navigate around the swarm. The next step in the shape formation process would be the gradient-formation and localization. Since localization has been implemented recently, the edge following method can be coupled to it and be used more efficiently. The Figure 7 below shows a pictorial representation of edge following in a swarm, borrowed from paper[4].

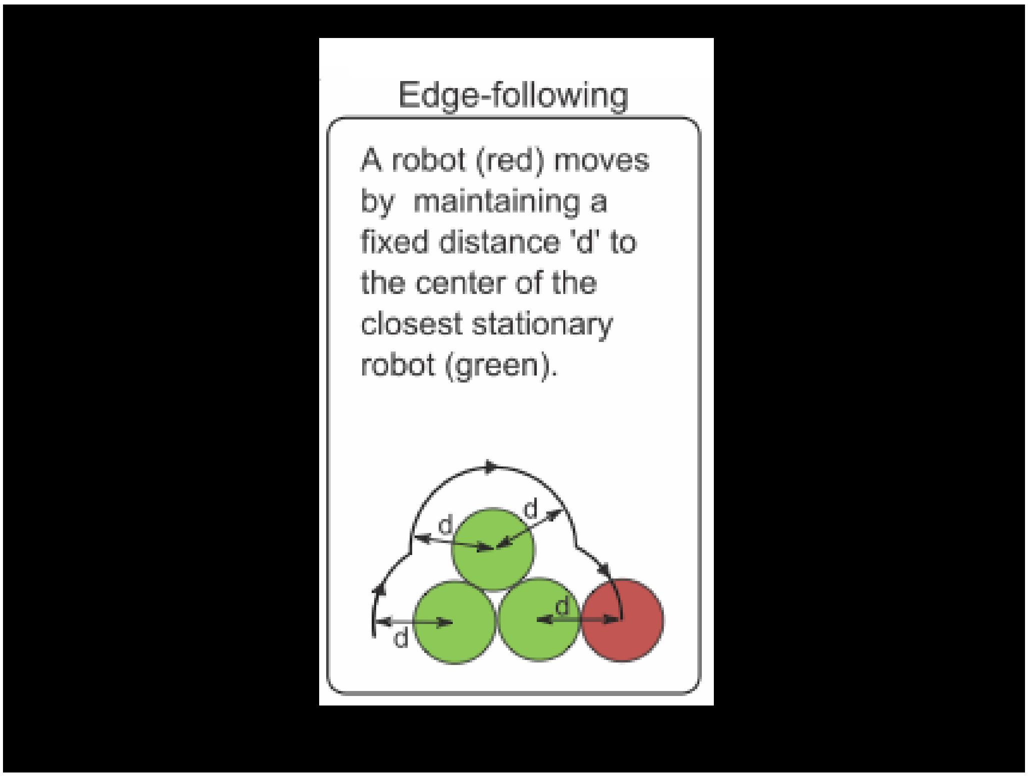

Figure 7: Edge following

3 Experiments

3.1 Problem with random hard vibrations

One of the problems that was present from the beginning and became more noticeable during the motor calibration and motion using edge following was the **random hard vibrations** that occurred whenever the droplet **made a change in direction** ie., clockwise to anticlockwise or vice-versa. The reason for this vibration was thought to be due to setting the motor settings values to very high but the problem also persisted when the motor values were average or low. This was an important problem to debug since these **random vibrations can throw the droplet off its desired path and sometimes even drain all its power causing it to reset.** Deciding to debug this problem I simulated a case of manually changing directions and measuring the PWM on each of the three motor pins using an oscilloscope while the random vibration occurred. It was noticed that the PWM of one of the motors were longer and this caused the vibrations. The problem was discovered to be due to **PWM in motors not being turned off when the droplet wasn't moving.** They were always on and just switched a particular GPIO when the droplet needed to move forward. This caused the tOFF times as described in paper [1] to be very small causing instability in the platform. The expected and the observed PWM waveforms similar to seen in the oscilloscope are shown in Figure 8 and 9 respectively.

