Assignment 7 - Sound Localization

Sreeram R EE23B075

November 6 2024

1 Introduction

The goal of this assignment is to write code for reconstruction an image using the Delay and Sum Algorithm which is a widely used algorithm in Ultrasound Image reconstruction. The setup mentioned in the assignment is that a set of microphones are placed at y-axis centered at origin and an obstacle is placed at a position with positive x coordinate. Various questions have been asked in the doc, which on answering guides us to the logic to be used to solve the given problem. The explanation of the code is done along with answering those questions.

2 Explanation of Code and Answers :

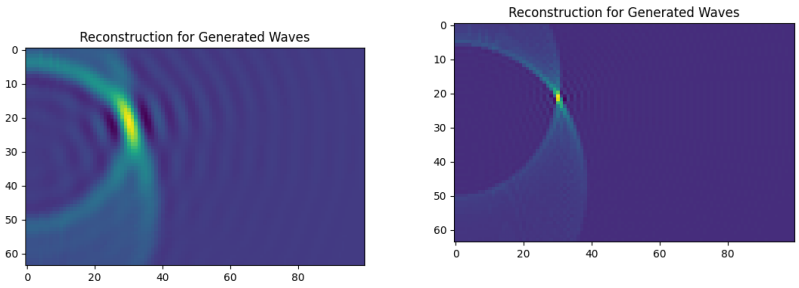
The first part of the assignment is as follows: We are given the number of mics which are placed along the y axis symmetrically. We are also given the position of the obstacle. The task is to write a code for generating samples recieved by the microphone due to the reflection of the sinc pulse from the obstacle and to plot the output of every mic.

The idea is that the output signal of the microphones is going to be a shifted sinc signal but to calculate by how much the signal has been shifted, we need to know what is the delay experienced. Delay is experienced due to the difference in the distance travelled by the wave to reach the microphones from the source. After computing the delay the microphones's output signal is calculated as the shifyted sinc signal where the shift is nothing but the respective delay.

2.1 Questions:

- The plots below show two example sinc pulses. How will you generate pulses that look like this? Which parameter should be changed? What effect do you think this will have on the final image?

  The parameter `SincP` determines how broad the curve is ie. If the value of `SincP` is high then it means that the Sinc signal is going to be compressed. On using a high value of `SincP`, we get compressed signal and the images obtained are of higher resolution.



(a) SincP=3 (b) SincP=10

Figure 1: Reconstructed images for different scaling factor

- Does it make sense to reconstruct up to Nsamp? What value is more reasonable as an upper limit for the x-axis here?

  No, it doesn't make sense to reconstruct upto Nsamp/2 since the farthest object's location which can be detected is Nsamp/2.This is becuase the distance travelled by the wave responsible for the delay is the distance to the obstacle from the source and then from the obstacle to the mic. Hence the more reasonable upper limit for the x-axis is (Nsamp/2,0).

- The (x, y) coordinates corresponding to the maximum amplitude (yellow colour) is approximately (30, 22). Explain why this is the correct expected position for the given obstacle?

  The x axis in the reconstructed image refers to the sampling index and not the distance. The distance can be calculated as `samp_index x dist_per_samp`. In this case, `samp_idx`= 30 and the distance is 30 X (0.1)= 3. Similarly y coordinate is the index of the microphone. Hence 22 refers to the 22nd microphone. According to the transformation used, 1st mic whose actual coordinate is -3.15 is at origin after transformation. Hence the actual y value for 22nd mic is `-3.15 + pitch*21`, where pitch =0.1, on calculating we get the actual y value as -1.05. Hence this is indeed the correct expected position for the obstacle.

- What is the maximum obstacle x- and y- coordinate you can use and still have an image reconstructed?

  The maximum x- coordinate that can be used is equal to `Nsamp * dist_per_samp /2`. This is because the distance traveled by the wave responsible for the delay is the distance to the obstacle from the source and then from the obstacle to the mic.

  Suppose the obstacle is located on the y-axis. For the signal to travel from the source to the obstacle and then reach the highest microphone, let y represent the maximum possible y-coordinate of the obstacle. The y-coordinate of the highest microphone is given by $\frac{\text{Nmic}}{2} \times \text{pitch}$.

$$y + (y - (\text{Nmic} \,//\, 2) \times \text{pitch}) = \text{Nsamp} \times \text{dist per sample}$$

  Rearranging, we get:

$$y_{\max} = \frac{\text{Nsamp} \times \text{dist per sample} + (\text{Nmic} \,//\, 2) \times \text{pitch}}{2}$$

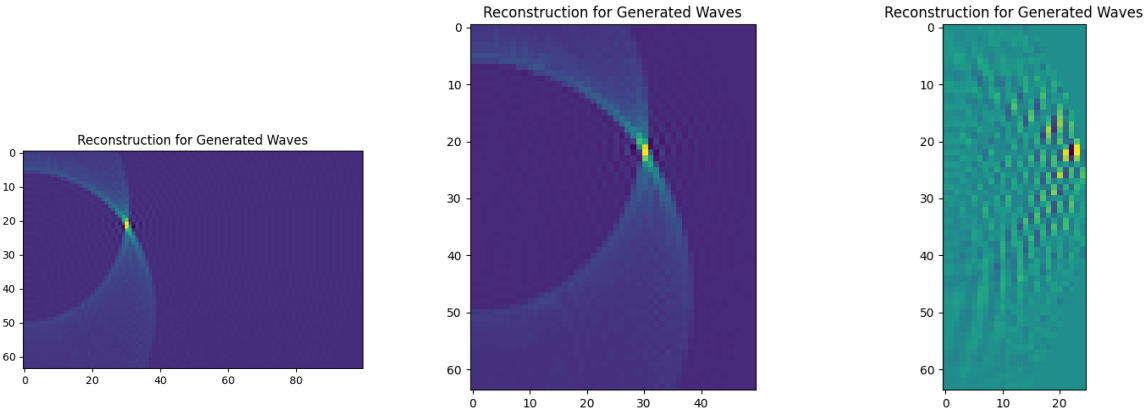
- What happens if C is different - if C is decreased it looks like the image becomes sharper. Can you explain why intuitively?

