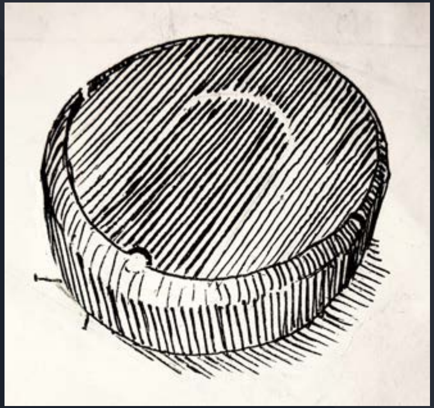
Roomba Location Using Acoustic Methods

Sean Golinski, Tim Huxtable, Pranay Jain, Sunny Patel



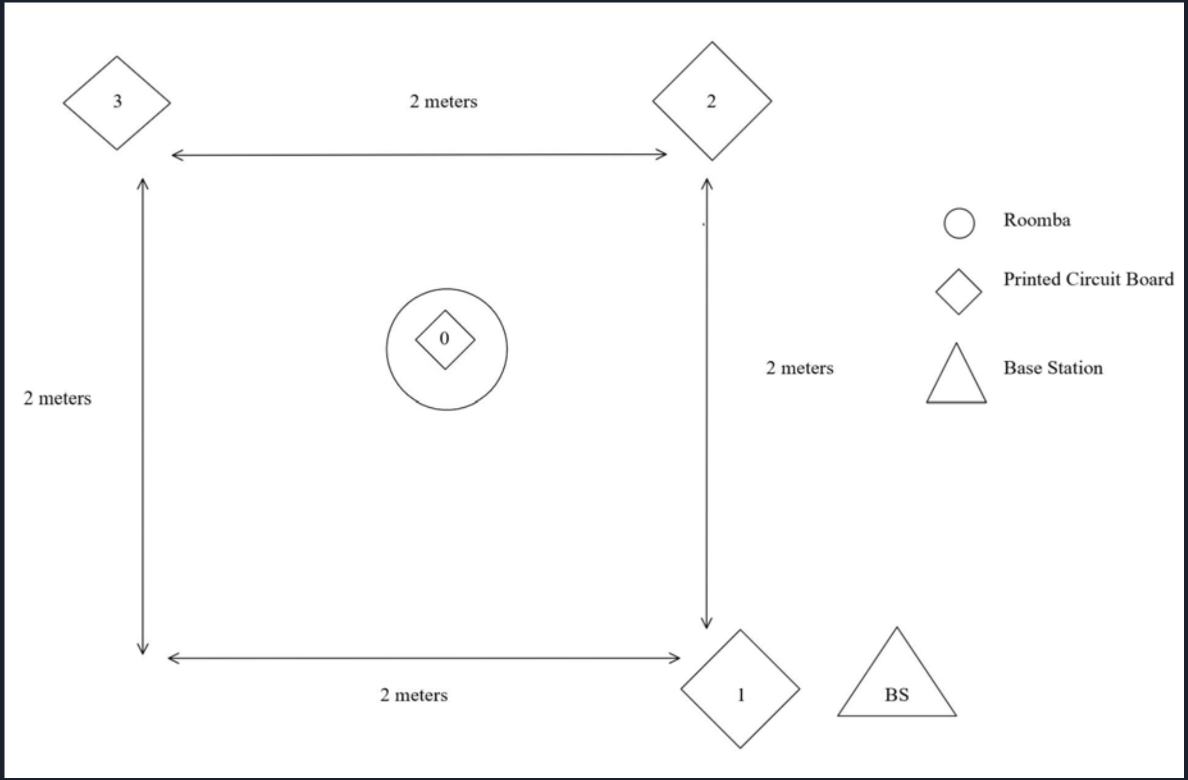
Project Overview

- Our group set out to find an alternative way of tracking Roomba and Roomba-like vacuum cleaners.
- First, we had to analyse current systems. They vary all the way from simple IR units (similar to TV remotes), all the way to 3-d topology construction using LiDAR on the more expensive units.
  - Plenty of drawbacks. Poor low-light performance and interference from obstacles hamper the IR models. Cost and implementation difficulties hold back the more advanced models.
- We decided to use acoustics to locate the Roomba. Sound, in general, requires less processing, is easier to implement and significantly cheaper in most cases.
- Our system consists of a sound source on the Roomba itself, in conjunction with 4 ‘listeners’. These listeners compute the properties of incident sound to determine the position of the Roomba.

Methodology

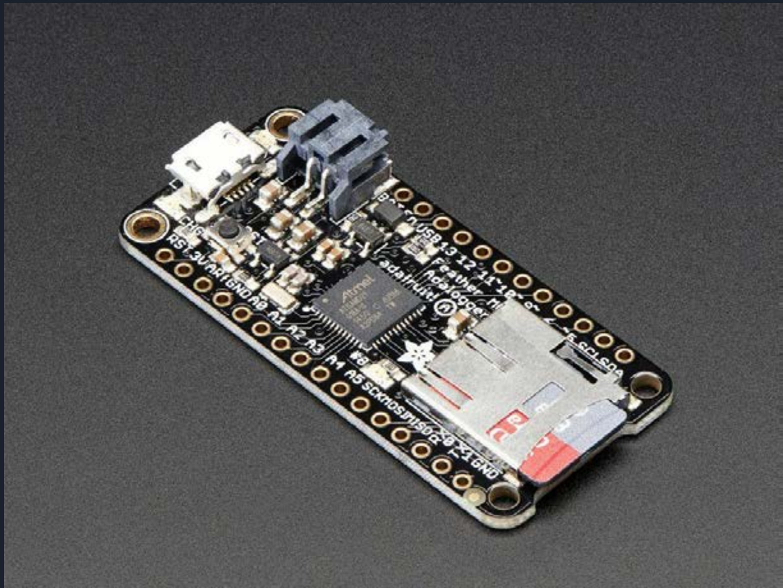
- Our experiment is described below:
  - Attached to the Roomba is one of our listener modules, and a tone generator (smartphone).
  - 3 other listeners are placed around the room.
  - A tone is played from the Roomba.
  - At each listener, the microcontroller performs a Discrete Fourier Transform on the incident waveform.
  - Mathematically, we can extract a phase difference between the listener on the Roomba, compared to the listener situated in the corner of the room. This is then very easy to convert to a 1-D distance.
  - The same process, when executed simultaneously on 3 listeners, can provide a very accurate (x, y) location for the Roomba.