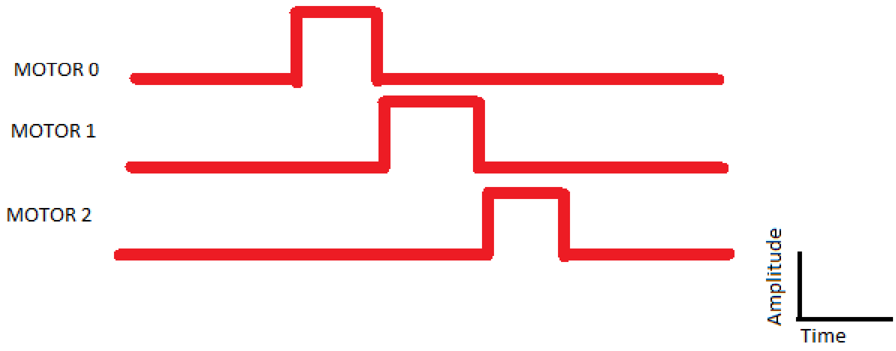

Figure 8: Expected PWM

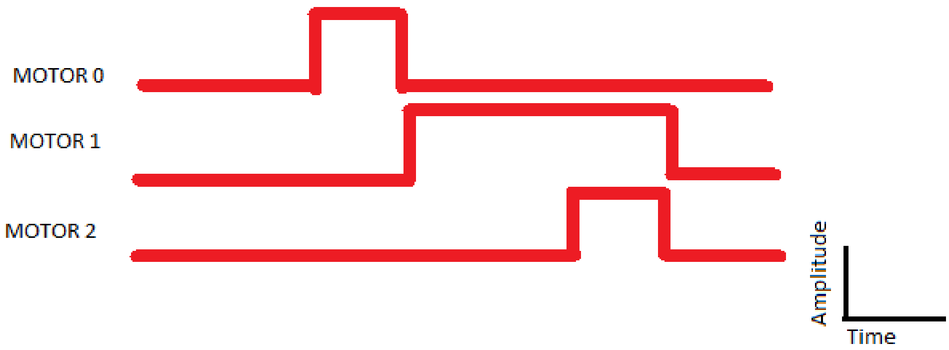

Figure 9: PWM obtained

The problem was fixed by altering motor initialization code in the **int motor.c** file under the **motor-forward()** function where the register **TCC0.CTRLC** was set to 0. This fixed the issue of the random vibrations during motion.

3.2 Edge Following - Different methods tried

Using range and bearing for edge following

Using RNB for edge following did not work well for me for the following reasons. When the droplets are very close to one another, their IR sensor- receiver pairs might not facing each other. This results in incorrect values of the range being displayed. For example, when the droplets are actually touching each other, a range of **90** mm may be observed. This distance is critical in edge following as the idea behind edge following is moving in close pairs and maintaining proximity for the shape formation. Moreover in some cases, the range and bearing values might not get updated if droplets are not in close proximity with each other and if they are very far away (more than **10-15 cm** away. If they are too close to one another, the RNB value might not get updated and if they are too far away, the bearing values were observed to be erratic. The video can be found in the attached appendix in the Videos folder.
Note: this was observed only in some cases even after setting the necessary flags for audio/ non-audio droplets. The reasons were thought to be because of the ambient light conditions. In order to make the edge following algorithm more dependable and generic across all droplets and environment conditions, I decided to use only the consistent IR collision values from droplets.

Using IR Collision values for Edge Following

When the droplets are close to one another at the beginning, the IR collision values give a good estimate of which the closest white surface is to the droplet. This white surface could be a wall or another droplet. Using just the IR collision values, the function called **Find IR max** which can be used to find the maximum collision value and the direction from which the maximum value was found. Edge following implemented using this gave more predictable results. A video demo of this edge following can be found in the appendix in the **Videos folder**. Another approach is using a combination of RNB and IR collision values: at far distances (within which RNB is accurate), determine droplet position and draw droplet closer. Once close enough, use the IR collision values to do edge following.

There was still a problem in this method namely how to determine if the droplet is close enough using IR collision values because RNB is inaccurate at close ranges? I have collected more experimental data in regards to finding a solution for this, explained in more detail in the following sections.

