

DOKUZ EYLÜL UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

SOUND LOCALIZATION
SIMULATOR SOFTWARE

by
Erkan UZUN

June, 2012
İZMİR

SOUND LOCALIZATION SIMULATOR SOFTWARE

**A Thesis Submitted to the
Graduate School of Natural and Applied Sciences of Dokuz Eylül University In
Partial Fulfillment of the Requirements for the Degree of Master of Science in
Computer Engineering, Computer Engineering Program**

**by
Erkan UZUN**

**June, 2012
İZMİR**

M.Sc THESIS EXAMINATION RESULT FORM

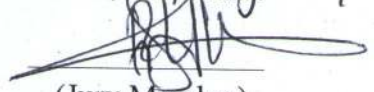
We have read the thesis entitled “**SOUND LOCALIZATION SIMULATOR SOFTWARE**” completed by **ERKAN UZUN** under supervision of **PROF. DR. ALP R. KUT** and we certify that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.



Prof. Dr. Alp R. Kut
Supervisor

Yrd. Doç. Dr. Canan Atay

(Jury Member)

Y. Doç. Dr. Beyat Yılmaz

(Jury Member)



Prof. Dr. Mustafa SABUNCU
Director
Graduate School of Natural and Applied Science

ACKNOWLEDGEMENTS

I would like to thank to my thesis advisor Prof. Dr. Alp R. Kut for encouraging me to complete my master, for his suggestions and patience.

I would like to also thank to all faculty members and research assistants at Dokuz Eylul University Computer Science Department for their help.

Special thanks to my family, my reason of life.

Erkan UZUN

SOUND LOCALIZATION SIMULATOR SOFTWARE

ABSTRACT

In this study, we aimed to simulate the sound, microphones and 2D localization in an interactive manner. We have used simple, educational, alternative and inexpensive methods. At the beginning, this working was limited to software implementation, but then it's understood that two ordinary microphones and stereo inputs of computers allow us to find the time delay of the arrival. Therefore, the hardware implementation is added to the study. Initially, mathematical approaches are examined to find the angle of the arrival of the sound. Error rates of the related formula were analyzed. Then, some properties of the sound, frequency, amplitude, speed, damping and sampling are explained. And finally, the software and the hardware implementations are explained. Computer programs are written in java language.

Keywords : Sound source localization, Time difference of arrival, TDOA, Sound simulation

SES KAYNAĞININ YER TESPİTİ İÇİN SİMİLASYON YAZILIMI

ÖZ

Bu çalışmada ses, mikrofon ve iki boyutta yer tespitinin interaktif bir şekilde simülasyonu hedeflenmiştir. Basit, eğitici, alternatif olabilecek ve pahalı olmayan yöntemler kullanılmaya çalışılmıştır. Başlangıçta çalışma yazılımla sınırlandırılmış ancak sonradan sadece iki mikrofon ve bilgisayarın stereo girişleri kullanılarak sesin mikrofonlara geliş süreleri arasındaki farkın hesaplanabileceği anlaşıncı donanımsal olarak gerçekleştirilmesi hedeflenmiştir. İlk olarak sesin geliş açısını bulmak için kullanılan matematiksel yaklaşımlar incelenmiş ve ilgili formüllerin hata oranları analiz edilmiştir. Daha sonra sesin bazı özellikleri, frekans, genlik, hız, sönümleme ve örnekleme konuları açıklanmıştır. Ve son olarakta yazılım ve donanım uygulamaları anlatılmıştır. Yazılım dili olarak Java kullanılmıştır.

Anahtar Sözcükler : Ses kaynağının yer tespiti, Sesin varış zamanları arasındaki fark, Ses simülasyonu

CONTENTS

	Page
M.Sc THESIS EXAMINATION RESULT FORM	ii
ACKNOWLEDGEMENTS	iii
ABSTRACT	iv
ÖZ	v
CHAPTER ONE – INTRODUCTION.....	1
CHAPTER TWO - PRELIMINARIES AND DEFINITIONS	3
CHAPTER THREE - FINDING TIME DELAY	5
3.1 Explanation of the Microphone System.....	5
3.2 Mathematical Proof for the Formula	6
3.3 Examination of the Error Rate of the Formula	8
3.4 Time Difference Calculation	12
3.5 Frequency Constraint	16
3.6 The Frequency That Can Be Determined	17
CHAPTER FOUR – SOFTWARE IMPLEMENTATION	19
CHAPTER FIVE - HARDWARE IMPLEMENTATION	21
5.1 Combined Microphones	21
5.2 Audio Signal Sampling	22
5.3 Getting Data From Sound Card	22
5.4 Finding Distance Difference	24
5.5 Pre-calculated Angle Values.....	27
5.6 Difference Signal And a Solution to Overcome Discrete Data	29
CHAPTER SIX - DISCUSSION AND CONCLUSION	33

REFERENCES 34

CHAPTER ONE

INTRODUCTION

The sound is an essential part of our everyday life. Humans, animals, machines and many other objects and events continuously produces sounds, which carry energy and information. Digital technology shows itself every area of our life. The sound is not an exception. Many studies have been done with sound, especially in music. One of the most known is MP3.

Sound source localization should be familiar to each of us. We and animals around us have this ability. We could decide where the sound came from. We do this with our brains and two ears. Computer systems have a central processing unit CPU and microphones. It seems they have similar tools to demonstrate this ability.

How can we decide where the sound source is? There are only a few acoustic cues for source location: Interaural Time Differences (ITD) or Time Difference Of Arrival (TDOA), Interaural Intensity Differences (IID) or Intensity Level Differences (ILD). The duplex theory of sound lateralization (Lord Rayleigh, 1907) says we use ITDs for low frequencies, IIDs for high frequencies. In our study only ITD method is used, but have future plans for the IID.

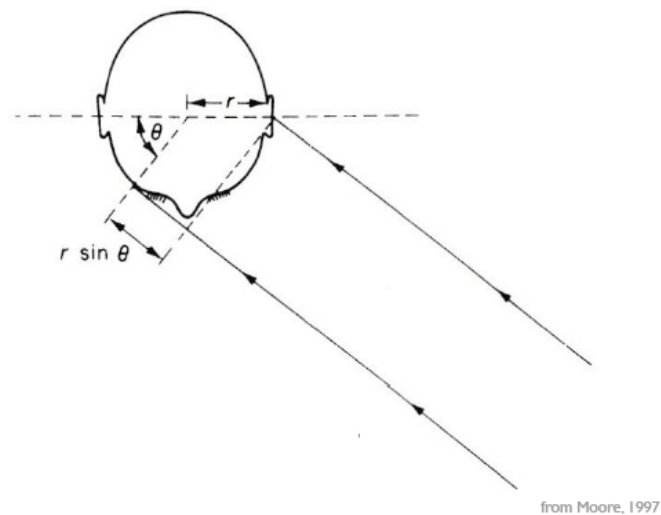


Figure 1.1 ITD method

In ITD methods, the key point is time differences. A normal head width is approximately 18 cm. (Distance between left and right ears). The velocity of the sound is 343 m/s (In 20 °C air at the sea level). The sound wave needs 29 μ s to travel 1 cm. The maximum time difference could be $18 \times 29 = 522 \mu$ s. The sound reaches to one ear at most half ms earlier or half ms later. In this 1 ms time interval we decide the angle of the sound source.

Throughout this research, we have attempted to understand and explain how it is possible to decide the direction of the sound in this short time. Two microphones are used to localize a sound source in both software and hardware. The study is limited to the two dimensional plane. One of a sample interactive simulator for sound waves can be found at the website of the university of Colorado.



Figure 1.2 A sample interactive simulation software

We realized a similar program, but specialized on TDOA. It allows you to measure time differences between microphones and calculates the direction of the sound source.

CHAPTER TWO

PRELIMINARIES AND DEFINITIONS

Sound is a mechanical wave that is an oscillation of pressure transmitted through a solid, liquid, or gas, composed of frequencies within the range of hearing (between about 20 Hz and 20,000 Hz.) and of a level sufficiently strong to be heard, or the sensation stimulated in organs of hearing by such vibrations.

Frequency (f) is the number of occurrences of a repeating event per unit time (cycle per second) expressed in hertz (Hz). The amount of time for one vibration is called the Period (T). The period is the reciprocal of the frequency. Human hearing is normally limited to frequencies between about 20 Hz and 20,000 Hz.

$$f = \frac{1}{T} \quad , \quad T = \frac{1}{f} \quad (1)$$

$$s(t) = A \sin 2\pi ft \quad (2)$$

A sound signal with the frequency f is expressed in Formula (2). The frequency changes when the coefficient is changed inside the mathematical function sine. Some sinusoidal functions with different frequencies are shown in the Figure 2.1. Note that all the signals have the same amplitude.

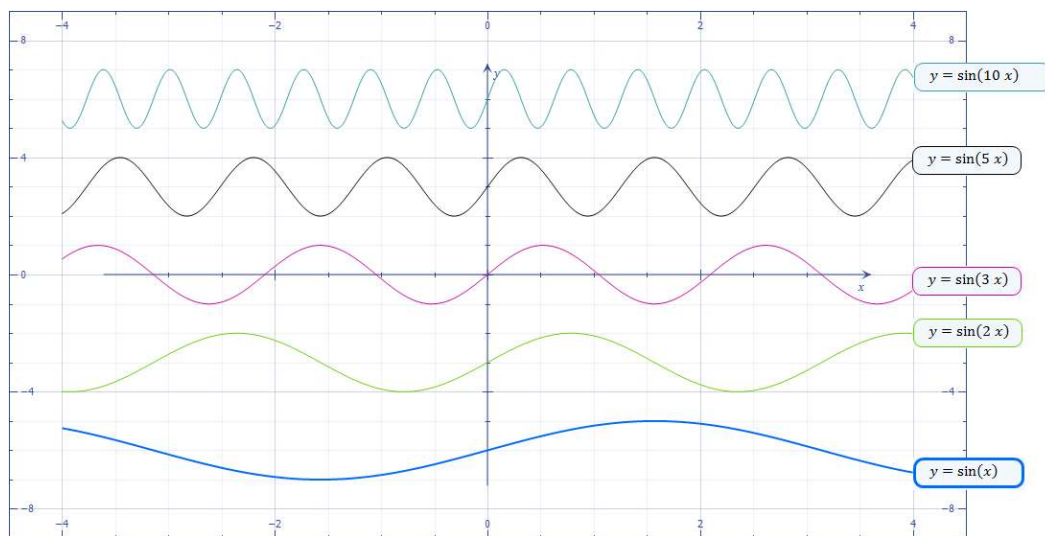


Figure 2.1 Sinusoidal functions with different frequencies

Amplitude is related to the amount of energy (intensity) in a sound wave and causes air pressure. The amplitude is shown by the height of a sound signal. Peak to peak amplitude (from $-A$ to A) makes $2A$ for a sinusoidal signal as in Formula (2). Some sinusoidal signals with different amplitudes are shown in the Figure 2.2. Note that all the signals have the same frequency.