Diagram of Setup



Hardware - Adafruit M0 Feather Adalogger

- Microcontroller with faster on-board processor when compared to the Mega 2560. We needed a fast processor because we are running Discrete Fourier-Transform calculations and our location algorithm on the adaloggers.
- Speeds of 48 MHz
- Storage of 256 kB and 32 kB of RAM



Hardware - GPS and RTC

- GPS - From the GPS we are interested in the PPS signal. The PPS signal is what allows for all of our printed circuit boards to stay synchronized with each other.

- Real Time Clock - We used this to ensure that all of our measurements were taken at the same time.

- Gather data, run DFT, and send data to our Base Station within four seconds.

Hardware - Amplified Electret Microphone

- Record amplitudes over time of incoming sound waves
- Comes with MAX4466 op-amp.
- Bandwidth of this microphone is 20-20kHz

Hardware

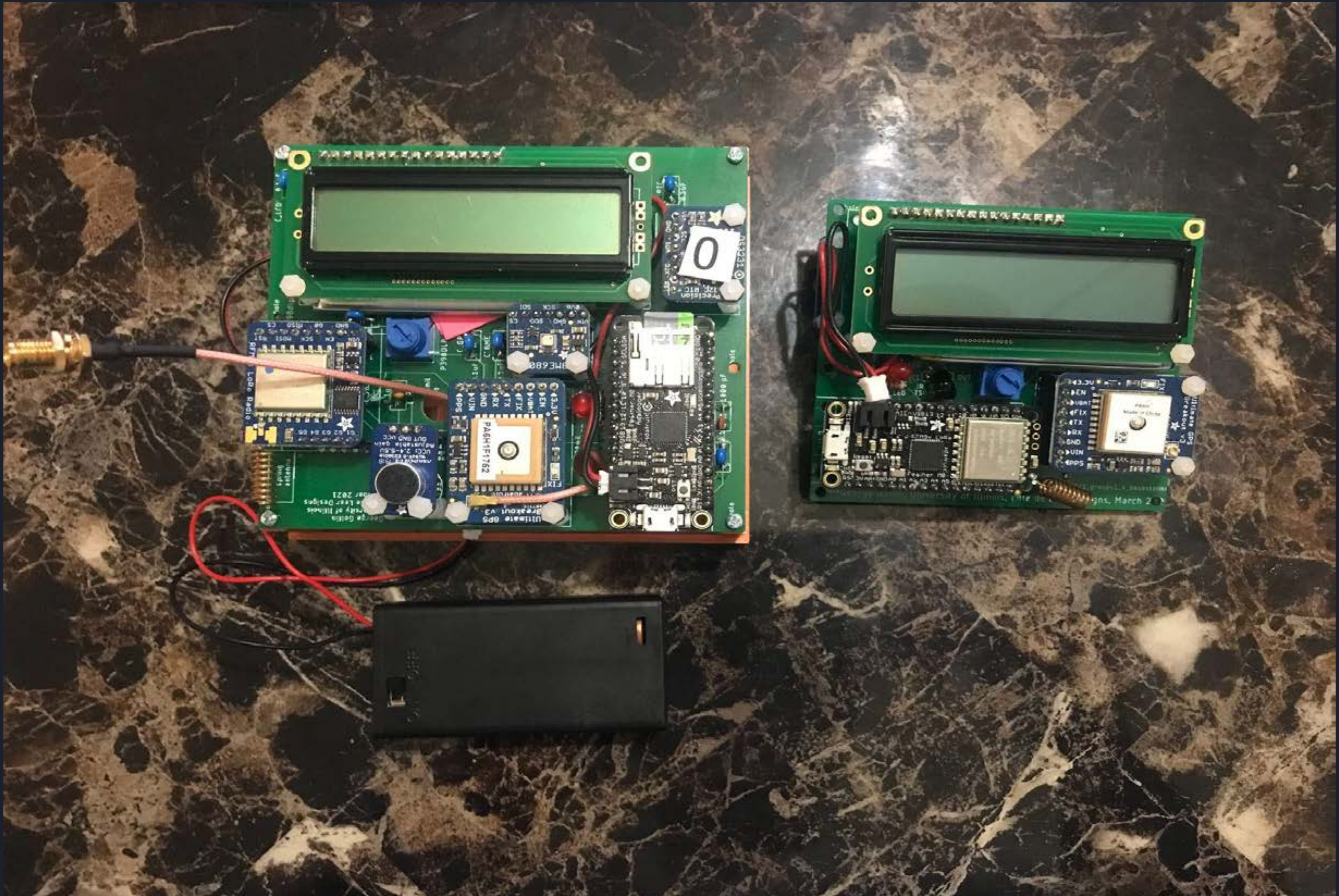
- Liquid Crystal Display (LCD) - This was just used for feedback information of what was going on in the printed circuit board. The LCD would tell us when we received signals from satellites, what part of the code was being run currently, and any errors that came up.

