Blekinge Institute of Technology

Sound Source Localization by Using Two Microphones

This thesis is presented as part of Degree of Bachelor of Science in Electrical Engineering with emphasis on Telecommunication

Author: Gulay Yilmaz
Supervisor: Dr. Nedelko Grbic
Examiner: Dr. Sven Johansson

June 18, 2014

Contents

1 Abstract

This thesis work presents the way of locating the sound source by using two microphone. The idea to approach the goal is based on the Time difference of Arrival Estimation (TDOA). There are several ways to the TDOA such as the generalized cross-correlation (GCC) and Steered Response Power (SRP).The most common technique used in TDOA estimation is the generalized cross-correlation (GCC). But Steered Response Power-PHAT (SRP-PHAT) together with the Windowed Discrete Fourier Transform(WDFT) are mainly focused on this thesis work.

2 Introduction

Nowadays finding the direction of a source of sound has many applications. Some of these applications are all kind of intelligent environment, teleconferencing, robot navigation, noise cancellation, automabile speech enhancement. There are may other examples since we are dependent to technology in every part of the life and the interaction between human and machines are getting more common and this interaction is based on locating and tracking [1], [2], [3], [4] . As an example automatic speech recognition can be done in a better way, if the speaker position is known [5] . For a instance, in a meeting or conference environment, it is very useful that to detect and locate all voices and create a beam form to capture the independent channels for each speaker [6] .

Finding the sound source direction depends on the TDOA (Time Difference of Arrival). There are several different techniques to achieve TDOA and these techniques can be separated into two different part. One stage and two stage algorithms. Maximum likelihood estimation, the least square error, linear intersection method are the examples of the two stage algorithm. One stage algorithms are more robust in real-time implementations. The most common method of one stage algorithm is steered response power (SRP) technique. Use of phase transform with steered response power improves the performance [7] . In order to find the sound source direction, steered beamformer power needs to be maximized among the predefined location space.

In this report I have also explained the most common methods for the TDOA estimation. In the fallowing sections you will find theory about the generalized cross-correlation, the steered response power, PHAT (The phase transform) in both methods. In the implementation part, I have apply the steered response power with the phase transform (SRP-PHAT).

3 Background

Algorithms to find the sound source direction is based on the Time Difference of Arrival estimation. TDOA estimation is based on steered response power the phase transform (SRP-PHAT) beamformer. In this project the most important elements are difference estimation and direction search. This report explain SRP-PHAT theory and implementation in MATLAB.

4 Methods To Find The Direction Of Sound Source

There are two effective types of algorithms for finding the direction of sound source. They are one stage and two stage algorithms [8] , [9] , [10]. Maximum likelihood estimation, the least square error, linear intersection method are the examples of the two stage algorithm and they include two steps algorithmic process to be achieved. First system produces TDOA of sound between the pair of acoustic microphones then as a second step, time delay and the position of the microphones generate the hyperbolic curve. As a example of one stage algorithm, beamforming is the most common method. Beamforming is used to add the output of the microphones after the delaying process. In beamforming process, system is scanned or steered over the predefined region to find out the possible sound source position. The sound source is located in where the system gets the maximum power of beamforming. This process is also know as Steered Response Power. Direction of the sound source can be found by estimating the TDOA and SRP method is one of the most powerful way of finding TDOA.

5 Time Difference of Arrival

Sound direction estimation is based on finding the TDOA between two microphones. Reliable estimates requires long segments of data. Applications

which needs short data segments are infected by reverberation easily. Therefore, performance of pairwise TDE(Time Delay Estimation) based techniques degrades greatly under high noise conditions [11] .
The generalized cross-correlation (GCC) is the most common technique which is used in TDOA estimation. Because of noise and reverberation in the environment, to be able to improve the performance of GCC, weighting functions is necessary to be used. There are many kind of weighting function such as maximum likelihood (ML), smoothed coherence transform (SCOT), the phase transform (PHAT), the eckart filter, and the roth processor [12].
Between all those weighting functions ML and PHAT has the best performances in the noise-only case and reverberation case. Even though ML weighting is good and has high performance compared to the others, in reverberation and noisy environment it does not work efficiently. On the other side, the PHAT weighting is more robust than ML weighting function under high reverberation [13] . PHAT is indeed optimal if we compare with ML when the noise is low; PHAT is robust to reverberation, because its performance is independent of the amount of environment reverberation [14] .

6 Generalized Cross Correlation GCC

We have two microphones in the system and the signal in one of the microphone is defined as;

$$x_1(t) = s(t) * h_1(\bar{ds}, t) + n_1(t) \tag{1}$$

Where $x_1(t)$ is the signal that we have in microphone(1), $s(t)$ is the source signal, $h_1(\bar{ds}, t)$ is the impulse response, $\bar{ds}$ is the source position and $n_1(t)$ is noise.
In the other microphone we have signal as it defined below;

$$x_2(t) = s(t - \tau_{12}) * h_2(\bar{ds}, t) + n_2(t) \tag{2}$$

Where τ_{12} is a time delay, is to show that there is time differences between the signals in two microphones. We have TDOA where cross correlation between these two have peak point.Cross correlation of these two signals $x_1(t)$ and $x_2(t)$ is;

$$c_{12}(\tau) = \int_{-\infty}^{\infty} x_1(t)x_2(t+\tau)\,dt \tag{3}$$

If we take the Fourier transform of the cross correlation, we get the cross power spectrum;

$$C_{12}(\omega) = \int_{-\infty}^{\infty} c_{12}(t)e^{j\omega\tau}\,d\tau \tag{4}$$