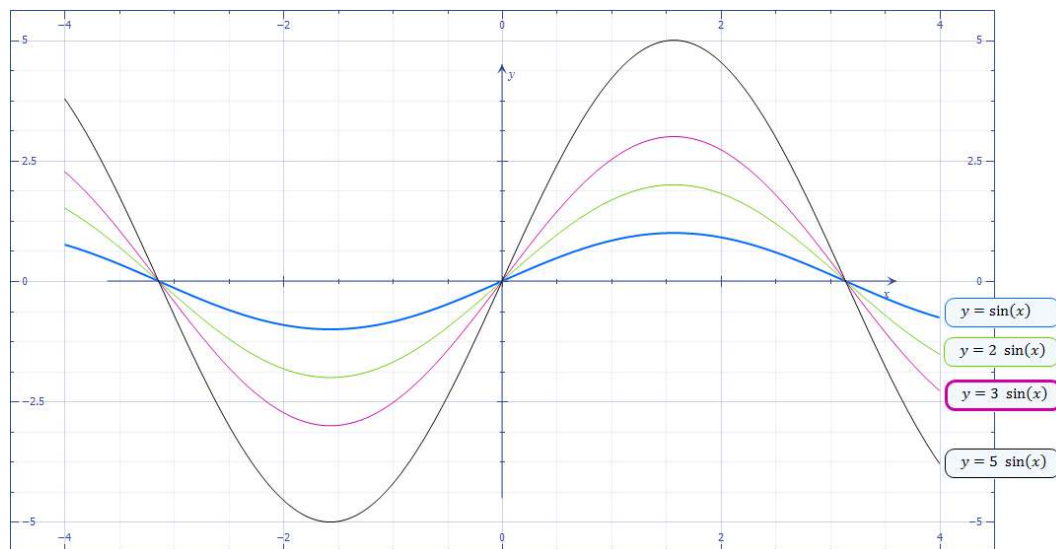


Figure 2.2 Sinusoidal functions with different amplitudes

CHAPTER THREE

FINDING TIME DELAY

3.1 Explanation of the Microphone System

The system with two microphones is illustrated in Figure 3.1. The aim is to find the direction of the sound source. We will show that the angle of arrival of the sound can be calculated with the following formula.

$$\alpha = \arcsin\left(\frac{d_1 - d_2}{d}\right) \quad (3)$$

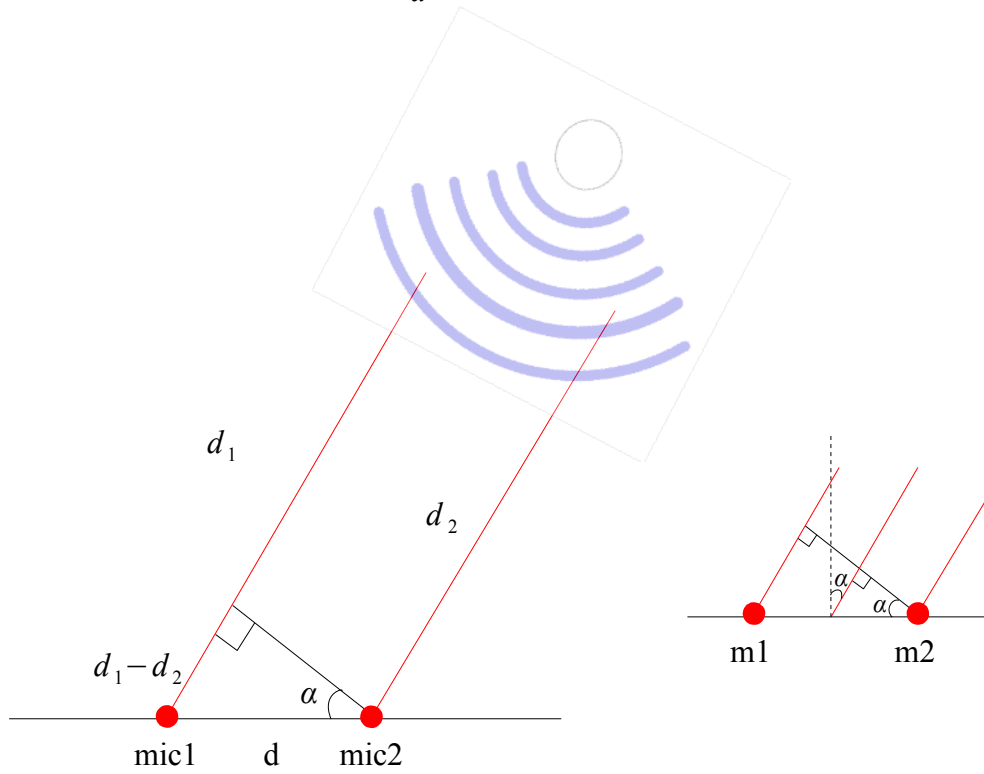


Figure 3.1 System with two microphones

d_1, d_2 : distance from sound source to the microphone 1

d_2 : distance from sound source to the microphone 2

d : distance between microphones

α : Azimuth angle of the sound source

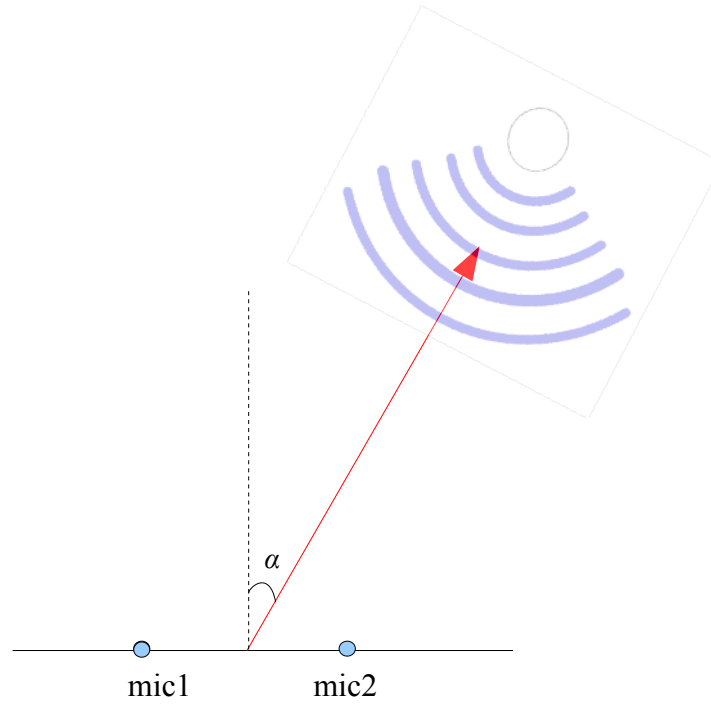


Figure 3.2 Simplified figure

3.2 Mathematical Proof for the Formula

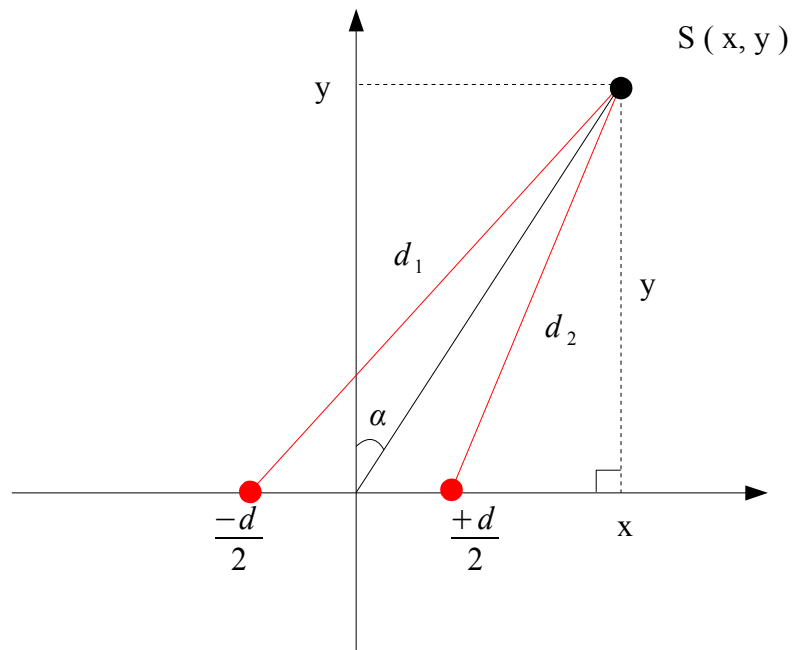


Figure 3.3 Same mechanism on the coordinate system

$$d_1^2 = \left(x + \frac{d}{2}\right)^2 + y^2 \quad (4) \quad d_2^2 = \left(x - \frac{d}{2}\right)^2 + y^2 \quad (5) \quad \tan \alpha = \frac{x}{y} \quad (6)$$

$$d_1 - d_2 = \sqrt{\left(x + \frac{d}{2}\right)^2 + y^2} - \sqrt{\left(x - \frac{d}{2}\right)^2 + y^2} \quad (7), \text{ from (4) and (5)}$$

$$d_1 - d_2 = \sqrt{\left(y \tan \alpha + \frac{d}{2}\right)^2 + y^2} - \sqrt{\left(y \tan \alpha - \frac{d}{2}\right)^2 + y^2} \quad (8), \text{ by substitute (6) on (7)}$$

$$d_1 - d_2 = \sqrt{y^2 \tan^2 \alpha + y d \tan \alpha + \frac{d^2}{4} + y^2} - \sqrt{y^2 \tan^2 \alpha - y d \tan \alpha + \frac{d^2}{4} + y^2} \quad (9)$$

$$d_1 - d_2 = \sqrt{(1 + \tan^2 \alpha) y^2 + y d \tan \alpha + \frac{d^2}{4}} - \sqrt{(1 + \tan^2 \alpha) y^2 - y d \tan \alpha + \frac{d^2}{4}} \quad (10)$$

Limit of both sides while y goes to infinity is,

$$\lim_{y \rightarrow \infty} (d_1 - d_2) = \lim_{y \rightarrow \infty} \left(\sqrt{\frac{1}{\cos^2 \alpha} y^2 + \frac{d \sin \alpha}{\cos \alpha} y + \frac{d^2}{4}} - \sqrt{\frac{1}{\cos^2 \alpha} y^2 - \frac{d \sin \alpha}{\cos \alpha} y + \frac{d^2}{4}} \right) \quad (11)$$

At this point we use the following limit rule

$$\lim_{x \rightarrow \infty} \sqrt{ax^2 + bx + c} = \lim_{x \rightarrow \infty} \sqrt{a} \left| x + \frac{b}{2a} \right| \quad (12)$$

So, our equation has simplified to,

$$\lim_{y \rightarrow \infty} (d_1 - d_2) = \lim_{y \rightarrow \infty} \frac{1}{\cos \alpha} \left(y + \frac{d \sin \alpha \cos \alpha}{2} - y + \frac{d \sin \alpha \cos \alpha}{2} \right) = d \sin \alpha \quad (13)$$

Finally we get,

$$d_1 - d_2 = d \sin \alpha \quad (14)$$

$$\sin \alpha = \frac{d_1 - d_2}{d} \quad (15)$$

$$\alpha = \arcsin\left(\frac{d_1 - d_2}{d}\right) \quad (16). \text{ We got the Formula (3).}$$

But this equation does not provide us the exact solution. The answer is close when the sound source is far from microphones. But if it is near, we get some errors.