- LoRa Radio (Long Range Radio) - How data was sent from the PCB receiver stations to the Base Station.

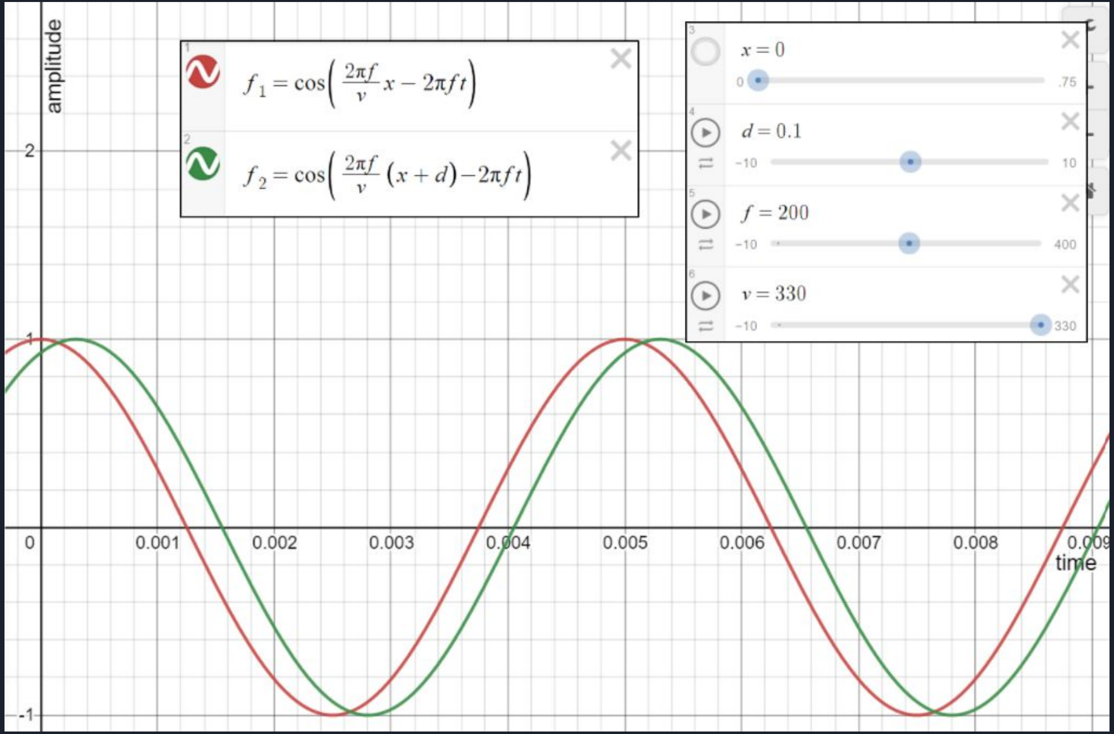
- BME 680 - Device used to measure Temperature, Barometric Pressure, Humidity, and VOC.

Hardware - Tone Generator

- We used a website called “Online Tone Generator by Tomasz Szynalski” for creating a tone of specified frequency.
- We used a frequency of 200 Hz
- 200 Hz creates long enough wavelength to ensure the Roomba doesn’t move more than a single wavelength away in a measurement period so the phase angle never relapses.



Phase Shifts



Fourier Transform

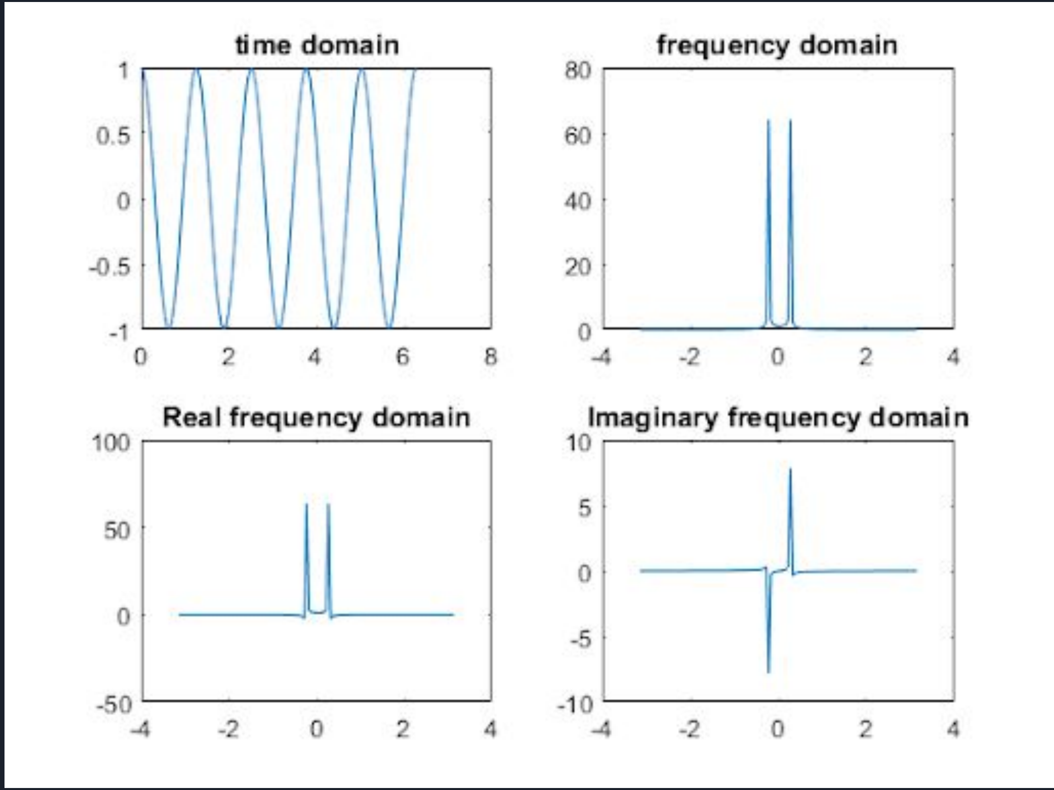
$$F(g(t)) = \int_{-\infty}^{\infty} g(t)e^{-2\pi ift}dt = \int_{-\infty}^{\infty} g(t)(cos(2\pi ft) - i * sin(2\pi ft))dt$$