Then we substitute equation 3 in equation 4 and we apply the convolution property of Fourier transform, we get;

$$C_{12}(\omega) = X_1(\omega)X_2^*(\omega) \tag{5}$$

where $X_1(\omega)$ and $X_2(\omega)$ indicate the Fourier transform of signals $x_1(t)$, $x_2(t)$ and '$X_2^*(\omega)$' is to show the complex conjugate of $X_2(\omega)$.
Inverse Fourier transform of equation 5 gives us the cross correlation function in terms of Fourier transform of the signals in microphones;

$$c_{12}(\tau) = \frac{1}{2\pi}\int_{-\infty}^{\infty} X_1(\omega)X_2^*(\omega)e^{j\omega\tau}\,d\omega \tag{6}$$

The generalized cross correlation is the cross correlation of two filtered version of signals $x_1(t)$, $x_2(t)$;

$$R_{12}(\tau) = \frac{1}{2\pi}\int_{-\infty}^{\infty} (W_1(\omega)X_1(\omega))(W_2(\omega)X_2^*(\omega))^* e^{j\omega\tau}\,d\omega \tag{7}$$

Where $W_1(\omega)$ and $W_2(\omega)$ is the Fourier Transform of $x_1(t)$ and $x_2(t)$;

Then weighting function is $\psi_{12}(\omega)$;

$$\psi_{12}(\omega) = W_1(\omega)W_2(\omega)^* \tag{8}$$

When we substitute this weighting function in equation 7, we get;

$$R_{12}(\tau) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \psi_{12}(\omega)X_1(\omega)X_2^*(\omega)^* e^{j\omega\tau}\, d\omega \tag{9}$$

Now we have GCC function, to be able to find the TDOA ,$\hat{\tau}_{12}$, between these two signal, we need to check where the GCC function has the maximum peak;

$$\hat{\tau}_{12} = \arg\max_{\tau} R_{12}(\tau) \tag{10}$$

6.1 *PHAT - The Phase Transform*

"The phase transform pre-whitens the signals before calculating the the cross-correlations to get a sharp peak. The time delay information is present in the phases of the various frequencies consist of the time delay information and these informations are not imposed by the transform. Because the transform tries to improve the true delay and suppress all spurious delays, it turns out to be very effective in rooms with reverberation and noise and the strongest peak can be picked out as the true delay. One disadvantage of the phase transform is the enhancement affects of frequencies that have low power when compared to the noise power. This can cause the estimates to be corrupted by the effect of uncorrelated noise" [15]. It has been shown that the phase transform (PHAT) weighting function is robust in realistic environments [16] , [17] even though it is sub-optimal [18] to the maximum likelihood (ML) weighting function which was studied in [12] , [13] under

reverberant-free conditions. PHAT is defined as follows;

$$\psi_{12}(\omega) = \frac{1}{\mid X_1(\omega){X_2}^*(\omega) \mid} \tag{11}$$

7 Steered Response Power-PHAT

Steered Beamforming

Beamforming is a signal processing technique used in sensor arrays for directional signal transmission or reception. This is achieved by combining elements in the array in such a way that signals at particular angles experience constructive interference while others experience destructive interference. The property of beamformers to enhance signals from a particular direction and attenuate signals from other directions can be used to perform TDOA estimation. A beamformer can be constructed for each direction of interest and the power of the array output can be computed.
Using a beamformer to find out the direction of the sound source is a simple idea. When applied to source localization, the beamformer output is maximized when the array is focused on the target location. The aim is to scan the beamformer over a set of candidate source locations, and then choose the source location as that which gives the maximum beamformer output power [15] .
In the system that we are using to find the direction of the sound source we have two microphones, together those microphones have the capability of focusing on signals generated from a specific location or direction. Such capability is referred to as a beamformer. The beamformer can be used to steer over a region containing the sound source location. The output of it is known as the steered response. When the point of focus matches the true source location, the steered response power (SRP) will peak [19] . The steered response beamformer, when used together with a phase transform filter, defines a one-stage method called steered response power using the phase transform, or SRP-PHAT. This method has been shown to be more robust under high noise and reverberation than the two-stage ones [17] . In a mathematical way we can express the beamforming as follows;
Consider we have two microphones and a signal source at different locations. We have the source signal $s(t)$ at location $\overrightarrow{r_s}$ and microphone at location $\overrightarrow{r_m}$.

$h(\overrightarrow{r_m}, \overrightarrow{r_s}, t)$ is the impulse response and $v(\overrightarrow{r_s}, t)$ is microphone's response. The microphone signal $x_m(t)$ can be expressed as follows;

$$x_m(t) = s(t) * h(\overrightarrow{r_m}, \overrightarrow{r_s}, t) * v(\overrightarrow{r_s}, t) + n_m(t) \tag{12}$$

Where m is the microphone index, $*$ is the convolution sign)
From equation 12 we can say that noise is not correlated to the source signal. $h(\overrightarrow{r_m}, \overrightarrow{r_s}, t)$ and $v(\overrightarrow{r_s}, t)$ are the convolution of the impulse response from the source output to the microphone output. Since microphone is in fixed position forever in our system, we can express the impulse function by $h(\overrightarrow{r_s}, t)$ then signal in microphone becomes ;

$$x_m(t) = s(t) * h(\overrightarrow{r_s}, t) + n_m(t) \tag{13}$$

Then we delay the microphone signal $x_m(t)$ with the appropriate steering delay and we can have weighted delay and sum beamformer in the microphone. Then we sum all these signals together. Steering delay can be written as below equation 14 , where τ_0 is constant delay.

$$\Delta_m = \tau_m - \tau_0 \tag{14}$$

$$y(t; \Delta_1, ..., \Delta_M) \equiv \sum_{m=1}^{M} x_m(t - \Delta_m) \tag{15}$$

where $\Delta_1, ..., \Delta_M$ are the M steering delays, which focus or steer the array to the source's spatial location or direction and $x_m(\cdot)$ is the signal received at the m^{th} microphone.

