ECE 345: Linear Systems and Signals
Professors Sarwate and El Rouayheb

Fall 2020
Due: 5 PM, September 25, 2020

Lab 1

To turn in this lab, you need to submit a `.zip` file containing the following three files:

- a completed lab report **in PDF format**.
- a file `lab1sim.m` containing your `lab1sim()` function, and
- a file `lab1est.m` containing your `lab1est()` function.

Points will be taken off for non-PDF submissions. Why are we being sticklers about formatting? In the real world, your work will also have to fit formatting guidelines. If you bid on a government contract, for example, and don't follow the guidelines, your bid/proposal could be rejected without review. Get in the habit of following the formatting rules!

Along those lines, do not forget to label your plots – it's part of the rubric!

1 Introduction: sound localization

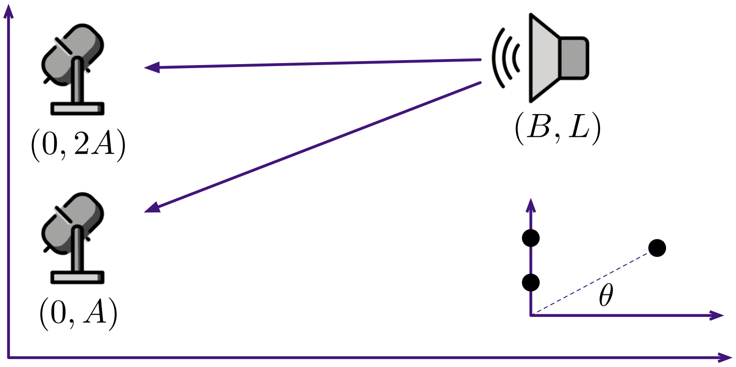

Figure 1: An illustration of a microphone array receiving a signal (image credit: Emojipedia).

We want to find multiple sensors to find the direction/location of a source in a 2D plane[1]. Suppose we have the setup in Figure 1. There are two speakers at locations $(0, A)$ and $(0, 2A)$ and a speaker at coordinates (B, L). Our goal is to identify the location of the speaker using signals measured at the microphones. If $x(t)$ is transmitted by the speaker then we can model the received signals as

$$y_1(t) = s(t - \tau_1) \tag{1}$$

$$y_2(t) = s(t - \tau_2) \tag{2}$$

for the two speakers. The delays τ_1 and τ_2 are caused by the time it takes for the signals to reach the speakers.

Note: We are doing this for audio signals but this works also for RF signals. Systems like radar use antenna arrays to find the direction of arrival (DoA) of received signals.

[1] Some of this lab is based labs from DSP First at Georgia Tech.

2 The Lab

2.1 Part 1: simulating the received signals

In the first part of this lab we will focus on simulating the signals that appear in this problem: the output of the microphone and the received signals at the two microphones. We're going to ignore the affect of *path loss* in this simulation, which is the volume (amplitude) that is lost when traveling over a distance.

(a) The speed of sound is $c_s = 333\frac{1}{3}$ m/s. Calculate formulas for the delays τ_1 and τ_2 that it takes for the sound to reach each of the two microphones. Your answer should be in terms of A, B, and L.

(b) Suppose that the transmitted signal $s(t) = 1000\cos(880\pi t)u(t)$. What is the frequency of this sinusoid in Hertz? Using the `sig = @(t) f(t)` way of defining a function/signal, write a function that generates $s(t)$. Don't forget the $u(t)$!

(c) Let $A = 0.5$ meters and $B = 100$ meters. Set $L = 100$ meters. Simulate sending this $s(t)$ over the channel using the sampling rate $f_s = 44.1$ kHz. Use `subplot` to plot $y_1(t)$ and $y_2(t)$ for $t \in [\min(\tau_1, \tau_2) - 0.01, \max(\tau_1, \tau_2) + 0.01]$.

(d) Write a function `lab1sim(A,B,L,sig)` that produces as an output two functions (using `sig = @(t) f(t)`) called `y1sig` and `y2sig` that simulate $y_1(t)$ and $y_2(t)$. You can verify that it works by reproducing the plot from the previous part.

2.2 Part 2: estimating the location

Next we will look at how we can use the relative delay between the two received signals to get an estimate of θ and L. This will involve a little bit of trigonmetry.

(a) It turns out that if the receiver doesn't know $s(t)$ then finding both τ_1 and τ_2 is not really possible. Explain why this is in words.

(b) We are really only able to estimate the *relative time shift* between the two signals. Use `lab1sim` to generate $y_1(t)$ and $y_2(t)$ for $t \in [0, 0.5]$. Then use the `xcorr` function to plot the cross-correlation between $y_1(t)$ and $y_2(t)$. If you take `[C,lags]` as the output of `xcorr` then `lags` will have the *integer* lag at which the cross correlation is computed. To convert this back to time, take `lags/fs` to account for the sampling rate and plot `C` versus `lags/fs` to visualize the cross correlation. Zoom in to find the peak and estimate the relative time shift.