3.3 Summary of important related papers and articles

The most relevant papers that I read and have used in this report are :

1. A stick-slip Omnidirectional Powertrain for low-cost swarm robotics: Mechanism, calibration and control

   The main points I took away from this paper include how the droplets move by the stick-slip motion in 1-DoF using the vibrational motors and how they can turn in different directions and move using the phasing of the three motors in different orders. This helped me in studying and calibrating the motion of the droplets- I understood how the motors can be placed differently while being soldered and mounted on the droplet and how the same PWM might cause different vibrational outputs on different motors. Understanding this is important as one can relate why the droplets can be low power and wouldn't need huge batteries as this motion is more efficient than using a conventional motor for motion. Later on while debugging the **random, hard vibration problem** in the droplets, it became simpler to figure out that the issue could not be due to incompatibility between the motors used and the H bridge since the models used were specified in this paper and a proof of concept of their working was also provided.

2. Robust coordinate Systems for swarms of miniature robots

   In this paper, I understood the core logic in how the Droplets might be able to communicate with each other using a Bayesian update- or a probabilistic method of getting information about its local neighbors in a swarm. I studied this paper as it might be useful for me during the shape formation implementation. I understood that a simple hop-count coordinate system deployment in swarm robotics based on IR communication is not consistent always because of the many shortcomings related to the firefly synchronization and the IR communication between droplets. The duration of this algorithm is restricted by the **width of the TDMA slot** for each droplet. So for this method to work and for the droplets to reliably get a position update about all its neighbors, it would take about 73 seconds long and this duration would increase if the number of droplets in the swarm increased.The problem related with using the Infra Red as a means of communication was that the surrounding environment where the experiment is carried out should be dark, completely free from any IR interference for this method to work accurately. Because of the above reasons a Bayesian update method is not the best way to get an estimate of the coordinate locations of the droplets.

3. Miniature Six-channel Range and Bearing System: Algorithm, Analysis and Experimental Validation

   This paper provided the basis of the range and bearing measurements. It explains in detail the mathematical model for range and bearing estimation. An automated system was used to test the high accuracy of the sensor, emitter and amplitude model by repeatedly positioning the angles by 0.9 degree increments and adjusting the range to an accuracy of

0.5mm. The orientation of the transmitting droplet relative to the receiver can also be directly calculated by this approach.

4. Programmable self- assembly in a thousand-robot swarm

   This paper provides some very deep insights and gave me ideas on how to start with shape formation. Using local interactions and by developing a collective algorithm for shape formation, a programmable self-assembly of a complex two-dimensional shape with a thousand-robot swarm was demonstrated. The main methods discussed in this paper are edge-following, gradient formation and localization. To avoid congestion in movement of outer robots in a swarm, randomness was used to allow only a subset to start motion at any one time. The robots used the distance to nearby neighbor information to collectively construct a coordinate system. The concepts used in this paper along with localization and self-programming were very helpful in understanding shape formation in a swarm of robots.

4 Discussion: Moving forward

4.1 Study to measure feasibility of using IR values to measure distance at close ranges

The main motivation for this study was to see if there a more accurate alternative to the RNB method to measure distances at close ranges. The RNB is not accurate at close distances for reasons mentioned in section 3.2 in this report. This is because the RNB method requires **both the transmitting and the receiving droplet** to participate in the measurement by sending and receiving commands via IR. Even if one of the droplets is not properly oriented (which is very likely to happen at close ranges) during the measurement, the data obtained is inaccurate. A simple alternative to this method is to measure the IR collision values by bouncing them off a stationary, non-participating droplet and thereby get an estimate of how far the measuring droplet was from this stationary droplet (or any white surface) Upon investigating that the sum of the maximum IR collision value and the IR baseline value for a particular IR sender-receiver pair was always equal to 2047, which was the maximum saturation of the ADC of the micro-controller. This meant that the saturation for any IR- receiver pair, irrespective of the droplet was always a constant. This meant that all the droplets (for every direction) reached the highest saturation when measuring the IR values, placed near a white surface. Figure 10 above shows the variation of the IR values at different distances for different droplets.