3.3 Examination of the Error Rate of the Formula

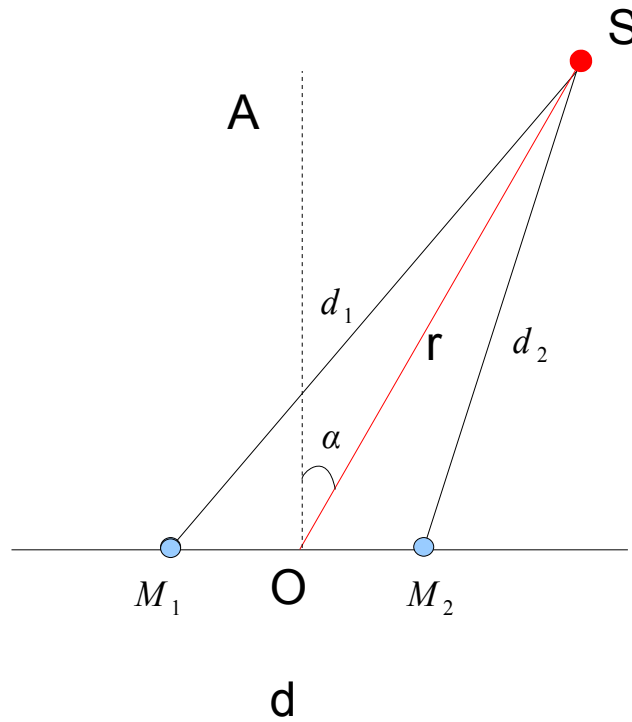


Figure 3.4 Example for error rate

d : distance between microphones $|M_1 M_2|$

O : middle point of the microphones.

S : sound source

r : distance from O to S ($|OS|$)

$[OA]$: perpendicular bisector of the microphones

α : real angle between $[OA]$ and $[OS]$

α' : calculated value of α

Example situation :

$$r = d = 20 \text{ cm}, \alpha = 45^\circ$$

Solution :

$$d1 = 27.98 \text{ cm} \quad d2 = 14.74 \text{ cm} \quad d1 - d2 = 13.24 \text{ cm}$$

$$\alpha' = \arcsin(13.24 / 20) = 41.46^\circ,$$

$$\text{angle difference} = 3.54^\circ \quad \text{error rate} \sim \% 8$$

Calculated angle value is %8 less than the real value. The acceptance of this error rate differs between applications. To make the error rate in acceptable limits, we would choose longer r values. If we choose $r = 5d$ for instance, we get $\alpha' = 44.86^\circ$ and the error rate %0.3.

How many times the distance between the microphones away from sound source must be? In other words, how short the distance between microphones should be kept? To answer these questions by looking our fault tolerance, we need a table.

Table 3.1 Average error rate for different ranges

d (cm)	r (cm)	r/d	Angle Difference $ \alpha' - \alpha $	Location Difference S-S' (cm)	Average Error rate
10	5	0.5	12.235 °	1.068	27.19%
10	1	1	3.536 °	0.617	7.86%
15	1.5	1.5	1.587 °	0.416	3.53%
20	2	2	0.895 °	0.312	1.99%
25	2.5	2.5	0.573 °	0.250	1.27%
30	3	3	0.398 °	0.208	0.88%
35	3.5	3.5	0.292 °	0.179	0.65%
40	4	4	0.224 °	0.156	0.50%
50	5	5	0.143 °	0.125	0.32%
60	6	6	0.099 °	0.104	0.22%
70	7	7	0.073 °	0.089	0.16%
80	8	8	0.056 °	0.078	0.12%
90	9	9	0.044 °	0.069	0.10%
100	10	10	0.036 °	0.062	0.08%
200	20	20	0.009 °	0.031	0.02%
300	30	30	0.004 °	0.021	0.01%
400	40	40	0.002 °	0.016	0.00%
500	50	50	0.001 °	0.012	0.00%

Now, we might decide by using this table. We understand that choosing r values less than d , gives unreliable results (% 27 error rate). Consequently, the following figure does not include that value.

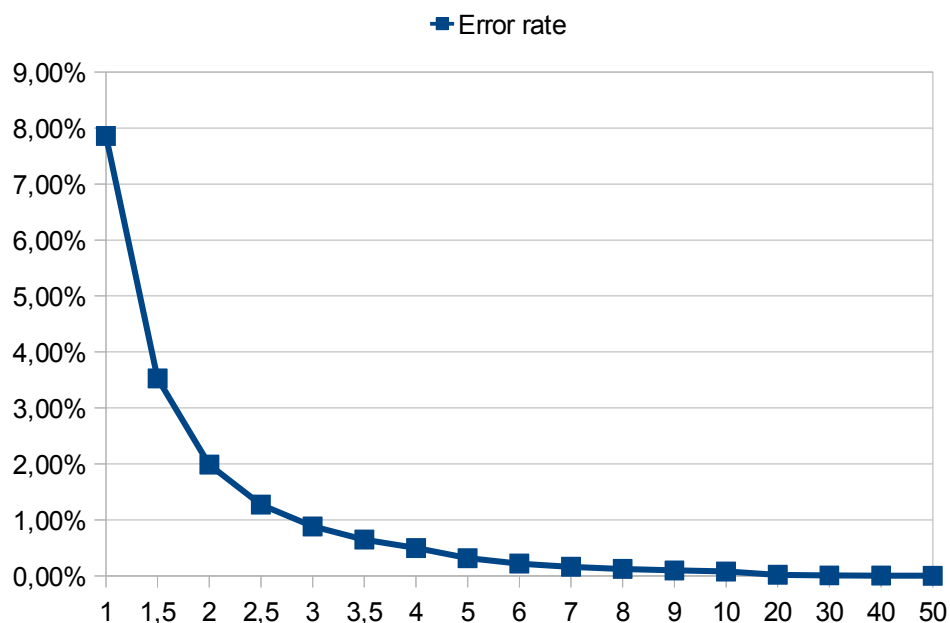


Figure 3.5 Error rates for the different r/d values

As seen on the figure 3.5, to get results with less than %1 fault we should choose at least the three times the distance between microphones for the range of the sound source, or we should change the distance between microphones to one third of the range of the sound source. Of course, the longer distances give the more reliable results. For $r > 5d$ we get nearly %0.1 error rate, which means very sensitive.

The preceding values calculated for the angle 45° . It gives average error rate. You may see detailed results from 0° to 90° in Figure 3.6. The real location of the sound source is shown in blue and the calculated one is shown in red.

The smaller angle values produce the smaller errors. Again it is seen that the long distances for the sound source gives smaller errors. The other half of the diagram gives the same result as shown in Figure 3.7. Similarly, the other half plane is the same.

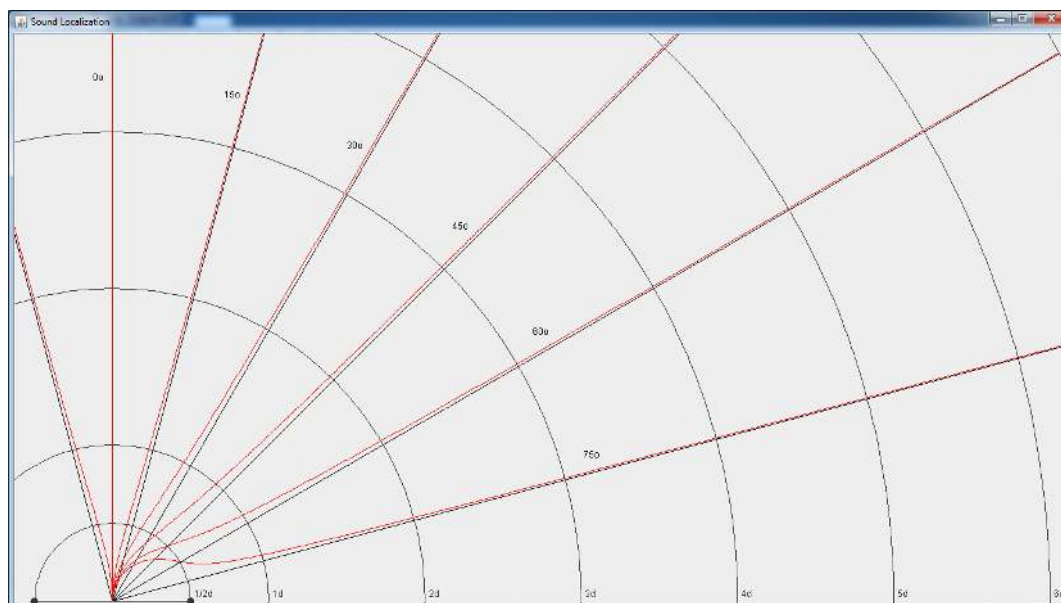


Figure 3.6 Comparison of the results of the formula with the real coordinates

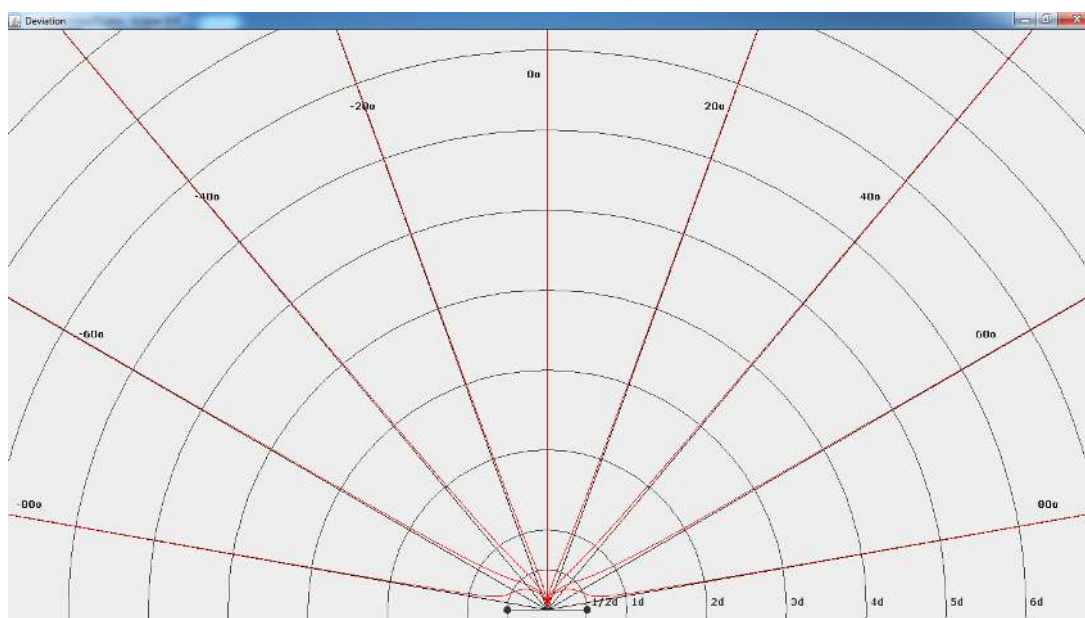


Figure 3.7 Half plane

Example situation : A user in front of a laptop computer

$$r \approx 70 \text{ cm}$$

Acceptable error rate : %1



Figure 3.8 Average distance from computer

In this situation we should make $d = 25 \text{ cm}$ or less to get the required error rate.

3.4 Time Difference Calculation

To calculate distance difference between sound source and both microphones, we need time differences.

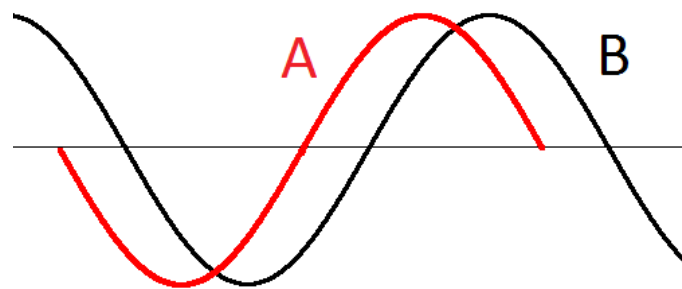
$$x = v \cdot t$$

$$d_1 - d_2 = v \cdot (t_1 - t_2)$$

$$\Delta d = v \cdot \Delta t$$

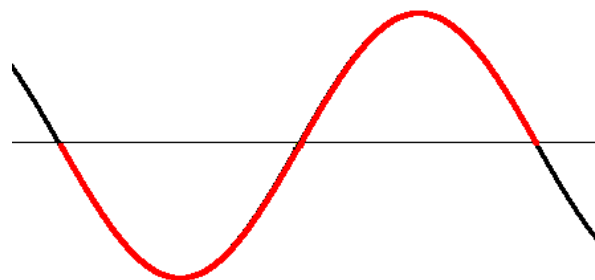
v is the speed of sound, all we need is calculating time differences of arrival of the sound to the both microphones. Both microphones get the same sound signal with different arrival time. There is a time shift between acquired signals. How can we calculate this?