(c) If A is relatively small and the distance between each microphone and the speaker is large [2] then the relative time shift and the angle θ are related by the (approximate) formula

$$\frac{A}{c_s}\sin(\theta) = \tau_1 - \tau_2. \tag{3}$$

Use this formula to find a formula for L in terms of A, B, c_s, and $\tau_1 - \tau_2$. You may wish to estimate θ first and then estimate L from that.

(d) Compare the L you calculated from the formula with the $L = 100$ meters that we set above. Are they exactly the same or different? If they are different, why are they different?

[2] You may have seen something similar when analyzing double-slit experiments.

2.3 Part 3: automating the estimates

Once we have a process for doing the estimation we would like to automate it. After all, who wants to find the peaks in the cross-correlation by hand? For this part continue generating signals only for $t \in [0, 0.5]$.

(a) Now we would like to automate the process of finding the delays. Use the `max` function to find the peak in the autocorrelation and convert the corresponding lag into an estimate of the relative time shift.

(b) Write a MATLAB function `lab1est(A,B,y1,y2)` that outputs both the estimated angle and L given the model parameters and the received signals.

(c) Use `linspace` to generate 100 evenly spaced points spaced between $L = 1$ meter and $L = 100$ meters. Write a loop that generates signals from a speaker at location L using `lab1sim()` and then estimates L using `lab1est()`. Plot the relative error between the true L and the estimate (which we can call $\hat{L}$):

$$\varepsilon_{\text{rel}} = 1 - \frac{\hat{L}}{L} \tag{4}$$

Plot ε_{rel} as a function of L. Explain the general trend that you see (there may be some oscillations).

2.4 Part 4: impact of noise

Real systems always have noise so it's important to investigate how *robust* the estimator you have is to the presence of noise in your system. We are going to use additive white Gaussian noise (AWGN), which means that the noise is made of of independent and identically distributed Gaussian $\mathcal{N}(0, 1)$ random variables. We can scale the power of the noise (variance) by multiplying the noise by a scalar gain. For this part continue generating signals only for $t \in [0, 0.5]$ and $L = 100$.

(a) Now let's see what happens when we add a little noise. Generate two independent random vectors of unit-variance Gaussian noise `w1` and `w2` using `randn`. These correspond to $w_1(t)$ and $w_2(t)$, two noise signals added in to make new received signals:

$$z_1(t) = y_1(t) + \alpha w_1(t) = s(t - \tau_1) + \alpha w_1(t) \tag{5}$$

$$z_2(t) = y_2(t) + \alpha w_2(t) = s(t - \tau_2) + \alpha w_2(t) \tag{6}$$

Run the same test as in (d) of Part 2 with $L = 100$. Generate $y_1(t)$ and $y_2(t)$ using the outputs of `lab1sim` but add the noise to get $z_1(t)$ and $z_2(t)$. Use $z_1(t)$ and $z_2(t)$ as inputs to `lab1est` instead of $y_1(t)$ and $y_2(t)$. Check what the difference between the estimated and true L is with $\alpha = 1$ and $\alpha = 10^2$. Do you see any difference in the errors?

(b) Use `linspace` to generate 10 α values between 10. and 150. For each value of α, generate $N = 100$ noisy received signals and produce 100 estimates $\hat{L}_1(\alpha), \hat{L}_2(\alpha), \ldots, \hat{L}_N(\alpha)$. Then compute the average squared error

$$\varepsilon = \frac{1}{N} \sum_{i=1}^{N} (\hat{L}_i(\alpha) - L)^2 \tag{7}$$

Plot ε as a function of α. What general trend do you notice as the noise power increases? Note: you can do this as a two nested loops over α and $i = 1$ to N.

(c) Explain where in your receiver program that the noise may cause errors in the estimate. Note: if you increase the noise power further you may start to get instability and the estimator will return complex numbers. Try to think of why that might happen.

2.5 Part 5: the bigger picture

Now we're going to ask you to think a more broadly about this problem setup. For the last sections you have dug into the implementation of an estimator, but once you understand some of those tradeoffs it's important to think about other tradeoffs that you could also explore.

(a) Think about changing the spacing A between the two antennas. How will the accuracy of your location estimates vary as you decrease the spacing A? Explain in words what you think will happen and why. If you want you can simulate what happens to support your claims.

(b) Suppose we added a third microphone at $(0, 3A)$. Explain in words how this would affect the performance of your estimator and why.

(c) Suppose you had a third microphone but could place it anywhere. Explain in words what new capabilities you could build into the system and why.