

NLOS Propagation

Introduction

In this lab we will measure the amplitude of a signal propagating over an indoor NLOS path. You will compare the probability distribution of the amplitude of the received signal to the Rayleigh distribution. You will also plot the average signal strength as a function of distance and fit a power law model.

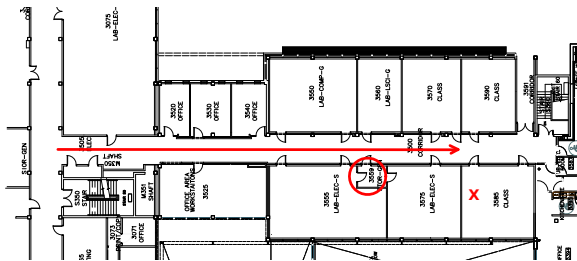
Indoor propagation involves refraction, reflection, and transmission mechanisms similar to those in outdoor areas but by making measurements indoors we avoid the risk of getting wet.

Since in-person labs are not possible this term, this version of the lab does not include data collection. Students will analyze data collected by the instructor. The data collection procedure is described in an appendix.

Data Files

Two data files are supplied on the course website. Each contains samples of the the magnitude of an unmodulated signal at a carrier frequency of 222 MHz ($\lambda = 1.35$ m). The sampling rate in each case was approximately 1 kHz with a duration of about 1 minute. There are 4 bytes per samples so each file contains about 240 kBytes.

The floor plan below shows the portion of the third floor of SW1 where the measurements were made.

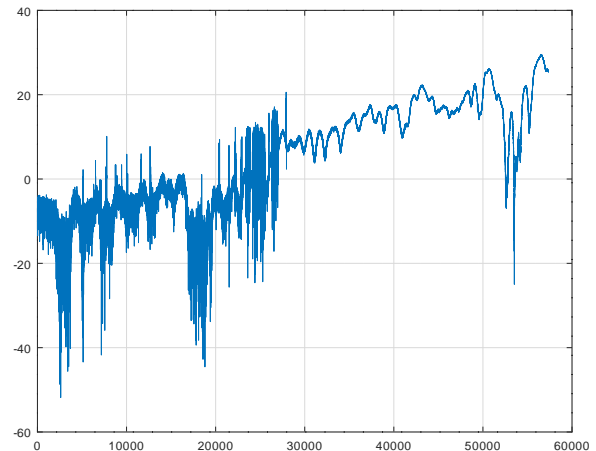
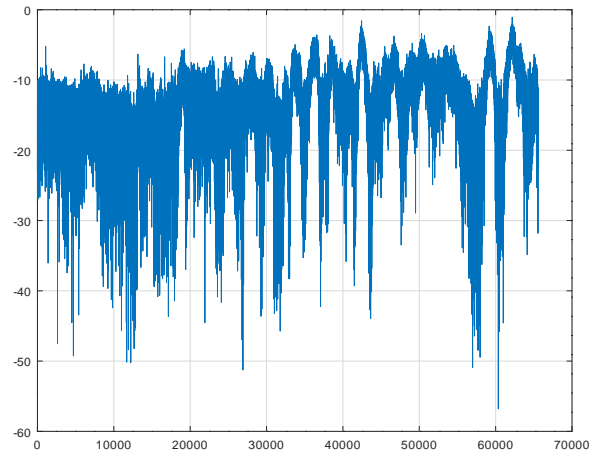


The X marks the location of the transmitting antenna.

The circle shows the location (a storage room) where the first data set was collected. The receiving antenna was moved around the room, a space of about 2 m by 1.5 m, 10 or 20 times.

The arrow shows the path used to collect the second data set. The receiving antenna was carried, at an approximately constant rate, down the corridor towards the transmitter.

Plots of the two data sets showing signal amplitude (dB) vs time (milliseconds), are shown below:



Data Analysis

You will check whether the collected data fits theoretical predictions for multipath fading and path loss versus distance.

The examples below use Octave (mostly compatible with Matlab) but you may use other numerical analysis software if you prefer.

The data files contain sequences of 4-byte (“single precision”) floating point values. The following Oc-

tave commands read the values into the vector **s** and plot them¹ as a sanity check:

```
f=fopen('d:/propdata2.dat','r') ;
s=fread(f,Inf,'float') ;
fclose(f) ;
```

```
plot(20*log10(s(2:end))) ;
```

If your data includes points that you do not want to include in your analysis (e.g. because you were not moving) you can use different starting and ending indices of the vector **s**. In the case of data collected by the instructor you can review the plots above and decide which parts of the data may not be representative of the effects you are trying to analyze.