The delay-and-sum beamformer output $y(t; \Delta_1, ..., \Delta_m)$ in equation 15, can now be expressed in terms of the microphone signal model $x_m(t)$ of equation 13 and the steering delays δ_m from equation 14 ;

$$y(t;\Delta_1,....,\Delta_M) \equiv s(t) * \sum_{m=1}^{m=M} h(\overrightarrow{r_s}, t-(\tau_m-\tau_0)) + \sum_{m=1}^{m=M} n_m(t-(\tau_m-\tau_0)) \quad (16)$$

Then we are using Windowed Discrete Fourier Transform (WDFT) to filter the signal in microphones. We use Hamming window to separate the noise from the microphone signal and we get the below equation;

$$y(t;\Delta_1,....,\Delta_M) \equiv s(t) * \sum_{m=1}^{m=M} h(\overrightarrow{r_s}, t-(\tau_m-\tau_0)) \quad (17)$$

In equation 17 we have output sum beamformer in time domain which has M element. To be able to get the beamformer in frequency domain we need the following equation;

$$Y(t;\Delta_1,....,\Delta_M) \equiv \sum_{m=1}^{m=M} G_m(\omega) X_m(\omega) e^{-j\omega\Delta_m} \quad (18)$$

where $X_m(\omega)$ is the Fourier transform of the microphone signal $x_m(t)$ and $G_m(\omega)$ is the Fourier transform of $h(\overrightarrow{r_s}, t-(\tau_m-\tau_0))$.

7.1 *Steered Response Power*

To be able steer the beam in specific position or direction, steering delay M is used. To obtain the steered response we sweep the focus of the beamformer. The time aligned signals in microphones are added up and the power of the steered response reaches the maximum value when the beamformer focus on the position where the sound source is located. When we steer the beamformer over all region we get the steered response power (SRP). We can express this with using following equation;

$$P(\Delta_1, ..., \Delta_M) = \int_{-\infty}^{\infty} Y(\omega, \Delta_1..., \Delta_M) Y^*(\omega, \Delta_1..., \Delta_M) d\omega \tag{19}$$

In above equation $Y^*(\omega, \Delta_1..., \Delta_M)$ is the complex conjugate of $Y(\omega, \Delta_1..., \Delta_M)$. We substitute the equation 18 into the equation 19 and we get the below equation;

$$P(\Delta_1, ..., \Delta_M) = \int_{-\infty}^{\infty} \left(\sum_{k=1}^{k=M} G_k(\omega) X_k(\omega) e^{-j\omega\Delta_k} \right) \left(\sum_{l=1}^{l=M} G_l^*(\omega) X_l^*(\omega) e^{j\omega\Delta_l} \right) d\omega \tag{20}$$

If we rearrange the equation 20;

$$P(\Delta_1, ..., \Delta_M) = \int_{-\infty}^{\infty} \sum_{k=1}^{k=M} \sum_{l=1}^{l=M} \left(G_k(\omega) G_l^*(\omega) \right) \left(X_k(\omega) X_l^*(\omega) \right) e^{j\omega(\Delta_l - \Delta_k)} d\omega \tag{21}$$

From this part we can say that in equation 21 expression $(\Delta_l - \Delta_k)$ can be written as $(\tau_l - \tau_k)$ and we substitute this into equation 21, we get;

$$P(\Delta_1, ..., \Delta_M) = \int_{-\infty}^{\infty} \sum_{k=1}^{k=M} \sum_{l=1}^{l=M} \left(G_k(\omega) G_l^*(\omega) \right) \left(X_k(\omega) X_l^*(\omega) \right) e^{j\omega(\tau_l - \tau_k)} d\omega \tag{22}$$

Since microphone signals have finite energy because of its integral convergence we can take the integral inside the summation;

$$P(\Delta_1, ..., \Delta_M) = \sum_{k=1}^{k=M} \sum_{l=1}^{l=M} \int_{-\infty}^{\infty} \left(G_k(\omega) G_l^*(\omega)\right) \left(X_k(\omega) X_l^*(\omega)\right) e^{j\omega(\tau_l - \tau_k)} d\omega \quad (23)$$

Weighting function is as fallowing;

$$\psi_{kl}(\omega) = \left(G_k(\omega) G_l^*(\omega)\right) \quad (24)$$

We can again substitute the $(\tau_l - \tau_k)$ with the τ_{lk} and we combine the equation 23 , equation 24 , now we have the final expression ;

$$P(\Delta_1, ..., \Delta_M) = \sum_{k=1}^{k=M} \sum_{l=1}^{l=M} \int_{-\infty}^{\infty} \psi_{kl}(\omega) \left(X_k(\omega) X_l^*(\omega)\right) e^{j\omega(\tau_{lk})} d\omega \quad (25)$$

As we can see in the final expression the Steered Response Power (SRP) is the summation of the Generalized Cross Correlation (GCC) of pairs of microphones.

7.2 *PHAT-The Phase Transform*

The intended purpose of the weighting function in SRP-PHAT is emphasizing the actual sound source over the undesired signals. The phase transform makes microphone signal spectrum flatten by using whitening technique. Use of PHAT improves the performance of SRP. If we compare the effect of PHAT on SRP with other weighting function, we can see that PHAT has better performance with low noise and reverberant environments [20] , [21] . SRP-PHAT has peak where actual sound source is located.

8 Implementation of SRP-PHAT

In this part I will explain step by step how I implement the SRP-PHAT algorithm. I used MATLAB environment to implement the algorithm. As a first step I will explain WDFT. As it shown in the 29 , we need microphone signal in frequency domain which means that we need to take Fourier transform of the signal and we need to filter them to be able get rid of the noise. By the help of using $WDFT$ provide us microphone signals which are filtered and in frequency domain.