Example for phase shift :



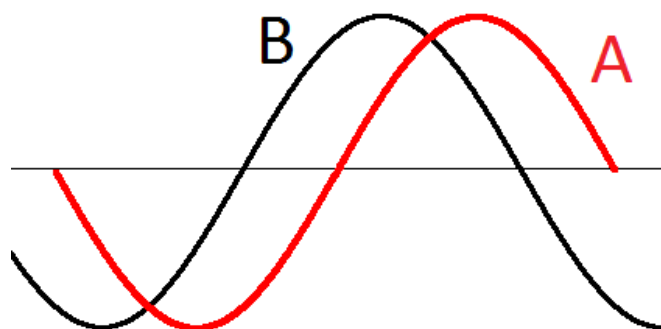
A leads B

Figure 3.9 The red signal (A) comes first, B follows A.



0 phase shift

Figure 3.10 Both signals arrive at the same time.



A lags B

Figure 3.11 The red signal comes later, after B

A simple solution to find the time shift explained step by step below

- Keep one of the signals fixed
- Move the other one to the left and to the right in time domain
- Calculate the differences between signals for each move
- Find the minimum difference

The time difference between the beginning case and the case of the lowest difference gives us the time shift.

Example 1: To understand the steps, we use sinusoidal signals. The red signal is shifted 60° . The blue signal is difference signal $s_1 - s_2$.

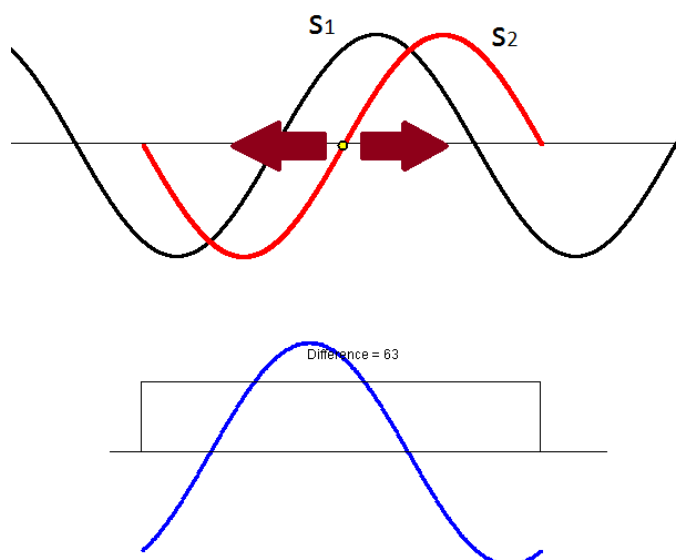


Figure 3.12 Example for shifted signal

Some results are showed in Figure 3.13. The two signals are the most overlap on the second case in the figure.

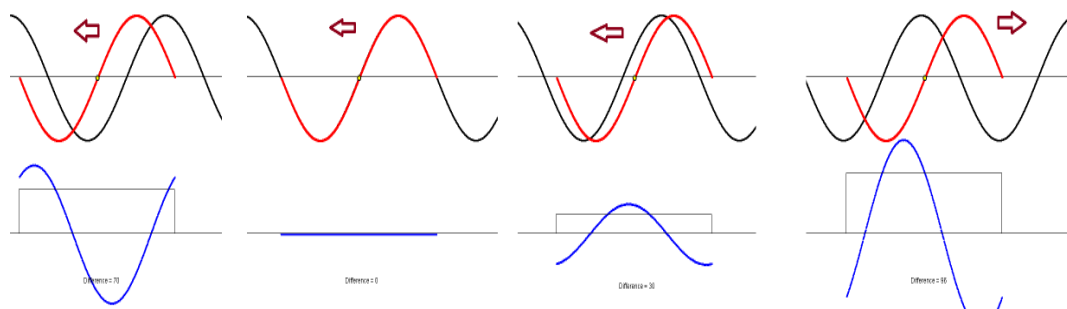


Figure 3.13 Moving shifted signal both sides

Example 2: This time, signals are a bit complex. But the technique is the same. At the beginning the red signal is 90° shifted in Figure 3.14. A computer program is written and used to calculate differences for one period time. The differences are drawn in blue on the Figure 3.15

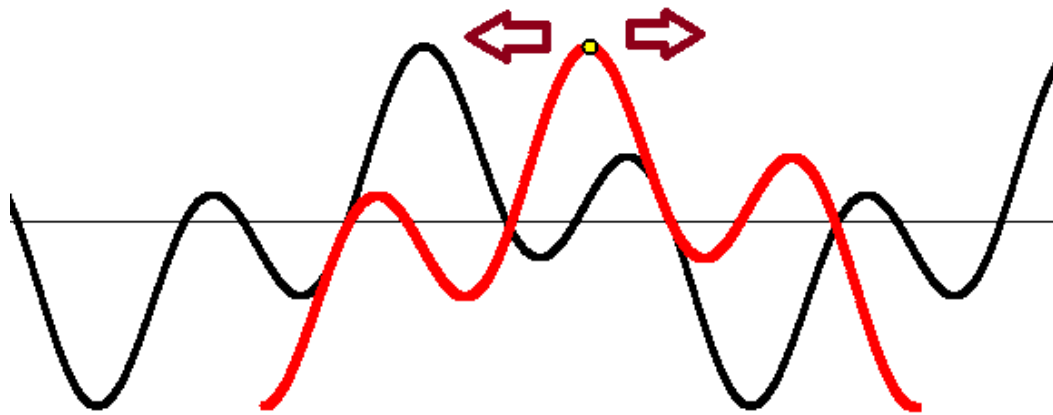


Figure 3.14 Another example with more complex signals

Program runs and finds the differences. The most similar point is showed in yellow circle.

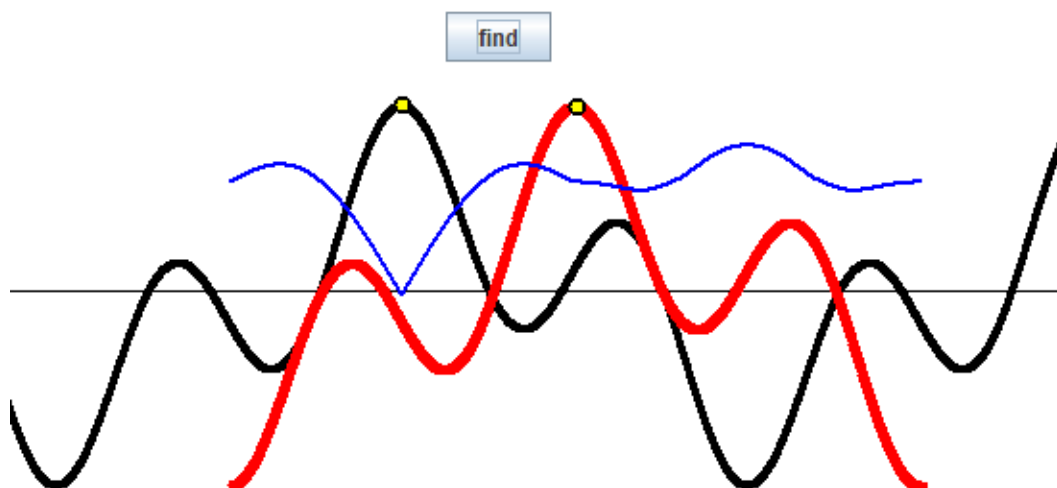


Figure 3.15 Solution of a previous figure

3.5 Frequency Constraint

Let the maximum frequency of the signals be f_m . So the period is

$$T_m = \frac{1}{f_m}.$$

Let the duration of moving the shifted signal be T_s ,

$\frac{T_s}{2}$ to the left and $\frac{T_s}{2}$ to the right.

What happens if $T_s > T_m$? Next figure shows the case $T_s = 2 \cdot T_m$.

The red signal is shifted 90° .

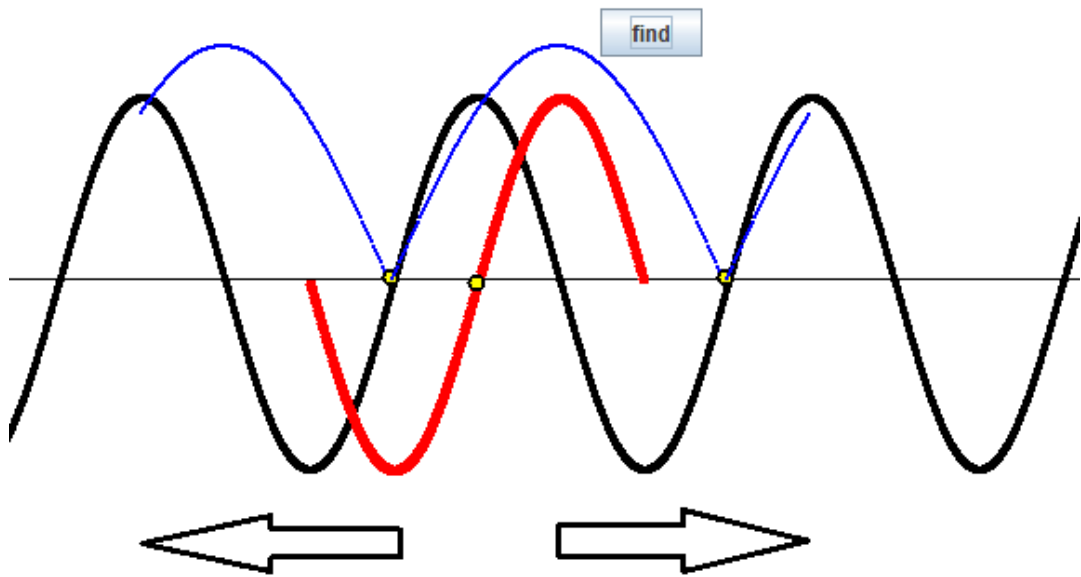


Figure 3.16 Example of a frequency constraint

The program found two most similar points. The first one is 90° left and the other one is 270° right. Which one is the real solution? It is impossible to answer this question. To get only one solution, we should choose the moving cycle time less than the maximum period $T_s < T_m$ or test signals with a period greater than the moving time $T_m > T_s$.

3.6 The Frequency That Can Be Determined

This constraint is also valid for our microphone mechanism. The maximum time shift occurs when the $\alpha = -90^\circ, \alpha = 90^\circ$.