Orthogonality Principle

$$\int_{-L}^{L} cos(\frac{n\pi x}{L})cos(\frac{m\pi x}{L})dx = \int_{-L}^{L} sin(\frac{n\pi x}{L})sin(\frac{m\pi x}{L})dx = 0 , m \neq n$$

$$\int_{-L}^{L} sin(\frac{n\pi x}{L})cos(\frac{m\pi x}{L})dx = 0$$

Fourier Transform



(Tim Bower, 2021) Kansas State University – Polytechnic Campus

Discrete Fourier Transform (DFT)

$$F(g(t)) = \sum_{j=0}^{J-1} g(j)e^{-2\pi i f(\frac{jT}{J})} = \sum_{j=0}^{J-1} g(j)(cos(-2\pi ft(\frac{jT}{J})) - i * sin(-2\pi ft(\frac{jT}{J}))$$

J = number of data points in microphone array

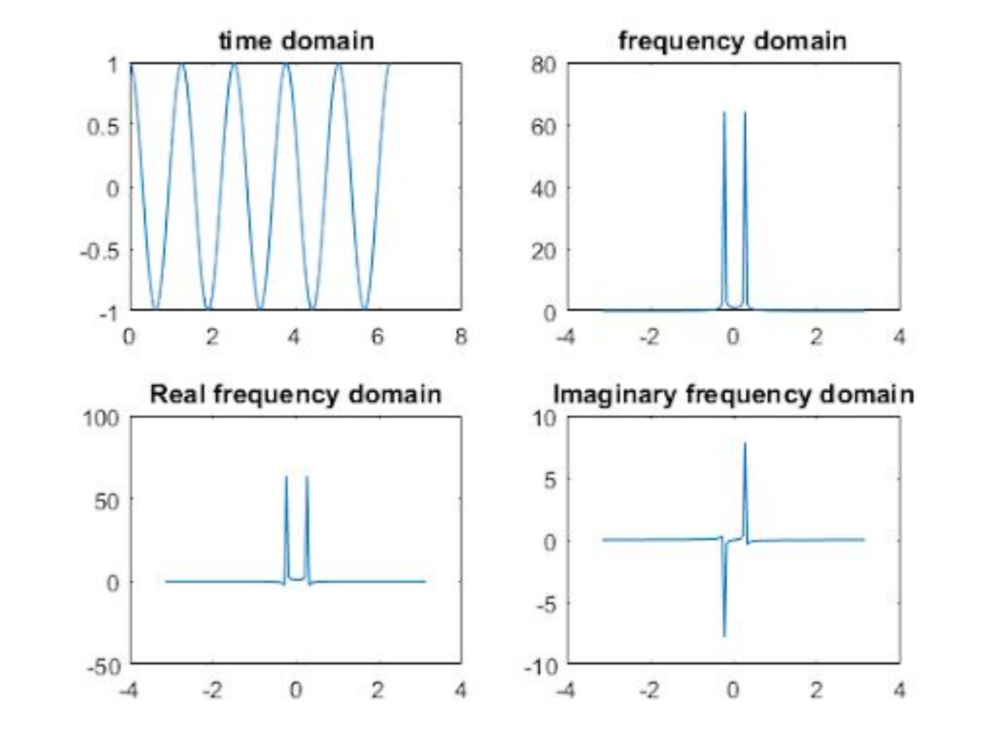
j = current index

T = length of time over which microphone was recording

g(j) = microphone reading at index j

f = frequency we are interested in measuring

Fourier Fun Facts

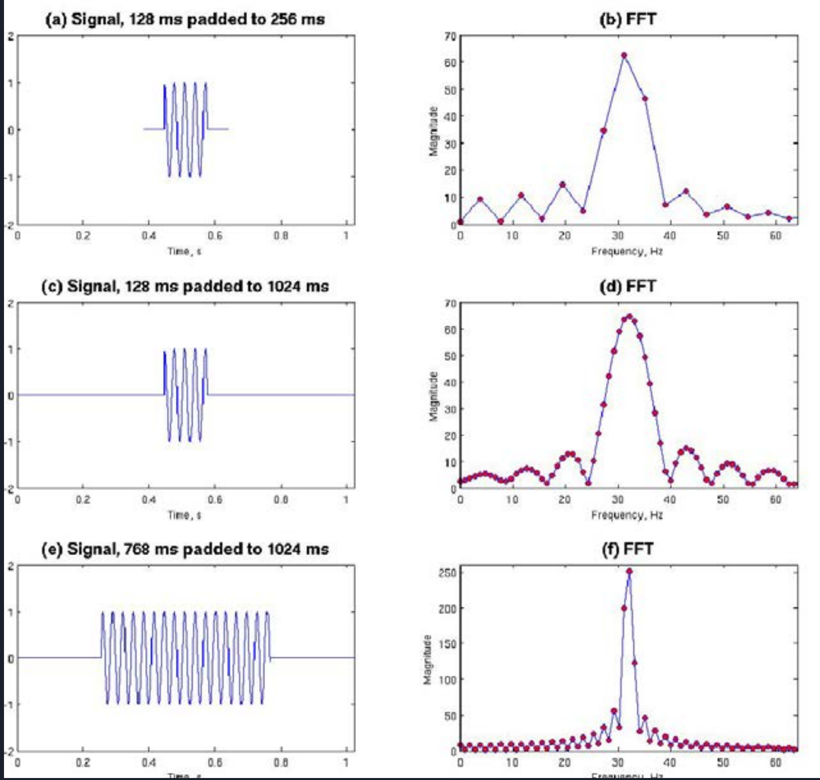


$$g(t) = \int_{-\infty}^{\infty} G(f) e^{i2\pi f t}\, df$$

$$g(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} G(\omega) e^{i\omega t}\, d\omega$$

(Tim Bower, 2021) Kansas State University – Polytechnic Campus

Fourier Fun Facts



Uncertainty in Spectral Decomposition, https://subsurfwiki.org/wiki/Uncertainty_in_spectral_decomposition

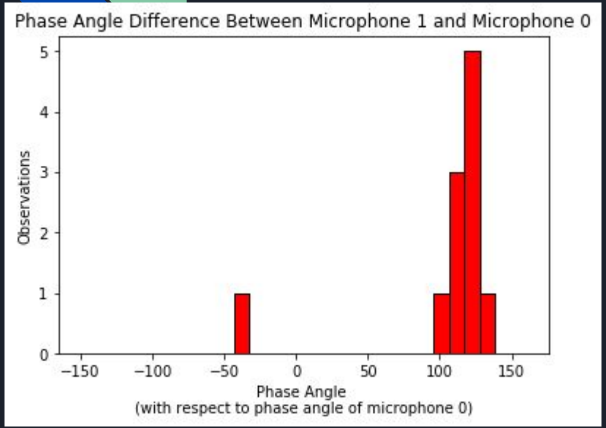
Location Algorithm

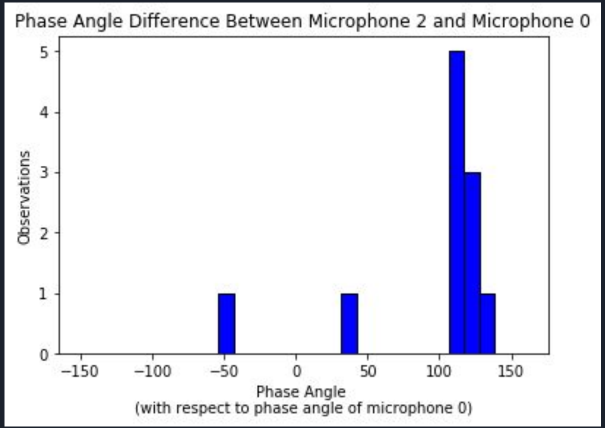
- After performing the Fourier Transform analysis, we are left with a phase difference between the Roomba and each of the 3 listeners. The location algorithm takes in these 3 inputs and tries to extrapolate an (x, y) location for the Roomba.
- From the phase data of a known wavelength, it is easy to calculate the path distance in metres. Simply, (phi * wavelength / 2π).