8.1 *Windowed Discrete Fourier Transform*

Window function is a mathematical function that is zero-valued outside of some chosen interval. When another function or a signal is multiplied by a window function, the product is also zero-valued outside the interval where both function are overlapping. Applications of window functions include spectral analysis, filter design, and beamforming. Rectangular window, Hamming window, Hann window are some examples of the windowing functions. Rectangular window is the simplest example of the the window functions, it is constant inside the interval and zero elsewhere as it seen in figure 2. In this project I used Hamming window which is in the figure 1, and it is defined as in equation 26 .

$$w(n) = \begin{cases} 0.54 - 0.46\cos(\frac{2n}{L-1}) & \text{if } 0 \leq n \leq L-1 \\ 0 & \text{if } otherwise \end{cases} \tag{26}$$

When we are using window function with discrete Fourier transform (equation 29) we have an input signal in the system and we are dividing this signal by the size of window function. But this is not regular division because we have overlapping between the windows. Overlapping scheme can be seen in figure 3. That is why separating signal into parts is done by according to overlapping percentage. And the reason why we have overlapping in our system is because overlapping helps to recover the lost data and reduce the

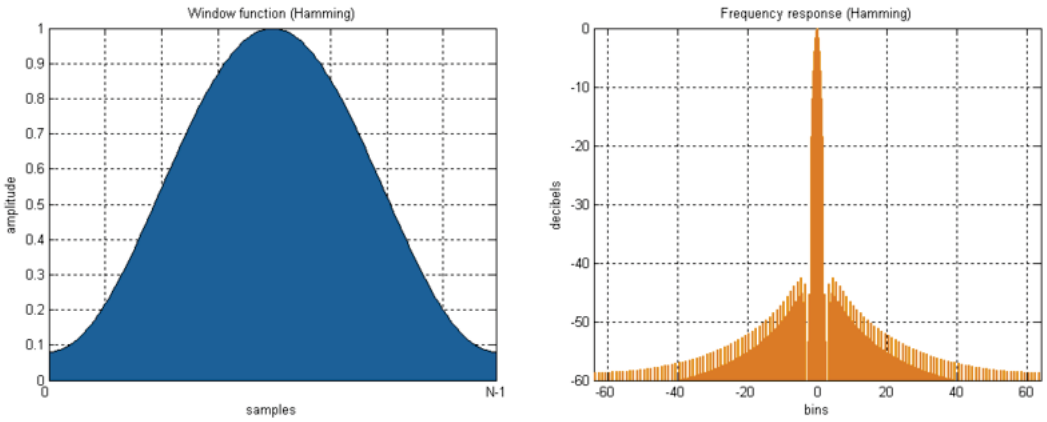

Figure 1: Hamming window

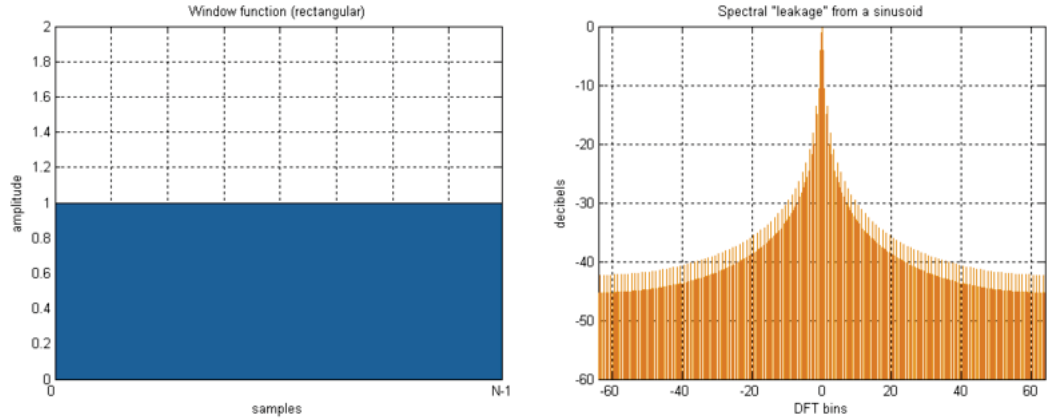

Figure 2: Rectangular window

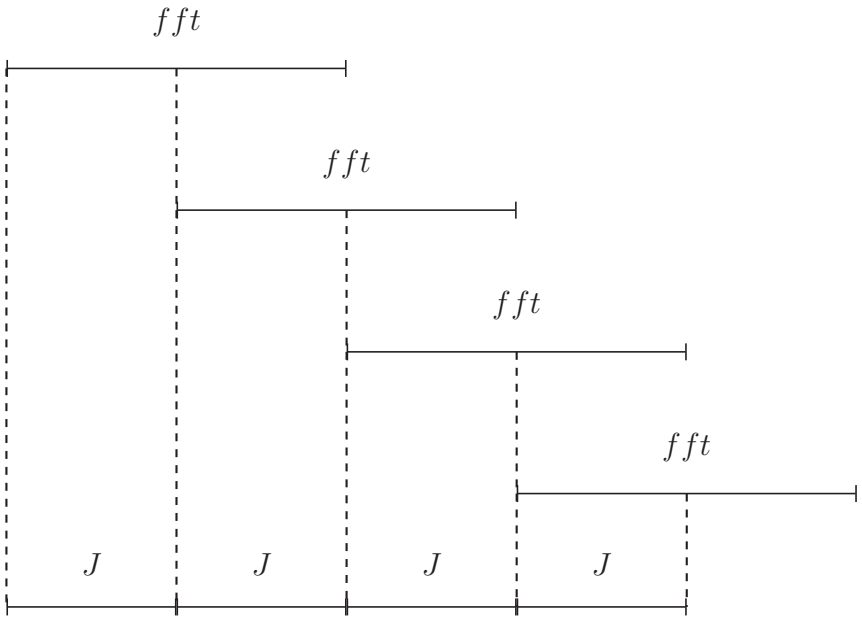

Figure 3: Overlapping segments

measurement time. This processing reduces the total measurement time by recovering a portion of each previous frame that otherwise is lost due to the effect of the windowing function.