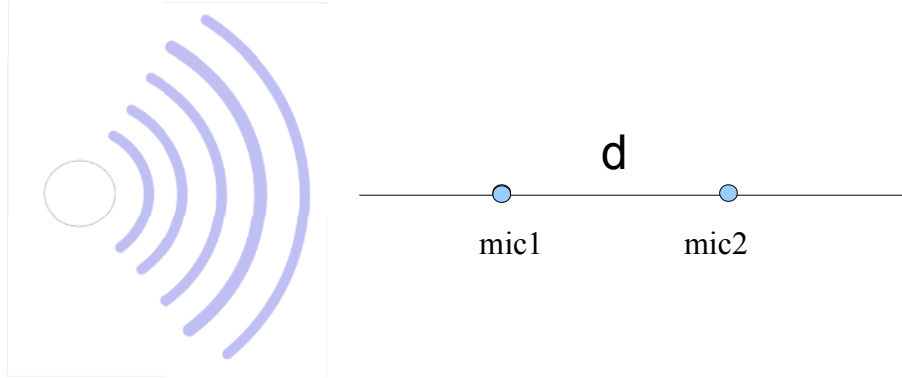


Figure 3.17 The sound source is completely on the left

The difference of the arrival times of the sound to the microphones is Δt .

$$\Delta t = t_2 - t_1$$

$$d = v \cdot \Delta t \quad , \quad (v : \text{sound speed})$$

The shifted signal may shift Δt to the left or Δt to the right or between them.

$$\text{Therefore } T_s = 2\Delta t = \frac{2d}{v} \quad .$$

We explained that $T_m > T_s$ before.

$$T_m > \frac{2d}{v} \quad , \quad f_m = \frac{1}{T_m}$$

$$f_m < \frac{v}{2d} \quad \text{or} \quad d < \frac{v}{2f_m}$$

Example 1: $d = 25 \text{ cm}$, what is the maximum frequency that can be determined?

$$v = 343 \text{ m/s}, \quad d = 0.25 \text{ m} \quad f_m < \frac{343 \text{ m/s}}{2 \cdot 0.25 \text{ m}} \quad f_m < 686 \text{ Hz}$$

Example 2 : The frequency of the sound source is 3000 Hz. What should be the minimum length of the distance between microphones ?

$$v = 343 \text{ m/s}, \quad f_m = 3000 \text{ Hz} \quad d < \frac{343 \text{ m/s}}{2 \cdot 3000 \text{ 1/s}} \quad d < 5.72 \text{ cm}$$

Table 3.2 lists the frequencies can be determined safely at certain distances between microphones.

Table 3.2 The frequencies that can be determined for pure tones

Distance between microphones	Max frequency can be determined
1 cm	17 kHz
2 cm	8500 Hz
5 cm	3400 Hz
10 cm	1700 Hz
20 cm	850 Hz
50 cm	340 Hz
1 m	170 Hz
2 m	85 Hz

These values are calculated for the simulation with pure tones. Fortunately, we don't encounter such pure tones in natural environment. For example, the human voice has a complex structure. This constraint shall not cause a problem for the hardware solutions for human speech.

CHAPTER FOUR

SOFTWARE IMPLEMENTATION

This software aims user to understand and practice sound source localization with two microphones on two dimensional planes. The sound source, microphones and the measurement bars are all moveable.

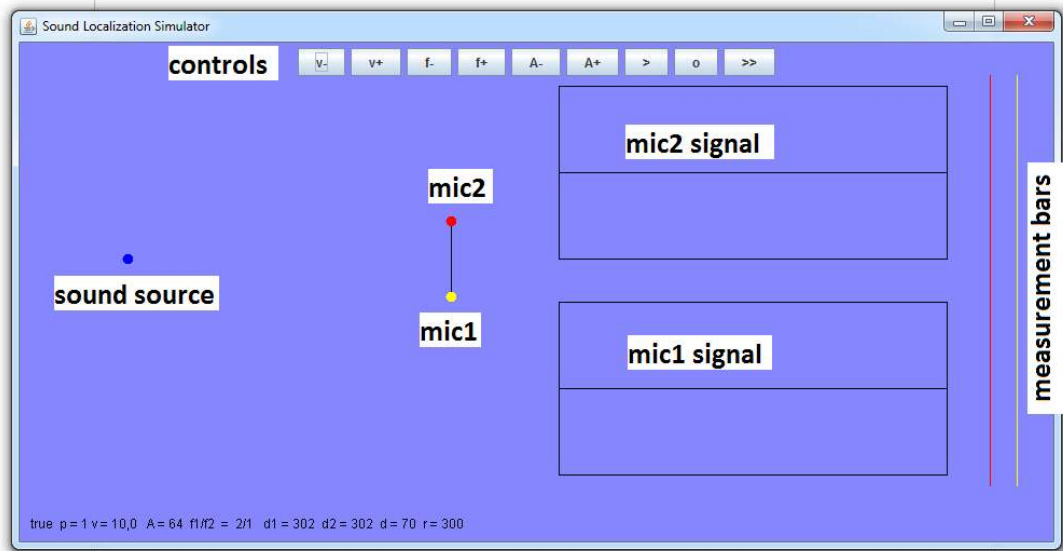


Figure 4.1 Simulator software without the waves

Control panel buttons;

- v-, v+ : adjust simulation speed
- f-, f+ : adjust source frequency
- A-, A+ : adjust source amplitude
- >, || : start, pause
- o : reset
- >> : simulate step by step

The following codes use the frequency and amplitude variables in a loop and shows the sound signal with changing colors of circles. The sound is also damped on the medium. The amplitude decreases and at the end the vibration stops. This effect can be added easily like following code “ `colors[i] = colors[i-1] * 0.997;` ”. Changing the coefficient 0.997, changes the damping speed.

```
...
int color = 128 - (int) ( A * Math.sin( Math.toRadians( p * f ) ) );
g.setColor( new Color( color, color, 255 ) );
g.drawOval( source.x - r, source.y - r, 2 * r, 2 * r );
...
```

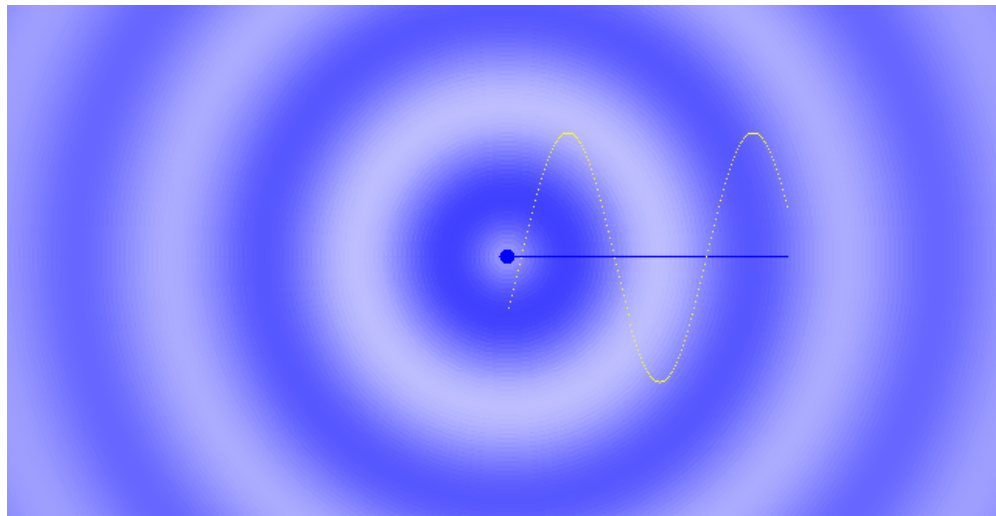


Figure 4.2 Simulated sound waves

The simulation starts and the microphones get sound. When you look at the microphone panels, you see that they have shifted signals. If you put the bars on the similar points on the graphs, like peak values, program automatically calculates the angle and shows it with a line from the midpoint of the microphones and writes the numeric value on the screen.

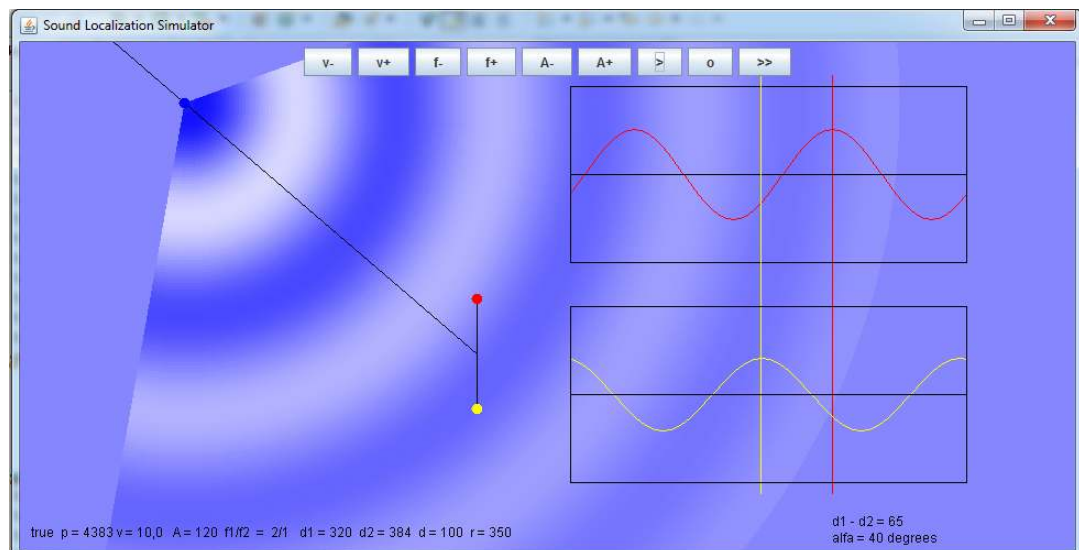


Figure 4.3 Simulator software with the waves

On the Figure 4.3 the red bar is on the peak value of the signal of mic2 and the yellow bar is on the peak value of the signal of mic1 (moved by user). While bars are moving, calculated angle value and the line from midpoint of microphones are also changing in an interactive manner.

CHAPTER FIVE

HARDWARE IMPLEMENTATION

This thesis has been considered an inexpensive, interactive and educational software implementation.. According to our research, most of the other applications use microphone arrays which are expensive, needs more complex approach, not accessible to everyone and consume more computing power of processors.

We have realized that the microphone inputs are stereo. The sound cards are capable to process two microphone signals separately. It's a kind of microphone array with two microphones. In previous chapters it's explained that two microphones are enough for the finding direction of the sound source with some errors. No need for extra costs, just two microphones and the sound card of the computer. So we decided to realize our software implementation in hardware.

5.1 Combined Microphones

In Figure 5.1 two microphones are combined as a one microphone with left and right channels.



Figure 5.1 combined microphones

5.2 Audio Signal Sampling

Sound waves are converted into electrical signals by microphones. Sound cards then sample this analog electrical signals and convert it to digital signals. Figure 5.2 shows 4 bits pulse code modulation.