- There is a problem with periodicity. Since the wave is periodic, there are an infinite number of solutions for the distance, each varying by 1 full wavelength.
- If d is a solution, so is d + lambda, d + 2*lambda, …
- Initial Algorithm
  - We initially tried using a purely mathematical approach .
  - We know the (x, y) location of each of the three listeners. We also know the most likely distance from each receiver.
  - Mathematically, a system of equations with 2 listeners should give us 2 possible locations. However, with 3 listeners, theoretically this should collapse to a single (x, y) solution that satisfies the x and y distances for each listener.
  - We tried SymPy to solve this system of equations. But, this did not work all the time.

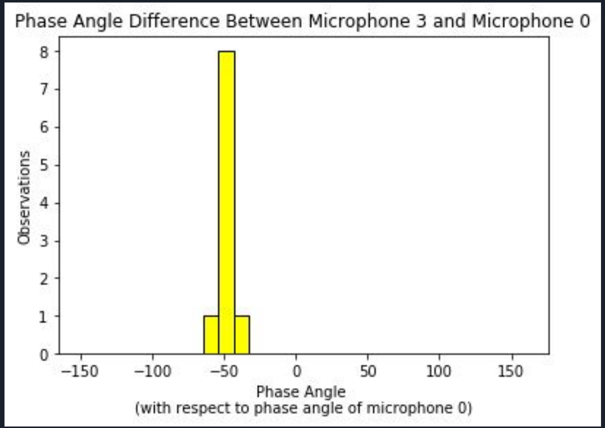
Location Algorithm

- Final Algorithm:
  - We employed a grid search.
  - This would create a grid of evenly distributed points throughout the room, and calculate the phase difference of each point to each listener.
  - This gives us 3 expected phase differences for each point.
  - Next, we took the 3 experimental phase differences, and ran a chi-square analysis to find the grid slot which had the closest values.
  - This becomes our most likely Roomba location.
- Brute force solution that relies on the computation power of the SAMD21 Chip on the Adalogger. We got it to update every 2 seconds.
- Periodic ambiguity? We increased the wavelength of the test sound to 200 Hz, so that all of our grid points would fit within 1 wavelength.
- This guarantees a unique solution.