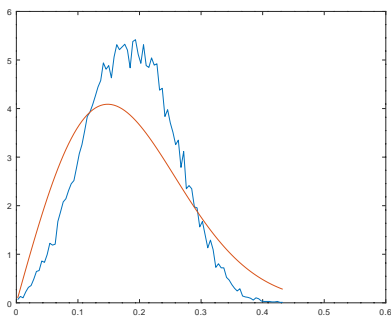
You can use `hist()` to compute the bin counts and bin centers of the histogram and `raylpdf()` to compute the theoretical pdf:

```
[n,x]=hist(s,100) ;
dx=x(2)-x(1) ;
h=n/sum(n)/dx ;
plot(x,h) ;
```

```
sigma=mean(s)/1.25
```

```
plot(x,[h;raylpdf(x,sigma)])
```

Which might result in something like the following (not very good²) fit:



If your data is not a good fit to a Rayleigh distribution you can compute the mean and variance and plot the corresponding Ricean distribution to see if it's a better fit.

A common model for distance-dependent path loss is that the received signal power drops off as a

¹Skipping the first sample – it seems to have a glitch

²This data looks a better fit to a [Ricean](#) distribution.

power of the distance. In free space (as given by the Friis equation) the path loss exponent is -2 . For NLOS propagation we would expect that the signal will drop off with distance more quickly.

If we express the received signal power as a function of distance:

$$P_R = kd^{-n}$$

then the received power in dB will be a linear function of the log of distance:

$$P_{R(\text{dB})} = -n10 \log(d) + k_{\text{dB}}.$$

This is a straight line with a slope $-10n$. By fitting a straight line to a plot of the received power in dB versus $10 \log(d)$ you can estimate n . *Hint: You might find the function `polyfit()` useful.*

Lab Report

Submit a lab report to the appropriate Assignments dropbox in the file format described in the course information document. In addition to the required identification information, include the following in your report:

- The histogram of your multipath data and a superimposed Rayleigh or Ricean pdf that has the same parameters as the data. Give the parameters (σ and possibly μ) and show how you obtained them (e.g. the Matlab or other commands and any calculations).
- The plot of the path loss versus distance (or suitable functions thereof) along with the best straight-line fit to a power law. Give the path loss exponent and show you obtained it.

You do not need to include other material (introduction, conclusions, etc.) unless you want to.

Appendix - Procedure

Overview

The instructor will connect a signal generator with an unmodulated³ output at a frequency⁴ of 222.28 MHz ($\lambda = 1.35$ m) to an “antenna” consisting of clip leads.

³Sometimes called “CW” for continuous wave.

⁴This frequency is in an amateur radio band which your instructor is licensed to use (“call sign” VA7EX).

You will use a [LimeSDR Mini](#) board that uses a Lime Microsystems [LMS7002M](#) IC to downconvert and sample an RF signal to a digitized complex base-band signal. The sampled signal is read over a USB 3.0 interface.

GNU Radio is a software package that can be used to prototype the signal processing required to implement a software-defined radio (SDR). The instructor will supply a GNU Radio flow graph, `survey.grc`, that selects the desired signal, computes the amplitudes of the complex samples and saves them to a file.

You will collect two files of received signal amplitude versus time. You will move antenna during the measurements so that the samples also correspond to signal amplitude versus position.

First you will look for a location where the signal level has large variations over short distances. move the antenna about randomly to sample a volume of a few wavelengths per side (the USB cable is about 2m long). For this measurement the distance to the transmitter (d) will not change significantly. If there are multiple propagation paths to the location where you are taking your measurements, the interference between the paths will cause constructive and destructive interference (fading). These measurements will allow you to verify models for the probability distribution of signal amplitude due to multipath fading.

Then you will walk the antenna from a distance of about $d = 3$ m from the transmitter to a distance about $d = 30$ m from the transmitter (or vice-versa). If you average out the short-term fluctuations due to multipath, this measurement will allow you to (roughly) model the variation of signal strength with distance in an indoor environment.

This data collection portion of the lab should not take very long so if you do not have a laptop with GNU Radio installed you may use someone else's laptop to collect the data and then post-process it yourself on the lab computers. If you work in pairs you should collect two sets of data so that each student has their own data to analyze.