  The time difference between consecutive samples is given by:

$$\Delta t = \frac{\text{dist per sample}}{C}$$

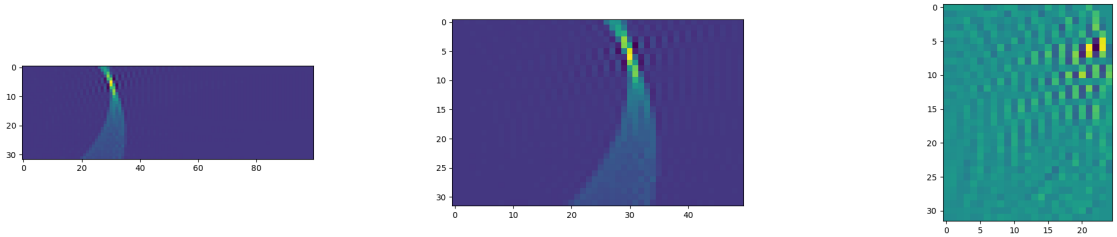
  The difference in the consecutive time samples is nothing but the difference in their distances/C, and since they are consecutive distance samples we get time per sample as distance per samp divided by C. Hence, as C decreases, the step change in time increases. This means that the sampling rate of the sin C signal is low, therefore we obtain a sharper image.

- What happens if Nmics is increased or decreased? Do the experiments with Nmics = [8, 32, 64] and Nsamp = [50, 100, 200] (all combinations). Attach the resulting images. Nmics=64



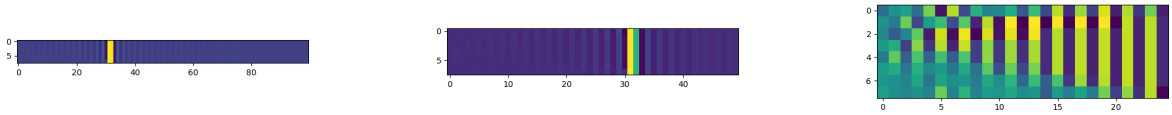
(a) Nsamp=200 (b) Nsamp=100 (c) Nsamp=50

Figure 2: Heatmaps for Nmics=64 with varying Nsamp values



(a) Nsamp=200 (b) Nsamp=100 (c) Nsamp=50

Figure 3: Heatmaps for Nmics=32 with varying Nsamp values



(a) Nsamp=200 (b) Nsamp=100 (c) Nsamp=50

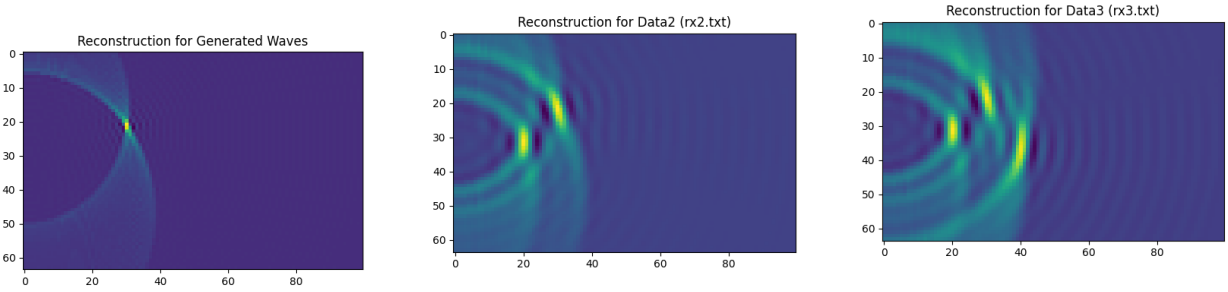
Figure 4: Heatmaps for Nmics=8 with varying Nsamp values

3 Delay and Sum Algorithm

Delays are caused due to difference in the distances travelled by the sound wave. If the delay of each of the mic signals by exactly the amount that we expect at each point on a grid, On aligning the signals in a particular way, They will add up to a maximum value at the point which is the position of the obstacle.

To reconstruct the image from the earlier generated data and the given files `rx2.txt` and `rx3.txt` we implement the DSA algorithm and regenerate the matrix and plot the heatmaps for the same and its evident from the heatmaps that that the pixel with high intensity is likely the position of the obstacle.

The reconstructed images obtained from the earlier generated data, `rx2.txt` and `rx3.txt` is given below:



(a) (b) (c)

Figure 5: Reconstructed images

4 Note :

The output of the code is a plot of the shifted sinc signals saved as `shifted_sinc.png` and a figure with all the heatmaps together saved as `heatmap.png`. When we change the `Nmics` to 32 and 8, its important to comment out the part where the `reconstruct` function is called since the files `rx2.txt` and `rx3.txt` are for 64 mics.