Test 1 Results - Consistency of Phase Differences







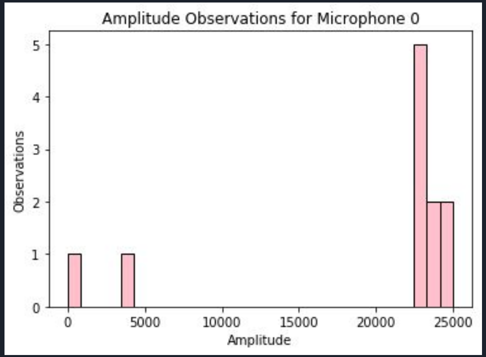
mean: 104.75
standard deviation: 47.16

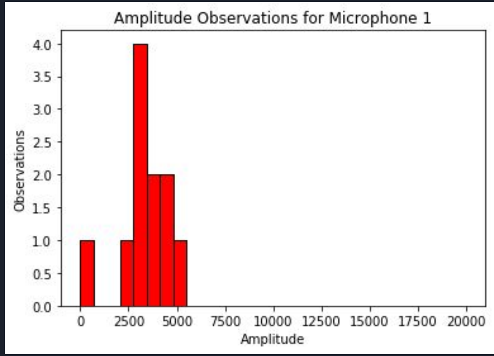
mean: 95.66
standard deviation: 50.69

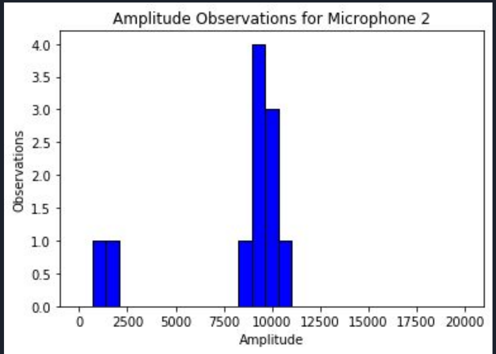
mean: -71.7
standard deviation: 73.28

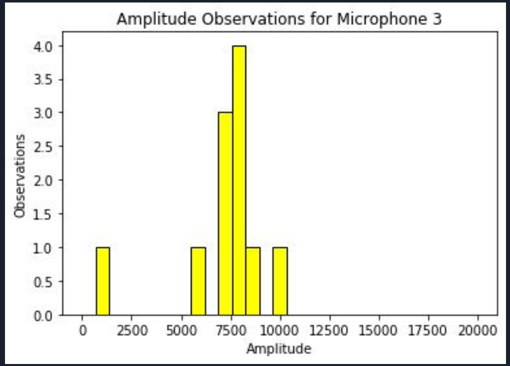
(outlier not shown near -300)

Microphone Reception

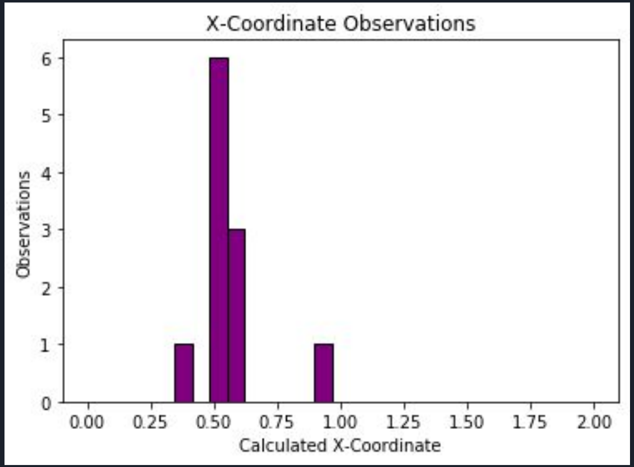


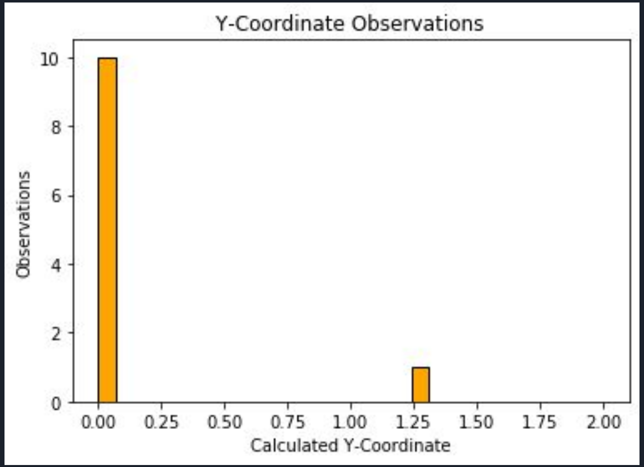






Results - Position (0.53, 0.318)