Discrete Fourier transform is used in each window, it starts again with the displacement of J samples. J represents the number of samples that the algorithm uses for displacement of $WDFT$ in each time and it can be found out by using equation 27 below.

$$J = \frac{N}{K} \tag{27}$$

N is the window size which is used in the $WDFT$ and K is the inverse of overlapping percentage (OP). For example, 50% overlapping means that K is equal to 2. Overlapping percentage can be calculated as shown in equation 28 . When we have K is equal to 2, it corresponds to overlapping 50%, if we have K is equal to 4, it means that overlapping is 75% and so on.

$$OP = \frac{J}{N} \tag{28}$$

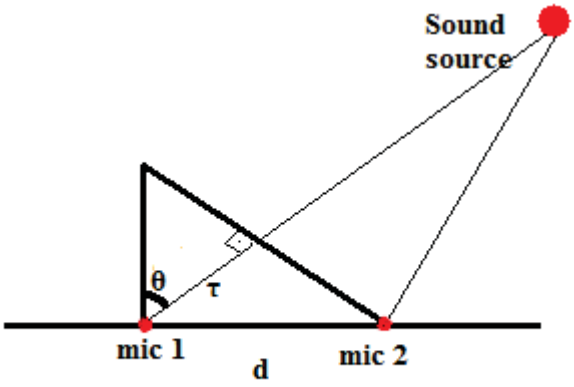

Figure 4: Microphone delay

By making overlapping percentage larger, more information is shared between blocks and this results more redundancy.

$$X[g, \omega_i] = \sum_{n=0}^{N-1} S[n]x[Mg + n]e^{-j\omega_i n} \tag{29}$$

As a result of implementing $WDFT$ in equation 29 we have filter bank matrix of original input signal. Filter bank has N (window size chosen) column and g rows, where g is the number of windows used for whole input signal. Each row corresponds to the Fourier transform of single window of input signal. Each column corresponds to samples of a single subband.

There are two microphones in the system and the distance between two microphone is d as it shown in figure 4. Because of this distance and the position of the microphones, there is time delay τ between two microphone signals. My approach is first finding this delay between microphone signals and then finding the direction of sound source by using this time delay.

Signal of the first microphone is $x_1(t)$ and signal of the second microphone is $x_2(t)$. I have implement the Windowed Discrete Fourier Transform function to both microphone signals and as a result I have $X_1(\omega)$, $X_2(\omega)$;

$$X_1[m,\omega_i] = \sum_{n=0}^{N-1} S[n]x_1[Mm+n]e^{-j\omega_i n} \tag{30}$$

$$X_2[m,\omega_i] = \sum_{n=0}^{N-1} S[n]x_2[Mm+n]e^{-j\omega_i n} \tag{31}$$

Now we have two matrices $X_1(\omega)$ and $X_2(\omega)$ which have g (which is the number of used window function) rows, N (which is window size) columns. As a next step we need to implement SRP-PHAT algorithm to these signals and I will explain it in following section.

8.2 *SRP-PHAT*

As I mentioned in the section 7 sound source is in the location where the SRP-PHAT has the maximum value. To be able to find out the maximum value, first we need to define interval range for steering the beamformer. This range can be found by following expression;

$$\tau_{max} = \frac{distance}{v} \tag{32}$$

where *distance* indicate the space between microphone and v indicates the speed of sound which is $342m/s$. So that steering range should be between $-\tau_{max}$ and τ_{max}. In my code I am checking 1000 positions between these intervals.

$$\hat{\tau} = \arg\max_{\tau} \left(\frac{X_1(\omega){X_2}^*(\omega)}{|X_1(\omega){X_2}^*(\omega)|} e^{-j\omega\tau} \right) \tag{33}$$

In above equation 33 $\hat{\tau}$ represents the maximum value of the SRP-PHAT. I am looking for the τ value which makes the $\hat{\tau}$ maximum in equation 33 .

Now it is time for implementing the SRP-PHAT and check for the maximum. $X_1(\omega)$ and $X_2(\omega)$ are the WDFT of the microphone signals which we have already calculated, each of them are gXN matrices.

1. Take the entire first rows of the both matrices and implement the equation 33 by using these rows.

2. We have steering range from $-\tau_{max}$ to τ_{max} and these range is divided into 1000 even position between them.

3. Use each position and substitute them as the τ value of equation 33 . Store the result in the first row of a new matrix which I called T.

4. Find the mean value of that row of the T matrix and store the result in matrix mn.

5. Now I check for the maximum value of the mn matrix. The important thing to be noticed here is I need the position of the maximum value, not the maximum value itself. Once the position of the maximum value is found in mn matrix and store the value in another matrix which I called T_{max}.

So far we have only worked with the first rows of the $X_1(\omega)$ and $X_2(\omega)$, next we do the same steps from $1-5$ for the rest of row of the $X_1(\omega)$ and $X_2(\omega)$.

In the end we have T_{max} which has the position of the maximum values of each window. To find out the delay of whole system I take the mean of T_{max} which suppose to be the delay between two microphones that I am looking for.

8.3 *Direction of Sound Source*

Now I have the delay between two microphones. And I can use it to find the direction of the sound source. As it seen in the figure 4 I can use the triangle and with the help of simple trigonometric functions, I can find out direction of sound source θ, [22] .

$$\tau = \frac{d}{v} \cdot \sin(\theta) \tag{34}$$

so that;

$$\theta = \arcsin\left(\frac{v}{d} \cdot \tau\right) \tag{35}$$

Where v is speed of sound, d is the distance between microphones and τ is the arrival delay between two microphones.

9 Results of Implementation

For the implementation of this algorithm I used my personal laptop. I implemented the SRP-PHAT algorithm by using MATLAB environment. In the beginning I used random signal to build and test my algorithm. I created a random signal with desired sampling frequency and length (in my case I chose to 48 kHz as sampling frequency). I delayed this random signal to be able have a second signal in my system. Original random signal and the delayed signal represent the signals in each microphones. Expected time delay can be found by the equation 36; where m is the number of sample we want to delay, τ is the expected time delay, Fs is the sampling frequency;

$$\tau = m/Fs \tag{36}$$

I run my code and once I got the approximately same time delay as expected, I started to test the code by using real time signals.

