Stochastic Signal Processing

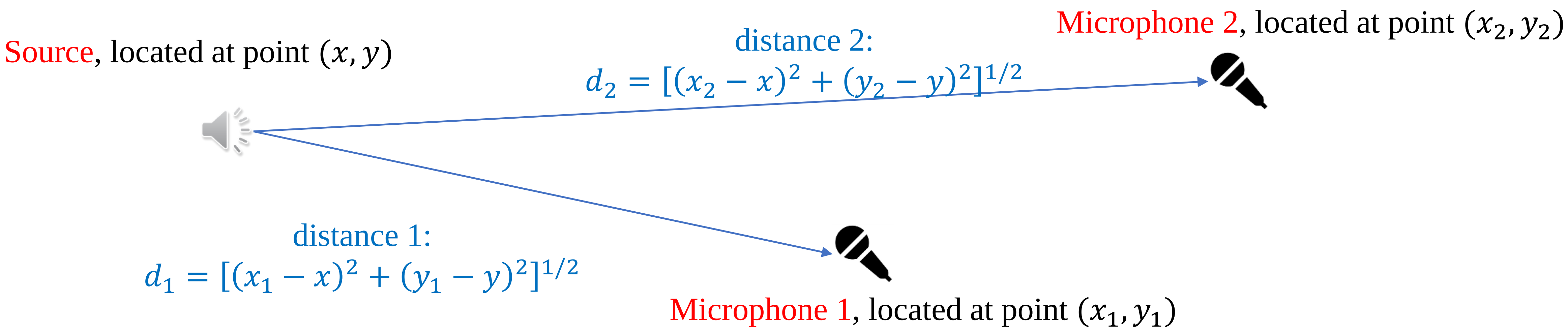
Lesson 8 – Experiment

Application of Autocorrelation: time domain beamforming and localization

Weize Sun

The Microphone Array

- At time 0, power on the microphones 1 and 2, start the source, then the source will send out speech signal
- The distances between the source and microphones 1 and 2 are d_1 and d_2, here we assume $d_1 < d_2$
- Note that the speed at which sound travels through the air is $c = 340m/s$, therefore, at time $t_1 = d_1/c$ seconds, the speech reach microphone 1 - microphone 1 can receive meaningful signal, but microphone 1 still receive nothing, or noise only.
- At time $t_2 = d_2/c$ seconds, the speech reach microphone 2
- Note that what microphone 2 received at time $t_2 = d_2/c$ is what microphone 1 received at time $t_1 = d_1/c$, the time delay is $t_1 - t_2$



The Microphone Array

- At time 0, power on the microphones 1 and 2, start the source, then the source will send out speech signal
- The distances between the source and microphones 1 and 2 are d_1 and d_2, here we assume $d_1 < d_2$
- Note that the speed at which sound travels through the air is $c = 340m/s$, therefore, at time $t_1 = d_1/c$ seconds, the speech reach microphone 1 - microphone 1 can receive meaningful signal, but microphone 1 still receive nothing, or noise only.
- At time $t_2 = d_2/c$ seconds, the speech reach microphone 2
- Note that what microphone 2 received at time $t_2 = d_2/c$ is what microphone 1 received at time $t_1 = d_1/c$, the time delay is $t_1 - t_2$
- For example, if $d_1 = 17$ and $d_2 = 34$, then $t_1 = 0.05$ and $t_2 = 0.1$, thus the time delay is $t_1 - t_2 = -0.05s$, '$-$' means that microphone 1 leads (领先) microphone 2 by 0.05 seconds.
- For example, if $d_1 = 34$ and $d_2 = 17$, then $t_1 = 0.1$ and $t_2 = 0.05$, thus the time delay is $t_1 - t_2 = +0.05s$, '$+$' means that microphone 1 behinds (落后) microphone 2 by 0.05 seconds.
- This is the situation of continuous case, how about discrete case?