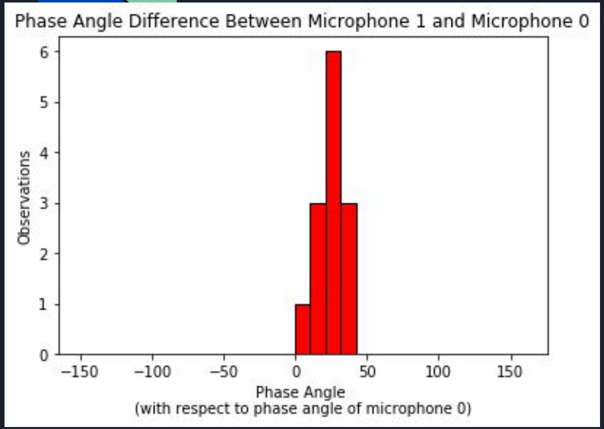


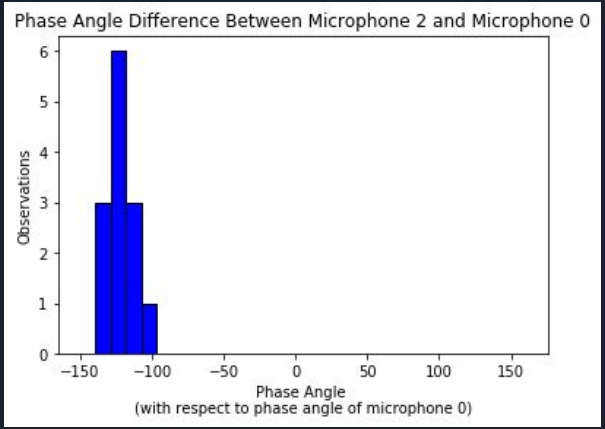


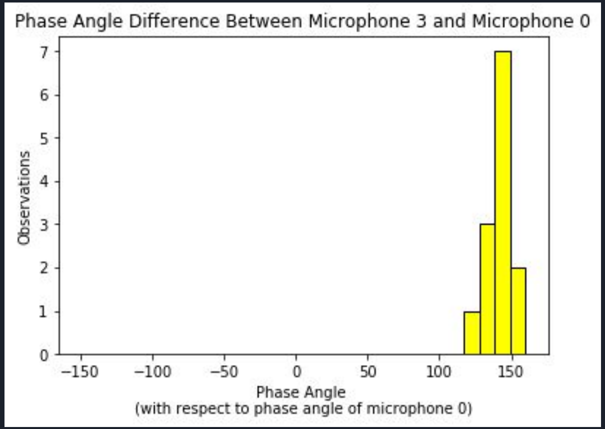
mean: 0.58
standard deviation: 0.12
mode: 0.55 (n = 6)

mean: 0.12
standard deviation: 0.37
mode: 0.0 (n = 10)