I record the sound source signal by using the Audacity programme. For recording the signals, I used my personal laptop which has two microphones in line. The distance between these two microphones is $0.08m$. In the Audacity programme, while recording the sound signals, there is a option

which provides you to pick different sampling frequencies. As an example I set the sampling frequency to 48 kHz for the beginning and I recorded the sound which is in perpendicular direction to the microphone line. I used this recorded sound to test the algorithm. I loaded the recorded signal in MATLAB and I stored it a matrix. The length of the sound signal is corresponding to the row number this matrix and it had also 2 column. One column was for the sound signal which is recorded from one microphone and the other column was for the other microphone. To be able to find the expected time delay; I separated the matrix into two column vectors, I subtracted column vectors from each other as element wise operation and I calculated the mean value of subtraction result.

When I run my code by using the real sound signal, at first I had a problem to find the correct time delay. I realized that I skipped to consider the equation 37. Taking the all subbands of the $WDFT$ was causing the problem. I had limitations of using the frequency, frequency range should be as in seen in equation 37 where F is the frequency, v is the speed of sound and d is the distance between microphones.

$$F < \frac{v}{(d \cdot 2)} \tag{37}$$

So above frequency limitations led me to use only certain number of subbands. The number of subband can be found as in equation 39. First I calculated the frequency width (FW) as in equation 38, where F_s is the sampling frequency and N is the window size. Ones I got the FW I could find the number of subbands that I could use in the algorithm.

$$FW = \frac{F_s}{N} \tag{38}$$

$$number of subband = \frac{F}{FW} \tag{39}$$

After I found the number of subband that I can use, I run my code to see the results. I tried my code with different frequencies and different window sizes and with the different number of subband to show that it will give the

best result when we use the exact number of subband that is calculated by using equations 39 and 38. Now I will show the results of them.

the MATLAB figures can be seen in figure 5 ($F_s = 24kHz, N = 256$), figure 6 ($F_s = 24kHz, N = 512$), figure 7 ($F_s = 32kHz, N = 256$), figure 8 ($F_s = 32kHz, N = 512$), figure 9 ($F_s = 48kHz, N = 256$) and figure 10 ($F_s = 48kHz, N = 512$) with the different selected F_s and N. In the figures blue dots are showing the expected time delay (expected time delay is found by subtracting the two microphone signals(which are recorded) from each other) and the green boxes are showing the time delays that I got as a result from the code. I have tried $24kHz$, $32kHz$, $48kHz$ as the sampling frequency and I have used 256 and 512 as the window size. I have used around ten different subband which are close to the the one calculated. I tried the code with different number of subbands but using same frequency and window size to find out where I can get the best result. As it seen in the figures using $24kHz$ and $32kHz$ sampling frequency didn't give correct result.
In the case that I used the $48kHz$ as sampling frequency and with both window sizes and source of the sound supposed to be in the perpendicular direction to the microphone line (which means that the angle between the normal vector of the microphone line and the sound source is is 0 degree), as the result; I got the degree 0.6159 when the window size is 256 and 0.6606 degree when the window size is 512. The result shows that the expected outcome and the output of the code matches very good. This shows that the code that I wrote is working optimal when we set the sampling frequency in $48kHz$ and use 27 subbands when $N = 512$ and 14 subbands when $N = 256$.