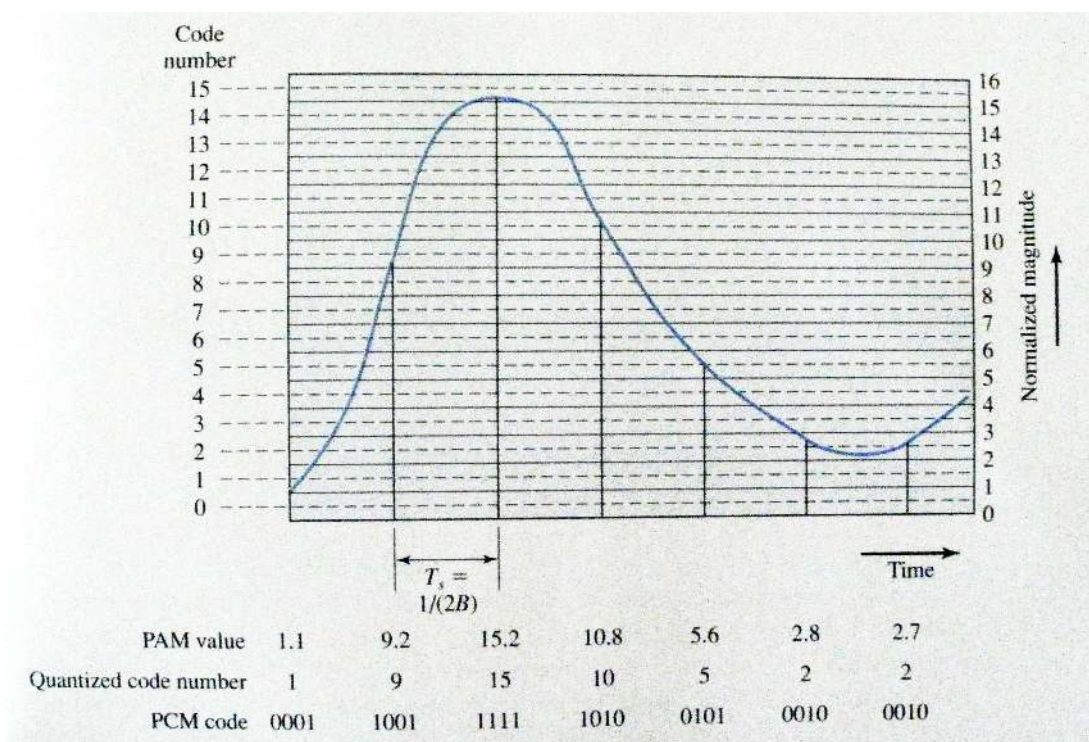


Figure 5.2 Pulse code modulation “PCM” example (Stallings, 2007, pg. 163)

At this time, sound cards usually samples with 16 bits (2 Bytes) signed format and 96000 Hz sample rate. 44100 Hz sample rate means the signal is sampled 44100 times per second, with 22.68 μ s time difference between samples. 16 bits signed format means sound signal gets integer values between -32768 and +32768.

5.3 Getting Data from Sound Card

Before getting data from the sound card, we set the properties to suitable values.

```
//Get everything set up for capture
float sampleRate = 44100.0F; //44100,48000,96000
int sampleSizeInBits = 16; // 16 bits = 2 x 8 bits = 2 bytes
int channels = 2; // 1:mono, 2:stereo 2 bytes Left + 2 bytes Right = 4 bytes
boolean signed = true; // true : [ -32768 32768 ] , false : [ 0 65536 ]
boolean bigEndian = true; // most significant bits first
...
```

Then, we get four bytes of data for each sample as seen on Table 5.1

Table 5.1 Data for one sample

0	1	2	3
1. Byte	2. Byte	3. Byte	4. Byte
Left Microphone		Right Microphone	

Ordinary microphones are not very sensitive. Figure 5.3 shows a part of recorded audio from an ordinary microphone. As seen on the graph, there are sudden ups and downs due to the quality of the microphone and the white noise which comes from internal and external environment.

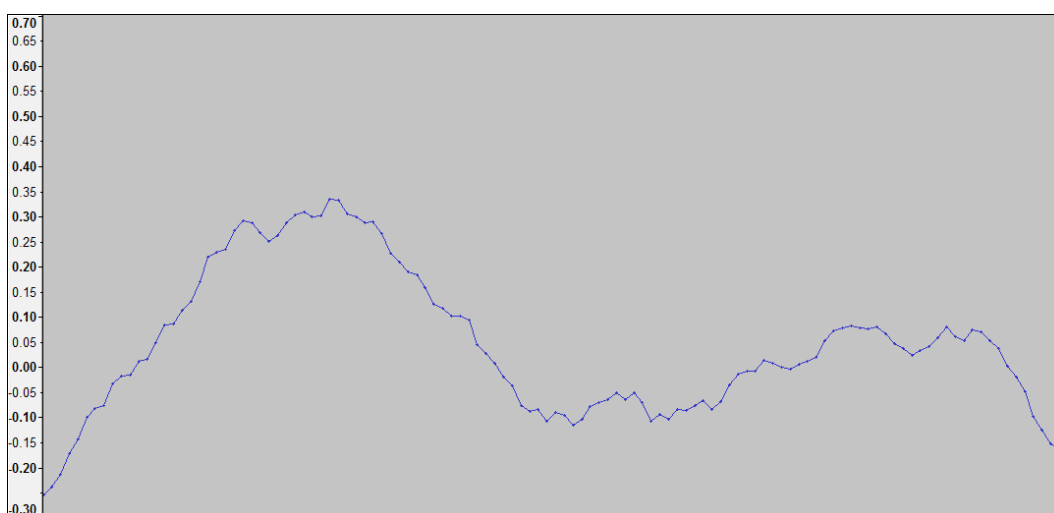


Figure 5.3 Recorded audio with ordinary microphone

This noise makes the second byte of the signal meaningless. The second byte is $\frac{1}{256}$ of the whole information. About %1 percent of the data is negligible. So only the first 8 bits of the data is used.

```
audioDataLeft[k] = sampleSound[4*i+0]; // the first byte of left mic
audioDataRight[k] = sampleSound[4*i+2]; // the first byte of right mic
```

5.4 Finding Distance Difference

The maximum time shift occurs when the sound source is completely on the left or on the right explained before with the figure 3.17. The minimum time delay occurs when the sound source is equidistant from microphones. These three cases figured again and sampled from sound card. The distance between microphones is chosen 25cm.

Case 1 : On the Left

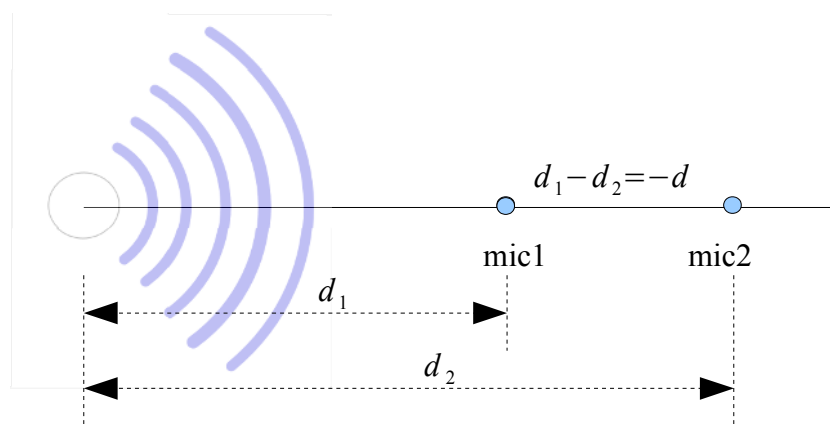


Figure 5.4 Completely Left, -90°

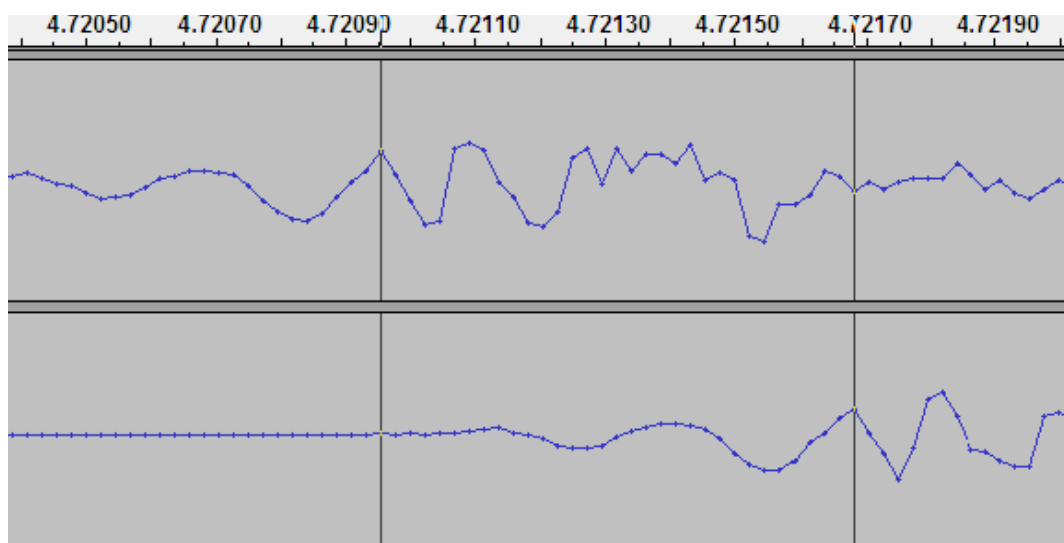
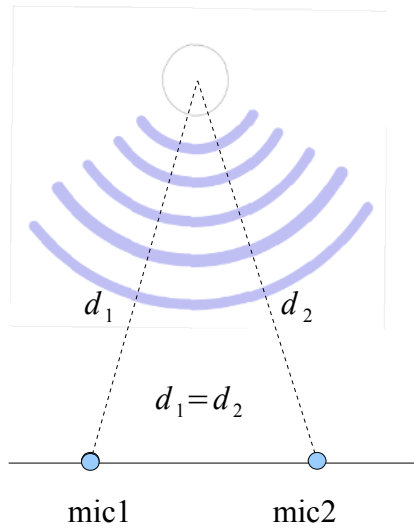
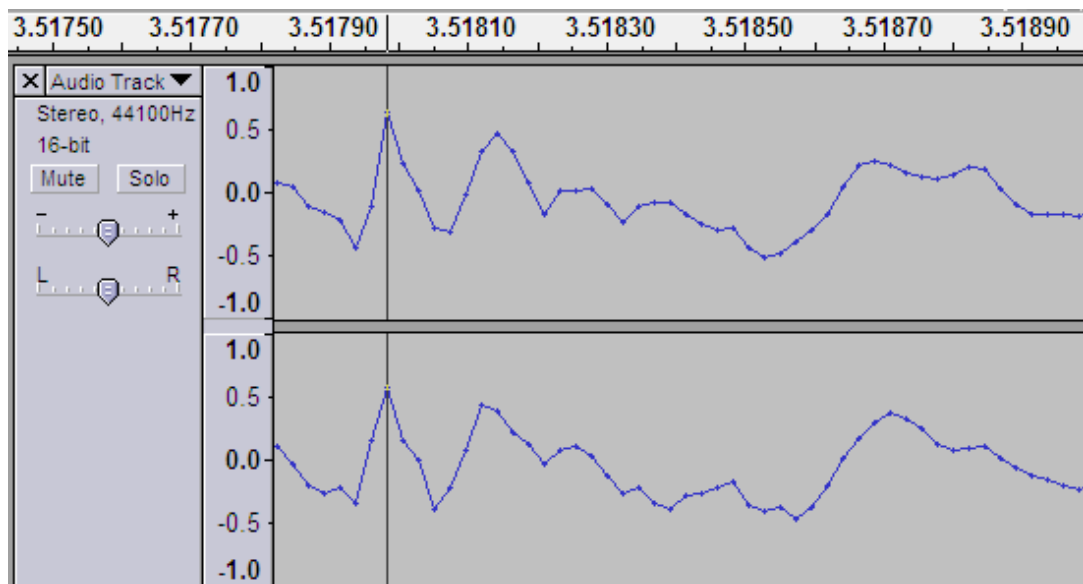
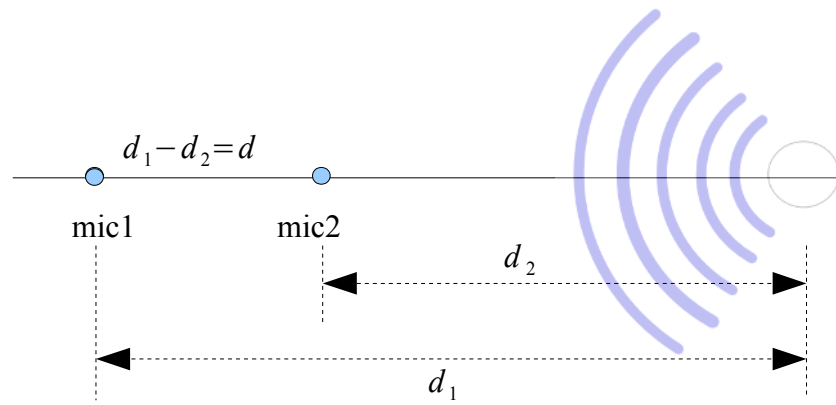
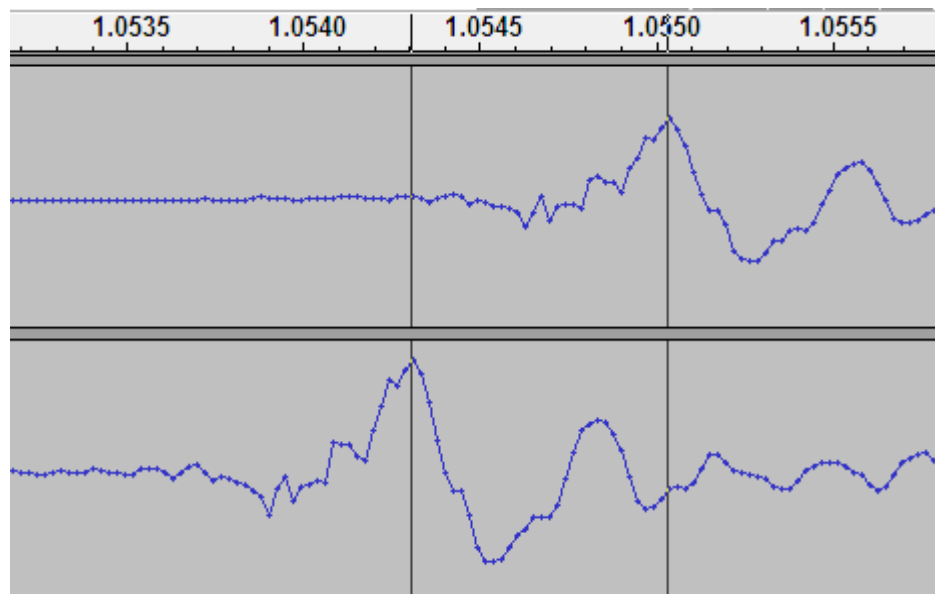