The Microphone Array

- At time $t_1 = d_1 /c$ & $t_2 = d_2 /c$ seconds, the speech reach microphone 1 & 2, the time delay is $t_1 - t_2$
- For example, if $d_1 = 17$ and $d_2 = 34$, then $t_1 = 0.05$ and $t_2 = 0.1$, thus the time delay is $t_1 - t_2 = -0.05s$, '$-$' means that microphone 1 leads (领先) microphone 2 by 0.05 seconds.
- If the microphone take 100 samples in 1 second, 0.05 second means 5 samples.
- For example, the source send out signal [1,2,3,4,5,6,7,8,9,10], 10 samples (the length is 0.1 second)
- Start from time 0, microphone 1 & 2 receive (from time 0 to 0.2 second):
  - Signal 1 from microphone 1: [0,0,0,0,0,1,2,3,4,5,6,7,8,9,10,0,0,0,0,0]
  - Signal 2 from microphone 2: [0,0,0,0,0,0,0,0,0,0,1,2,3,4,5,6,7,8,9,10]
- If there are no noise, any microphone can give a clear and clean speech result.
- However, there are noise!
- It is better to add what received in both microphones together to get a better speech, instead of using only one!
- However, if add them up simply, will get [0,0,0,0,0,1,2,3,4,5,7,9,11,13,15,6,7,8,9,10], there is overlap! Even worse!
- How to solve this problem?

The Microphone Array

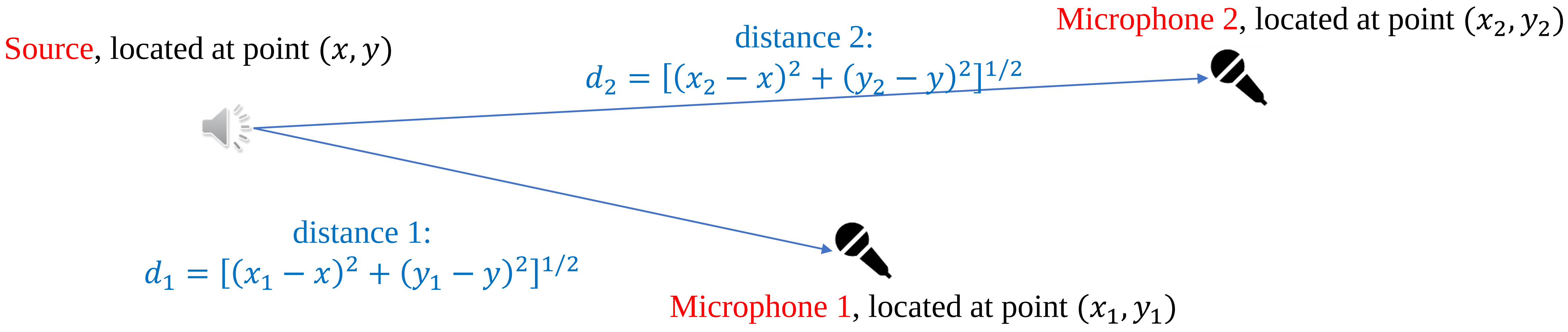
- Cross-Correlation!
  - Signal 1 from microphone 1: [0,0,0,0,0,1,2,3,4,5,6,7,8,9,10,0,0,0,0,0]
  - Signal 2 from microphone 2: [0,0,0,0,0,0,0,0,0,0,1,2,3,4,5,6,7,8,9,10]
- Calculate the Cross-Correlation of the two signal, and find the maximum at best_lag=-5 (最佳时延)
- It means that, we can delay Signal 1 by 5 samples, (therefore, add 5 zeros in the end of Signal 2), and get:
  - Signal 1 after delay: [0,0,0,0,0,0,0,0,0,0,1,2,3,4,5,6,7,8,9,10,0,0,0,0,0]
  - Signal 2 with zero padding: [0,0,0,0,0,0,0,0,0,0,1,2,3,4,5,6,7,8,9,10,0,0,0,0,0]
- Now add them up and get: [0,0,0,0,0,0,0,0,0,0,2,4,6,8,10,12,14,16,18,20,0,0,0,0,0]. Great! We had enlarged the signal!
- This is called time domain beamforming (时域波束形成).
- Note that best_lag=-5, with sampling rate 100 per second, the time delay is -5/100=-0.05 second: microphone 1 leads (领先) microphone 2 by 0.05 seconds.
- Now lets go to the code