Test 2 Results - Consistency of Phase Differences





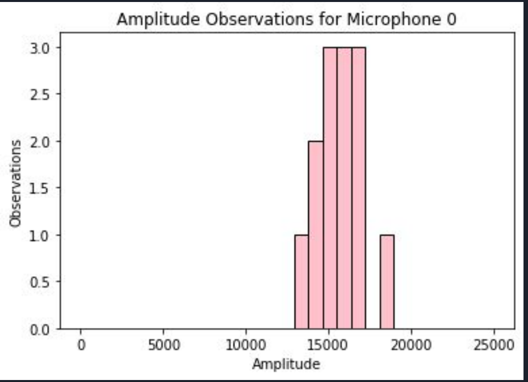


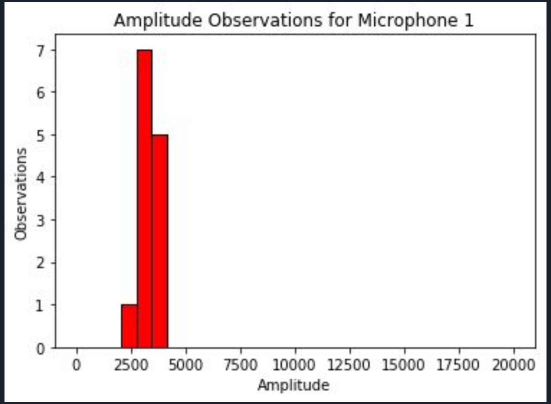
mean: 25.31
standard deviation: 9.98

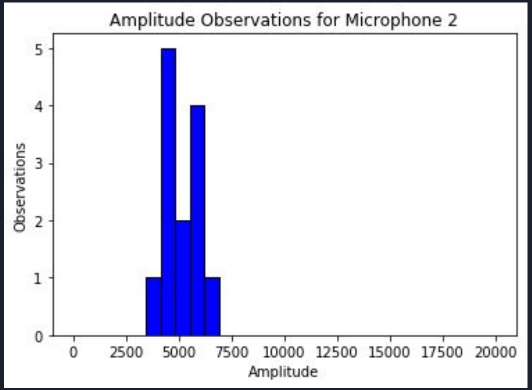
mean: -120.96
standard deviation: 8.42

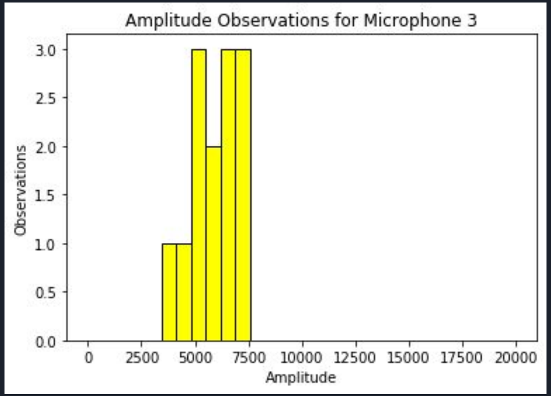
mean: 141.35
standard deviation: 8.95

Microphone Reception

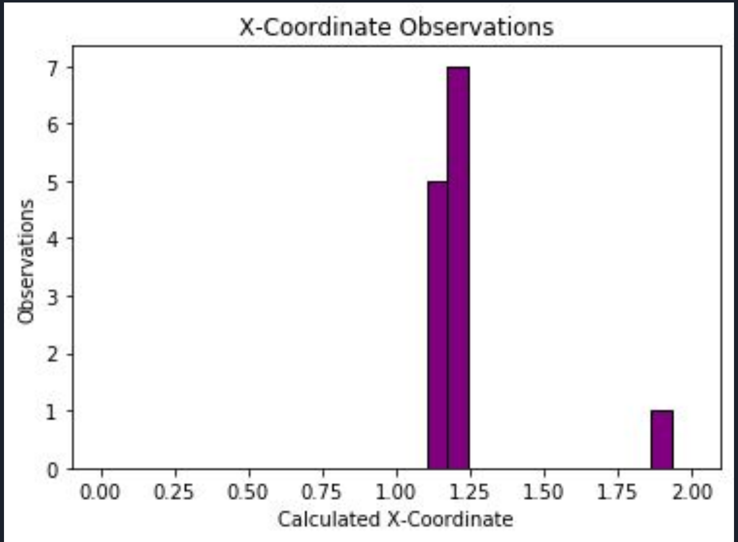


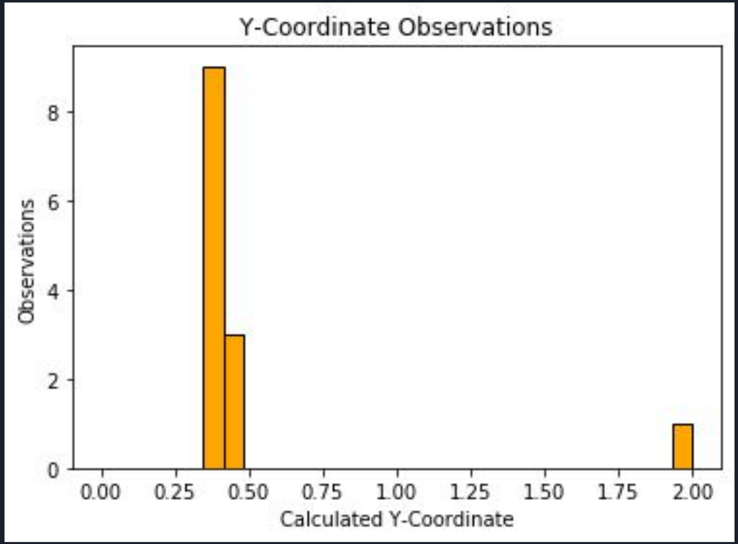






Results - Position (1.12, 0.41)





mean: 1.23
standard deviation: 0.19
mode: 1.2 (n = 7)

mean: 0.52
standard deviation: 0.41
mode: 0.4 (n = 6)

Conclusion

- We set out to come up with a better, cheaper, or easier solution to the problem of locating a Roomba within a room.
- Current solutions were either bad, or expensive. Acoustic methods was a novel approach that would be much cheaper to implement.
- We had limited success. Our first algorithm was not really functional, but the grid search algorithm had good success.
- Errors: standing waves, reflections, echoes and reverberations.
- Results: most of our results were good. Some were not, which we attributed most significantly to echoes. We were in a room with hard walls.
- Proof of Concept: we can find the location of a Roomba using acoustic methods. With refinements to our equipment and our algorithm, this seems like a genuinely functional approach.

Conclusion (contd.)

- Successes:
    - Cheaper than most other solutions. Particularly LiDAR Roombas.
    - Very easy to implement.
    - No line of sight required. Light guidance requires direct line of sight.
    - Open-source software + hardware.
    - (x, y) mapping. Cheaper Roombas only look for obstacles and avoid them. They cannot actually determine where they are in a room.
- Drawbacks:
    - Annoying sound. Higher frequencies exacerbate our errors. Cannot go lower.
    - Obstacles. More echoes, reverb, scope for errors.
    - Needs GPS connectivity, which may not always be possible.
    - Needs listener modules to be placed around the room. They also need to be powered.
    - Current Roombas do not need external modules.
    - Precision. The grid search can never be an exact location, but is rather an approximate location. We can maybe find the location to less than 6 inches, but that is still enough error to bump into chair legs, fall off staircases, etc.
- We can call our project a limited success. It was a successful proof of concept, but is not yet a rigorous or commercially implementable solution, though it does have its market.

Questions?

References

"Electret Microphone Amplifier - MAX4466 with Adjustable Gain." *Adafruit*, https://www.adafruit.com/product/1063. Accessed 3 Dec. 2021.

Szynalski, Tomasz. "Online Tone Generator." *Online Tone Generator - Generate Pure Tones of Any Frequency*, https://www.szynalski.com/tone-generator/.

"Adafruit Feather M0 Adalogger." *Adafruit*, 2021, https://www.adafruit.com/product/2796. Accessed 3 Dec. 2021.

Bower, Tim. "Real and Imaginary Components of a Fourier Transform." Kansas State University – Polytechnic Campus, http://faculty.salina.k-state.edu/tim/mVision/freq-domain/DFT.html. Accessed 1 Dec. 2021.

"Fourier Transform Uncertainty Principle." Uncertainty in Spectral Decomposition, https://subsurfwiki.org/wiki/Uncertainty_in_spectral_decomposition. Accessed 1 Dec. 2021.