Figure 5.5 mic2 gets 32 samples later, 726 μ s time delay

Case 2 : Equidistant

Figure 5.6 Equidistant, 0° Figure 5.7 mic1 and mic2 gets the sound signal at the same time, 0 μ s time delay

Case 3 : On the Right

Figure 5.8 Completely Right, 90° Figure 5.9 mic2 gets sound 32 sample earlier, 726 μs time delay

The results are compatible with our calculations.

$$\Delta t_{max} = \frac{d}{v} = \frac{25 \cdot 10^{-2}}{343} \approx 729 \cdot 10^{-6}$$

The delay between microphones is at most 32 samples and at least 0. It's sufficient to look in the range 32 samples left and 32 samples right to get the time delay. To compare the signals in this range, we must have recorded data $2 \cdot \Delta t_{max}$ length. For these example situations we need about 1.5 ms recorded data. The time delay multiplied by the speed gives us the distance difference. We then use this value to get the angle.

5.5 Pre-calculated Angle Values

The data from sound card is discrete. Therefore, the time difference of the signals of left and right microphones are also discrete. This allows us to pre-calculate the angle values. Table 5.2 shows some example situations and Table 5.3 lists the calculated angle values for $f=96000$ Hz and $d = 25$ cm.

The following formula is used for the speed of sound

$$v = 331 \sqrt{1 + \frac{T}{273}} \quad T : \text{temperature in Kelvin}$$

There is another simpler formula, nearly gives the same result. $v = 331 + 0.6T$
T is temperature in Celsius. This formula is used in the following table.

Table 5.2 Maximum sample shifts

Temperature	Sound Speed = $331 + 0.6T$	Distance Between Microphones	Maximum Time Difference = $d / v (\cdot 1,000,000)$	Sample Rate (sample per second)	Sampling Period	Maximum Sample Shift
T, °C	v, m/s	d, cm	μs	f, 1/s	t, μs	±90°
20	343	25	728.9	44100	22.68	32.14
				48000	20.83	34.99
				96000	10.42	69.97

These calculated angle values can be used in a program. Sample code is below.

```
// calculated angle values (70 left + 1 middle + 70 right = 141 values)
int[] angles = {-90,-80,-76,-73,-71,-68,-66, ... -3,-2, -2,-1,0,1,2,2,3, ... 90};
```

Table 5.3 Pre-calculated angle values

Sample Shift Count	Time Difference	$\Delta d = d1-d2$ $=v \cdot \Delta t$	Angle = $\arcsin(\Delta d / d)$
n	$\Delta t, \mu s$	$\Delta d, cm$	$\alpha, degrees$
0	0.00	0.00	0.00
1	10.42	0.36	0.82
2	20.83	0.71	1.64
3	31.25	1.07	2.46
4	41.67	1.43	3.28
5	52.08	1.79	4.10
6	62.50	2.14	4.92
7	72.92	2.50	5.74
8	83.33	2.86	6.57
9	93.75	3.22	7.39
10	104.17	3.57	8.22
11	114.58	3.93	9.04
12	125.00	4.29	9.88
13	135.42	4.64	10.71
14	145.83	5.00	11.54
15	156.25	5.36	12.38
16	166.67	5.72	13.22
17	177.08	6.07	14.06
18	187.50	6.43	14.91
19	197.92	6.79	15.76
20	208.33	7.15	16.61
21	218.75	7.50	17.47
22	229.17	7.86	18.33
23	239.58	8.22	19.19
24	250.00	8.58	20.06
25	260.42	8.93	20.93
26	270.83	9.29	21.81
27	281.25	9.65	22.70
28	291.67	10.00	23.59
29	302.08	10.36	24.49
30	312.50	10.72	25.39
31	322.92	11.08	26.30
32	333.33	11.43	27.22
33	343.75	11.79	28.14
34	354.17	12.15	29.07
35	364.58	12.51	30.01

Sample Shift Count	Time Difference	$\Delta d = d1-d2$ $=v \cdot \Delta t$	Angle = $\arcsin(\Delta d / d)$
n	$\Delta t, \mu s$	$\Delta d, cm$	$\alpha, degrees$
36	375.00	12.86	30.96
37	385.42	13.22	31.92
38	395.83	13.58	32.89
39	406.25	13.93	33.87
40	416.67	14.29	34.87
41	427.08	14.65	35.87
42	437.50	15.01	36.89
43	447.92	15.36	37.92
44	458.33	15.72	38.96
45	468.75	16.08	40.03
46	479.17	16.44	41.10
47	489.58	16.79	42.20
48	500.00	17.15	43.31
49	510.42	17.51	44.45
50	520.83	17.86	45.61
51	531.25	18.22	46.79
52	541.67	18.58	48.00
53	552.08	18.94	49.24
54	562.50	19.29	50.51
55	572.92	19.65	51.82
56	583.33	20.01	53.16
57	593.75	20.37	54.55
58	604.17	20.72	55.99
59	614.58	21.08	57.48
60	625.00	21.44	59.04
61	635.42	21.79	60.67
62	645.83	22.15	62.39
63	656.25	22.51	64.21
64	666.67	22.87	66.16
65	677.08	23.22	68.27
66	687.50	23.58	70.60
67	697.92	23.94	73.24
68	708.33	24.30	76.37
69	718.75	24.65	80.44
70	729.17	25.01	90.00

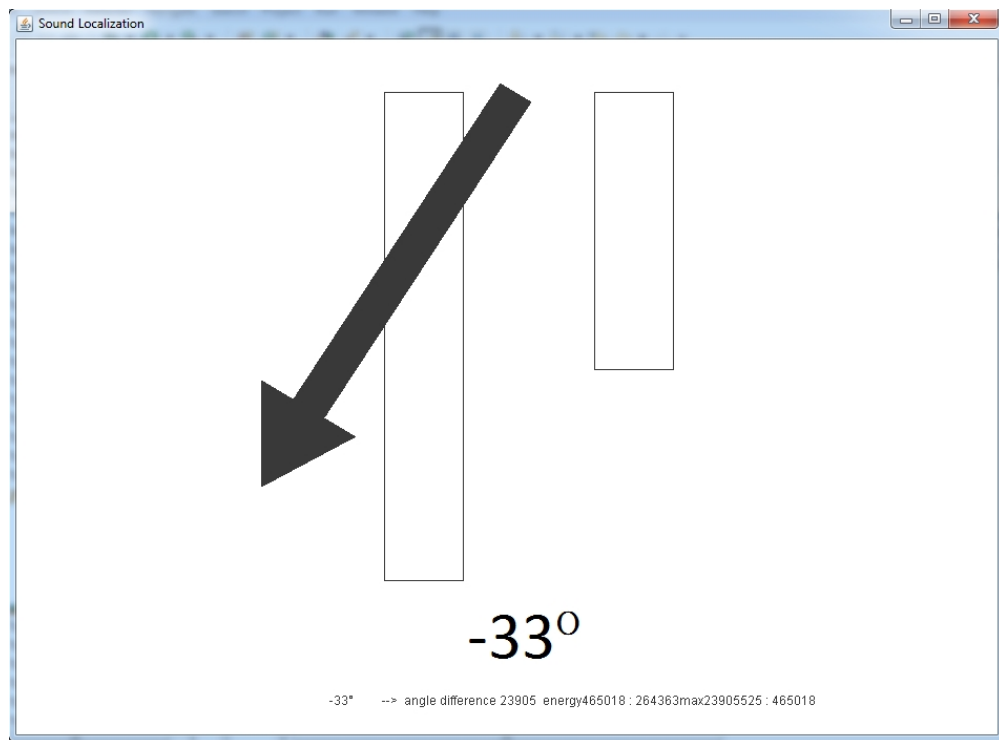


Figure 5.10 Screen shot of the program

Figure 5.9 shows a screen shot of a running program that uses pre-calculated angle values. The two rectangles are indicators of the energy of the signals of the microphone, but they are not used for calculations.

5.6 Difference Signal and a Solution to Overcome Discrete Data

If you calculate the differences between the sound signals of left and right microphones for each shift, you get a difference values as illustrated in Figure 5.11. Remember that we use the minimum point to calculate the time shift between signals. After that we multiply this time by sound speed to get the distance difference. And finally we use distance difference to get the angle of the sound source. The time shift errors mean the angle errors. The better time difference values give the better angle values.