The Microphone Array

- Go through the code (SNR=1000, Is_add_special_noise=0, 2 microphones)
- Test the 4 microphones case
- Test different SNR: 1000, 20, 0 (Is_add_special_noise=0, 2 microphones)
- Test ‘Is_add_special_noise’: 0, 1 (SNR=20,15,0, 2 microphones)
  - When Is_add_special_noise=1, 2 microphones, SNR=0, the real lag is -2278 but the estimated lag is -2277, although it is close, it fails to full remove the noise.

Extra's extra: localization

- Once we get the time delay (lag) between microphones 1 and 2, the distance difference between microphone 1/2 and the source (which is, $d_1 - d_2$) can be calculated.
- Then the source must be in a hyperbolic curve (双曲线)
- If there two more microphone, namely, microphones 3 and 4. the microphone pairs 1&3 and 1&4 can give two more hyperbolic curves
- The three curves can then finally locate the source correctly.
- If there enough time, will go back to this
- Please refers to H. C. So. @ http://www.ee.cityu.edu.hk/~hcso/



Advance part of Experimental Report 2 (40 points)

1. Under the default 4 microphones settings with (SNR=60, Is_add_special_noise=1) {microphone locations: (0,0)m, (25,5)m, (50,0)m and (75,5)m; source located at (1,1)m}, correctly add the signals from the 4 microphones with correct lags, explain your program with flowchart, output the figures, output the clean speech, and give analysis.

Hint:

a. you can calculate the cross-correlation between microphones 1&2, 1&3 and 1&4, and move the signal received from microphones 2,3,4 by delay some samples or adding some zeros; and then you can add them up to get a good clean speech.

b. Therefore, you should change the ‘main.m’ file to achieve this task

Advance part of Experimental Report 2 (40 points)

2. Under the default 2 microphones settings with (Is_add_special_noise=0) {microphone locations: (0,0)m and (50,10)m; source located at (1,1)m}, calculate the probability of success of ‘correct lag detection’ under at least 7 cases: SNR = -20,-10,0,10,20,30,40

文件 导航 编辑 断点

```
plot(lag_list, R_12)
title('the Cross-Correlation')
[Lag_12_value, Lag_12_index] = max(R_12);
Lag_12_estimate = Lag_12_index-(Max_lag+1);

% just to see the real lag, cannot use Real_l
% code when add the signal from different mic
Real_lag = L_TD(1)-L_TD(2);
error = Lag_12_estimate - Real_lag
%------------------------------------------
```

- The definition of ‘correct lag detection’ is:
  - If error in the figure below is 0, it is ‘correct’
  - Otherwise, it is ‘wrong’
- You should write a ‘for’ loop, randomly test 100 independent times, write down your flowchart in your experimental report.
- Then list all the results of ‘correct lag detection’ under different SNR in one table, and show your analysis.
- Is there other criterion suitable to judge the time domain beamforming method instead of ‘correct lag detection’? Give your discussion and results.

Extra part of Experimental Report 2 (+10 points)

1. Now, you are given 16 microphones, with id = 0:1:15. These microphones are located at (0,0+0.17*id)m. The source is now located at (0,10)m. Correctly add the signals from the 16 microphones with correct lags, explain your program with flowchart, output the figures, output the clean speech, and give analysis. Please discuss: is there better methods (comparing to Hint a in advance part 1) to calculate the lag?

- Parameter settings: SNR=20, Is_add_special_noise=1
- Note that the lags between microphones 1&2, 2&3, 3&4, …, (id-1)&id, …, 14&15, are the same!