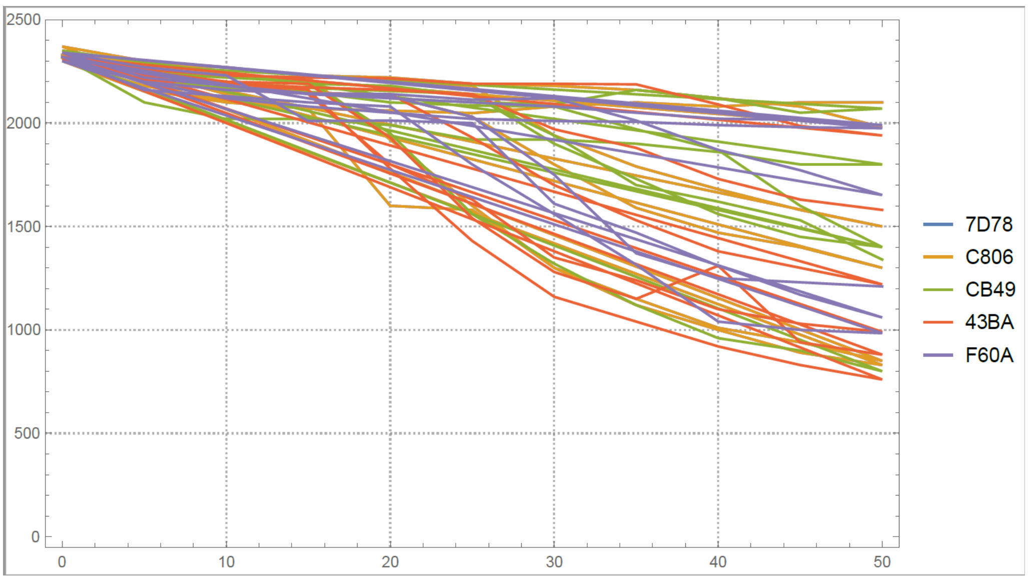

Figure 10: IR values by droplets

For this study over 520 data points were measured across several droplets at different distances and surfaces. Each IR-emitter pair for every droplet was moved from a stationary white surface (a flat surface like a box or a stationary droplet) at different distances and their corresponding raw IR collision values, subtracted from the baseline values was updated every 5 seconds to account for any change in the ambient light. The results were plotted to observe any similarities between data at different distances. The values on the x-axis denote the distance in mm between the droplet and the surface from which the IR light was bounced off. The y-axis denotes the raw IR collision values obtained from the ADC pin of the droplet. Figure 11 shows the variation for different IR-emitter collector pairs on different droplets.

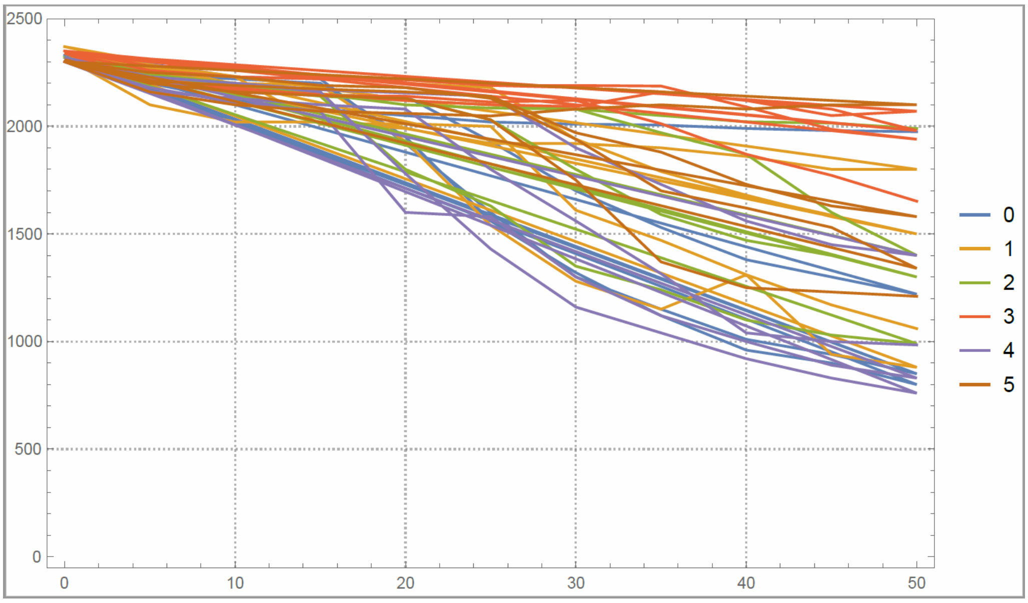

Figure 11: IR values by direction

On looking at the IR collision values, I noticed that the ranges of the values obtained were not the same for each droplet and not the same even for each IR sender-receiver pair. The baseline values took into account, the ambient light and adjusted the collision values accordingly to give more uniform results. Some other data collected later also revealed that there is a decrease in the IR values with distance, but there is no correlation between the rate of decrease of values and the distance itself. This could be because the emitter power for different IR emitter pairs could be different.