Once you have collected the data you will compute the histogram of the short-term variations of signal amplitude and compare it to a Rayleigh distribution. You will also fit a curve to the the amplitude vs distance data.

It's a good idea to plot your data before leaving the lab as it's common for things to go wrong.

Install the Software

If you have a Windows laptop with about 500 Mb of free disk space you can install GNU Radio and the drivers for the LimeSDR board.

The process is as follows:

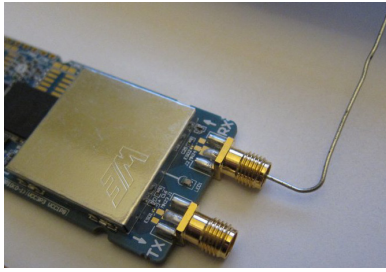
- download the `PothosSDR-2018.11.04-vc14-x64.exe.zip` file from the course web site. Run the PothosSDR installer (the `.exe` file) and accept the default options.
- copy the folders (`bin`, `lib` and `share`) in the PothosSDR folder in the above zip file into the `C:\Program Files\PothosSDR` folder. This will merge the LimeSDR drivers for GNU Radio into the installation.
- download and install Python 2.7.15 from <https://www.python.org/downloads/release/python-2715/>, again accepting all default options⁵.
- run GNU Radio Companion from the Windows menu; it will proceed to download and install additional Python packages.
- if the LimeSDR device is not recognized when you plug it into your USB port you may have to download and install FTDI drivers from: <http://www.ftdichip.com/Drivers/D3XX.htm>

Connect the SDR Module

Connect a Software Defined Radio (SDR) module to a USB 3.0 cable and the connect cable to a USB (USB 3.0, if possible) port of a laptop. If the device is not recognized you may have to install the FTDI USB drivers as described above. The LED should start blinking red/green.

Use the supplied antenna or make one as shown below (open up a paper clip, plug one end into the RX SMA port as shown below and hold it in place with electrical tape). The module will have a case; the receive port is indicated by a triangle pointing inwards.

⁵Unfortunately, GNU Radio still uses Python 2.7.



You may want to run the LimeSuite utility from the Windows Menu to verify the connection to the SDR board. Select Options/Connection Settings, select the LimeSDR Mini device, and click on connect. If you now click on the buttons Defaults and Calibrate All, the device will take a few seconds to calibrate itself and then show new DC and gain offsets. This verifies the board is connected and functional.

Run the Propagation Measurement Flow Graph

Download the `survey.grc` file from the course web site and open it. If you've installed GNU Radio, it should open with the GNU Radio Companion application.

Select Run/Execute (F6) to run the flow graph. After a few seconds this will open up a GUI that shows you the received signal as a function of time and frequency at two sampling rates (5 MHz and approximately 1 kHz) as well as a polar plot of the complex signal and the signal amplitude versus time in dB. The GUI also has controls to adjust the receive (actually, LO) frequency, the gains of various receiver stages and the name of the file being (over-)written, `D:/propdata.dat` by default.

Normally the only control you will need to change is the file name. If you don't have a `D:` drive you should change the default file name value by double-clicking the QT GUI Entry box in the flow graph and changing the Default Value value. You should do this before running the flow graph.

If you are in range of the transmitter you should see impulses in frequency, sine waves in time and the phasor diagram should be a point rotating around the origin and moving in and out depending on the signal amplitude. We will cover GNU Radio and signal processing in more detail later in the course.

Collect Data

Rayleigh Fading

Move to a location where the signal is likely to be received over multiple paths (e.g. another room with a door that opens up to the same hallway as the lab) and where the signal is significantly stronger than the noise. There should be strong variation of signal level with antenna position (e.g. you might find nulls a quarter-wavelength away from a reflecting object). This confirms the existence of multipath propagation.

Change the file name (e.g. `propdata2.dat`) to start collecting data to a new file and move the antenna around to uniformly sample the signal strength within a volume of a few wavelengths (a few metres) cubed. Collect data for about a minute. At a sampling rate of about 1 kHz this will give you about 60k samples, and a file approximately 240 kB large. Close the GUI window to stop the flow graph and stop collecting data.

Path Loss

For the second measurement you will repeat the collection process above but moving at an (approximately) constant pace down the hallway away from (or towards) the lab.

Start at a distance where the signal level is not saturating the receiver (i.e. the level changes with distance). Change the file name (e.g. `propdata3.dat`) and walk down the hall at a steady rate until the signal disappears into the noise. Close the GUI window to stop collecting data.