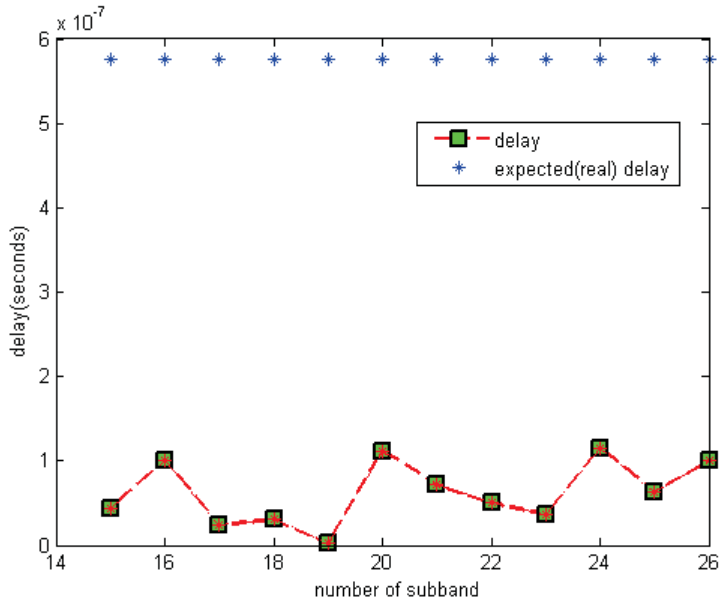

Figure 5: Fs=24kHz, N=256

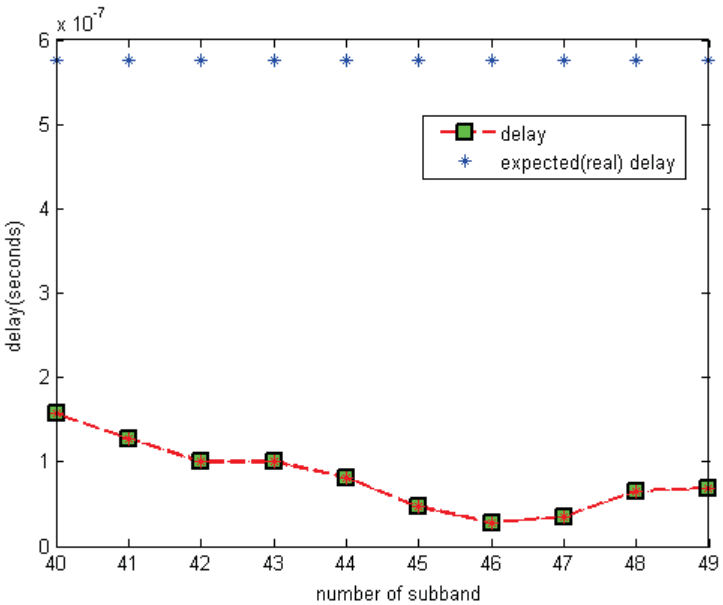

Figure 6: Fs=24kHz, N=512

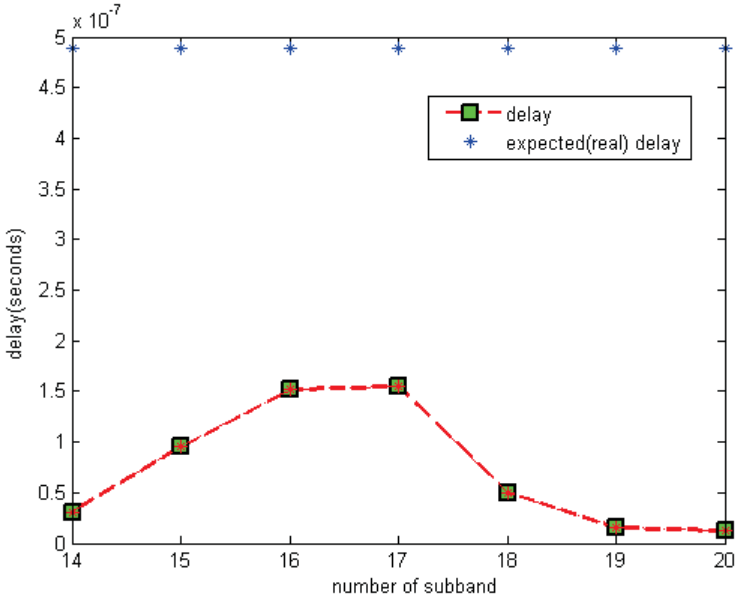

Figure 7: Fs=32kHz, N=256

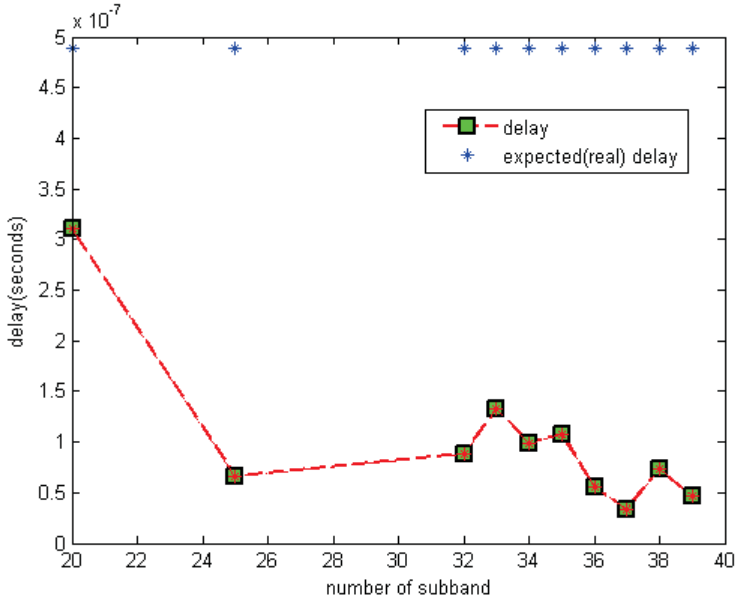

Figure 8: Fs=32kHz, N=512

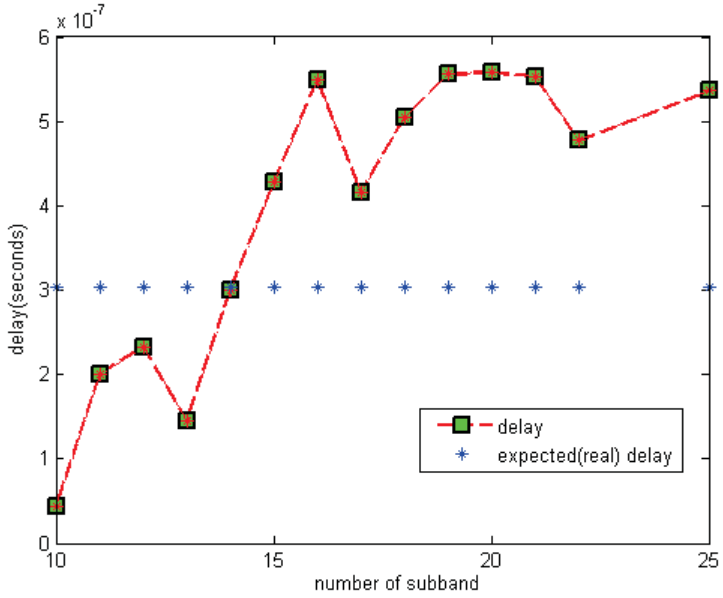

Figure 9: Fs=48kHz, N=256

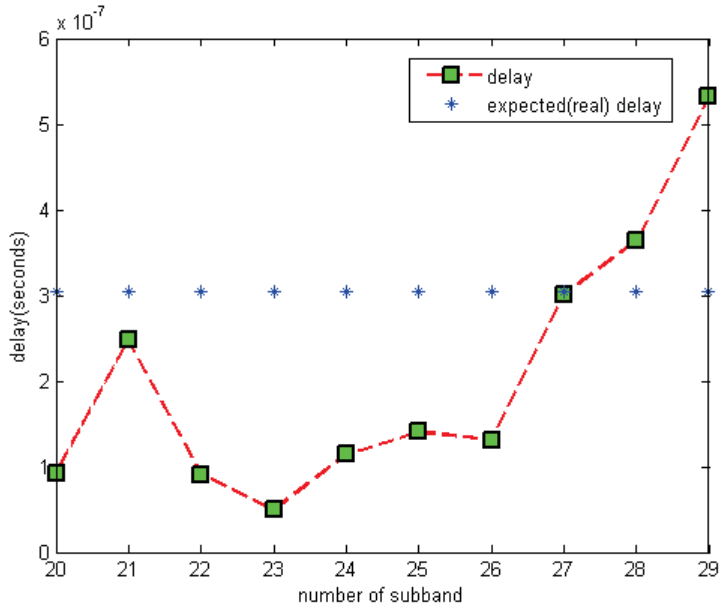

Figure 10: Fs=48kHz, N=512

10 Conclusion

This project shows the methods for finding the direction of sound source and it is mainly focus on SRP-PHAT technique. Implementation of finding the sound source direction is done by SRP-PHAT and using windowed discrete Fourier transform. It is giving the best results in $48kHz$, I could not get any good result for the another sampling frequencies or for the other window sizes. The code is not running efficiently. It is consuming a lot of time because of the big matrices and specially because of the loops that I used for matrices. Since it is known that Matlab is not the best way of running loops. For the future work algorithm could be made more time effective and the whole system could be improved and make it works more optimal, fast and more accurate.

References

[1] J. Dmochowski, J. Benesty, and S. Affes, "Fast steered response power source localization using inverse mapping of relative delays", 2008.

[2] J-M. Valin, F. Michaud, and J. Rouat, Robust localization and tracking of simultaneous moving sound sources using beamforming and particle filtering, Robot. Auton. Syst., vol. 55, pp. 216228, 2007.

[3] F. Michaud, C. Cote, D. Letourneau, Y. Brosseau, J.M. Valin, E. Beaudry, C. Raevsky, A. Ponchon, P. Moisan, P. Lepage, Y. Morin, F. Gagnon, P. Gigu‘ere, M.A. Roux, S. Caron, P. Frenette, and F. Kabanza, Spartacus attending the 2005 AAAI conference, Auton. Robots, vol. 22, no. 4, pp. 369383, 2007.

[4] Y. Tamai, S. Kagami, Y. Amemiya, Y. Sasaki, H. Mizoguchi, and T. Takano, Circular microphone array for robots audition, in Proceedings of IEEE Sensors, Oct. 2004, pp. 565570.

[5] T.B. Hughes, Hong-Seok Kim, J.H. DiBiase, and H.F. Silverman, Using a real-time, tracking microphone array as input to an HMM speech recognizer, Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 1, pp. 249252 vol.1, May 1998.

[6] Ajoy Kumar Dey, Susmita Saha, "Acoustic beamforming: Design and development of steered response power with phase transform (SRP-PHAT)", August, 2011

[7] Ramamurthy, Anand, "Experimental evaluation of modified phase transform for sound source detection", (2007). Masters Theses. Paper 478.

[8] T. Gustafsson, B. Rao and M. Triverdi, Source Localization in Reverberant Environments: Modeling and Statistical Analysis, IEEE Transactions on Speech and Audio Processing, pp. 791-803, 2003.