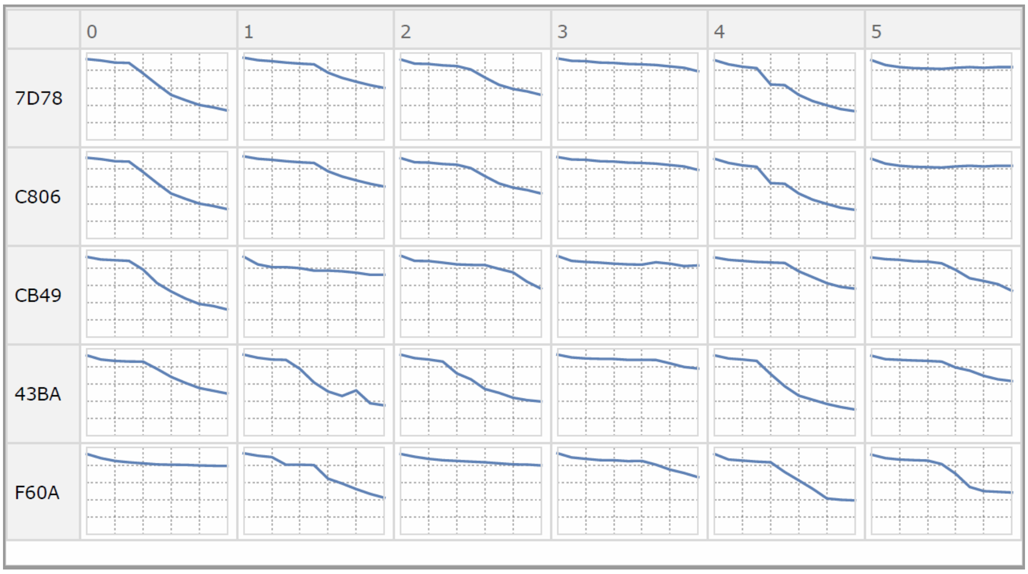

Figure 12: All droplets in different directions

This study has provided some valuable insights on how the range varies with distance and that it is not uniform across droplets or among IR-pairs on the same droplet.

4.2 Using Localization and self-programming for shape formation

Using the current localization algorithm that is being developed, a **map** of the droplets' movements can be recorded in real time and a visualization of any obstacles in the swarm's motion of field can be recorded. This can be used to get an idea of the actual field or terrain where the droplets are deployed. This map can also be used to keep track of where the various droplets are positioned to be used in shape formation. Coupled with self-programming, the droplets will have more flexibility towards shape formation by updating any related and necessary positional information.

5 Conclusion

To summarize, this study has given me more insights on calibration and movement using stick slip motion and how to calibrate and use vibrational motors associated this type of motion. On the droplets platform specifically, I learnt about the range and bearing measurements, how to use them, some of the shortcomings related with them and an alternative to overcome these shortcomings. I also looked at the first step towards autonomous shape formation without the use external devices like a camera using edge following. I discovered some of the problems associated with platform instability during motion and fixed other roadblocks along the way towards shape formation. I would like to thank John Klingner for guiding me in every step along the way and suggesting a lot of useful ideas and solutions to help me when I was stuck in any problem. I would also like to thank Professor Nikolaus Correll for allowing me to work on his research platform and giving me the liberty to contribute my own ideas to it.

6 Appendix

The appendix can be found attached in this report. The various folders attached in the appendix are

1. API calls for calibration - This folder contains documents on how to calibrate a droplet with API calls. It also contains example calibration values for many droplets which have already been tested for motion.
2. Code snippets - Code snippets of function calls and APIs I had implemented.
3. Study on IR values with distance - This contains the raw IR values with distance, to be used in future for any related experiments with the IR values with distance in droplets.
4. Videos - Videos of the droplet functionality I implemented and faced problems with.