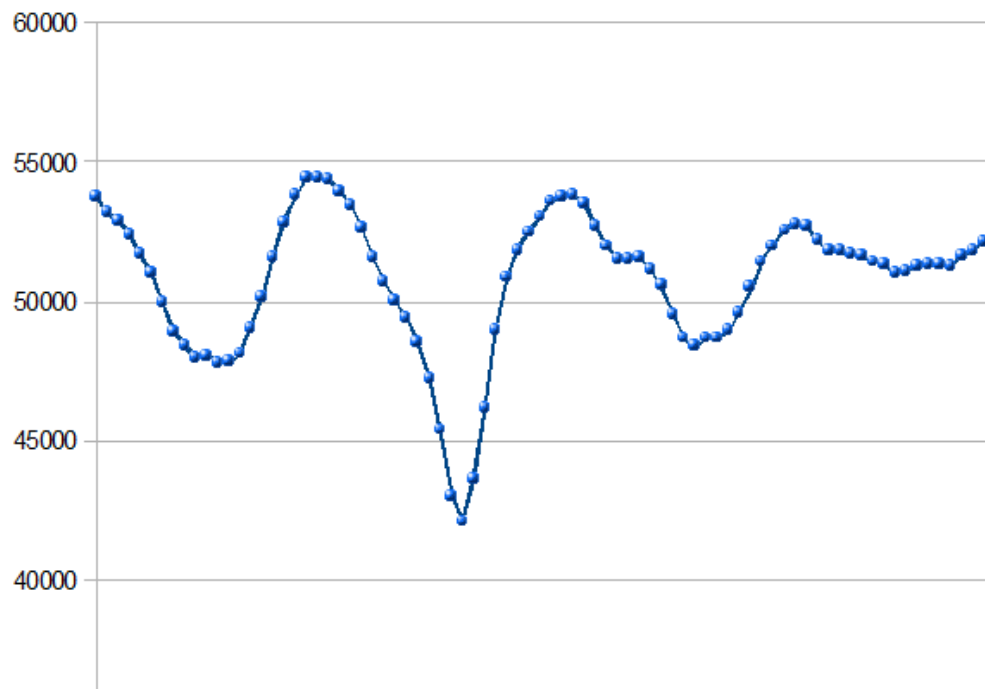


Figure 5.11 Difference values

The audio signal is sampled. It requires us to choose one of the sampling points. But this point may not be the real point. It may be between two samples. We can overcome this obstacle with some geometric calculations. At first, we analyze the three possible situations figured below.

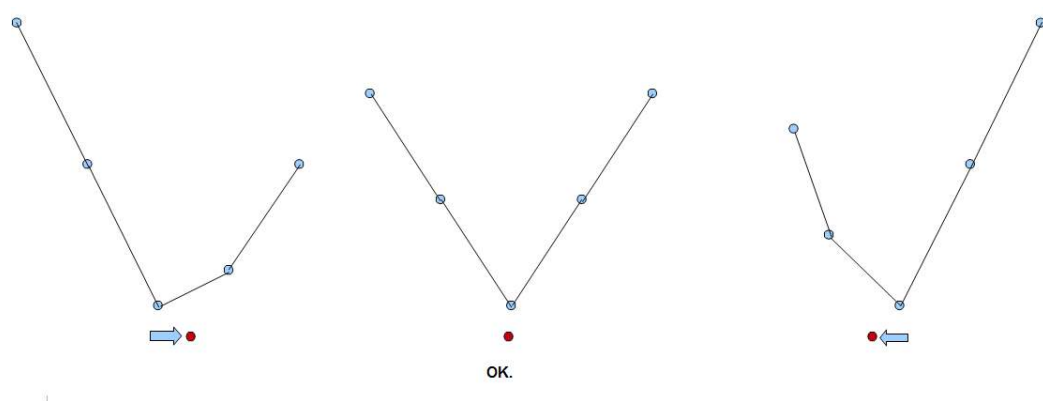


Figure 5.12 Possible situations for the minimum point

Red dots show the possible real minimum points. The x coordinates are sampling times. We need only the x coordinate of the real minimum point.

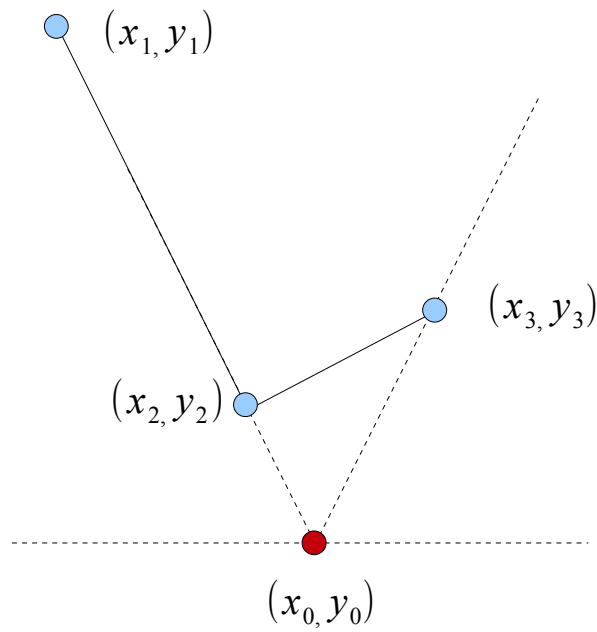
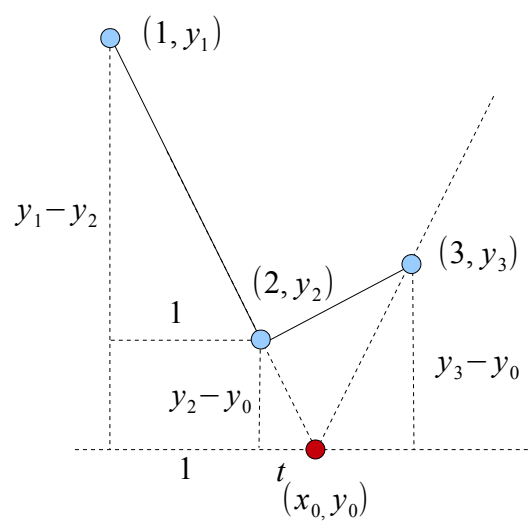


Figure 5.13 Real minimum point

Using some geometric equations, we get it using coordinates of the three points.

$$x_0 = x_2 + \frac{y_1 - y_2}{2(y_1 - y_3)}$$



$$\frac{y_1 - y_0}{1 + t} = \frac{y_3 - y_0}{1 - t} = \frac{y_1 - y_2}{1}$$

$$\frac{y_1 - y_3}{2t} = y_1 - y_2$$

$$t = \frac{y_1 - y_3}{2(y_1 - y_2)}$$

Figure 5.14 Geometric calculations

Some examples are shown in Figure 5.15

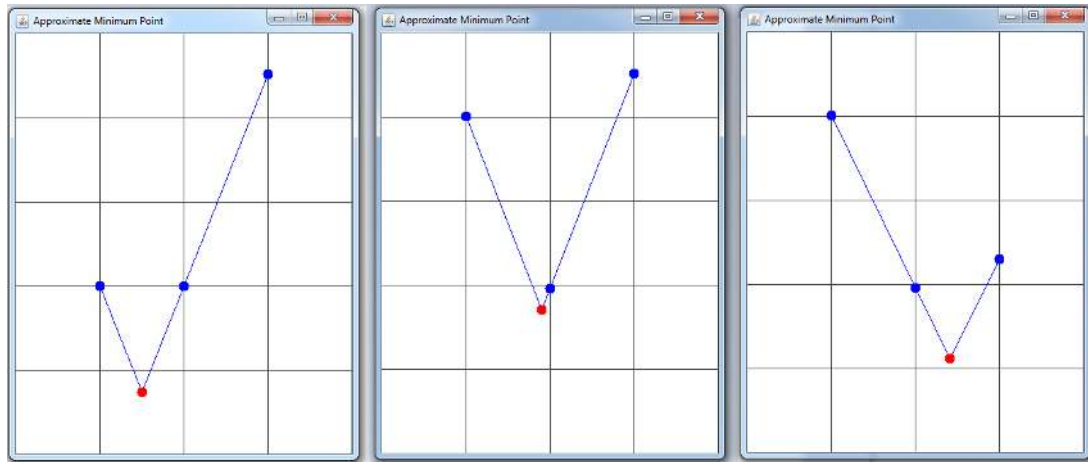


Figure 5.15 Some examples of real minimum points

We had satisfactory results from the experiment using this approximation. In the experiment figured below, following values are chosen; $d = 25$ cm, $r = 100$ cm, sample rate = 44100 Hz. Without estimation, 53° , 54° and 55° degrees gave the same result 54° . With estimation we got degrees 53° , 54° and 55° respectively.

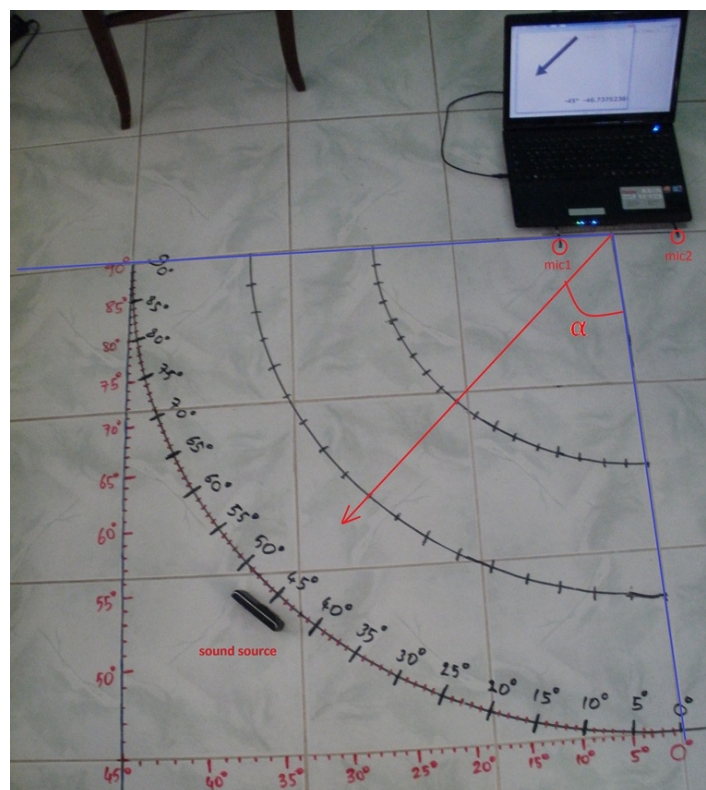


Figure 5.16 Experiment using three point approximation

CHAPTER SIX

DISCUSSION AND CONCLUSION

The aim of this study is to understand the basics of sound source localization. The software and hardware implementations are realized. We are able to change the characteristics of the sound such as frequency and amplitude interactively and simulate the sound source in an absorbent medium. In addition, real time sound locator is implemented.

The error rates are examined in detail for the most common formula in TDOA methods. And also a solution is offered to overcome discrete data. It's answered that how we could decide the direction of the sound with 1 millisecond of time difference, by showing sampled data from sound card.

To get more accurate results, the sampling frequency can be changed, better quality microphones can be used and all 16 bits of data from sound card can be processed.

This study can be expanded by increasing the number of microphones, finding the coordinates of the source, processing intensity differences, adding a thermometer, showing the sound source by laser light, etc.

REFERENCES

- Arthur, N.P., & Richard R.F. (2005). *Sound Source Localization*, Springer Handbook of Auditory Research. NY : Springer Science Business Media
- Phet Interactive Science Simulations* (July 7, 2011). Retrieved October 18, 2011, from <http://phet.colorado.edu/>
- Pourmohammad, A., & Ahadi S.M. (2010). TDE-ILD-Based 2D Half Plane Real Time High Accuracy Sound Source Localization Using Only Two Microphones and Source Counting, *2010 International Conference on Electronics and Information Engineering*.
- Rossing, T.D., Moore, F.R., Wheeler P.A. (2002). *The Science of Sound* (Third Edition). Addison Wesley.
- Stallings, W. (2007). *Data and Computer Communications* (Eight Edition). New Jersey: Pearson Prentice Hall.