[9] P. Svaizer, M. Matassoni and M. Omologo, Acoustic Source Location in a Three-Dimensional Space Using Cross Power Spectrum Phase, IEEE International Conference on Acoustic, Speech and Signal Processing (ICASSP-97), Munich, Germany, pp. 231-234, 1997.

[10] H. F. Silverman, W. R. Patterson III, J. M. Sachar and Y. Yu, Performance of Real Time Source Location Estimators for a Large Aperture Microphone Array, IEEE Transaction of Speech , Audio Processing, pp. 593-606, July 2005.

[11] Hoang Tran Huy Do, Real-Time SRP-PHAT Source Localization Implementations on a Large-Aperture Microphone Array, Brown University, Providence, RI, Sep. 2009.

[12] C. H. Knapp and G. C. Carter. The generalized correlation method for estimation of time delay. IEEE Trans. Acoust., Speech, Signal Process., Aug. 1976.

[13] M. S. Brandstein. Time-delay estimation of reverberated speech exploiting harmonic structure. J. Acoust. Soc. Amer., 1999.

[14] Cha Zhang, Dinei Florencio and Zhengyou Zhang. "Why does PHAT work well in low noise, reverberative environments?", Microsoft Research,One Microsoft Way, Redmond, WA 98052, USA,chazhang,dinei,zhang@microsoft.com

[15] Krishnaraj Varma. Using a beamformer for source localization is a conceptually simple idea. The aim is to scan the beamformer over a set of candidate source locations, and then choose the source location as that which gives the maximum beamformer output power.

[16] H. F. Silverman, Y. Yu, J. M. Sachar, and W. R. Patterson. Performance of real-time source-location estimators for a large-aperture microphone array. IEEE Trans. Speech, Audio Process., 4(13):593-606, July 2005.

[17] J. H. DiBiase. A High-Accuracy, Low-Latency Technique for Talker Localization in Reverberant Environments Using Microphone Arrays. PhD thesis, Brown University, Providence, RI, May 2000.

[18] M. S. Brandstein and H. F. Silverman. A robust method for speech signal time-delay estimation in reverberant rooms. In Proc. IEEE Int. Conf. Acoust. Speech, Signal Process., Apr. 1997.

[19] D. H. Johnson and D. E. Dudgeon. Array Signal Processing: Concepts and Techniques. PTR Prentice Hall, 1993.

[20] J. Dmochowski, J. Benesty, and S. Affes, A Generalized Steered Response Power Method for Computationally Viable Source Localization, IEEE Transactions on Ausio, Vol.15, pp. 2510-2526, Nov. 2007.

[21] P.L. Chu, Super Directive Microphone Array for a Set Top Video Conferencing System, IEEE International conference of Acoustic, Speech, Signal Processing, vol. 1, pp. 235-238, May 1997.

[22] Mikael Swartling, Nedelko Grbic, "Calibration errors of uniform linear sensor arrays for DOA estimation: an analysis with SRP-PHAT." pp. 1071-1075, 2010.

[23] Anthony Badali, Jean-Marc Valin, Francois Michaud, Parham Aarabi, "Evaluating real-time audio localization algorithms for artificial audition in robotics".