ECE 738 Introduction to Signal Processing

Matlab Project II

Fall 2007

Issued: Monday, November 5, 2007 **Due:** Wednesday, November 21, 2007

The goal of this exercise is to use time delay beamforming to analyze linear frequency-modulated pulses received on a vertical array. This project will allow you to explore some of the issues associated with implementing a time-delay beamformer. You will apply the Matlab function you develop to process a simulated reception.

Your report should include any analytical (*i.e.,* pencil/paper) work, Matlab plots and code, and relevant explanations for all parts of the project. Please write a report separate from the Matlab code itself. You are certainly encouraged to include comments in your Matlab code, but I will not consider the comments part of your official report. On the due date, please provide me with a hard copy of your report and an electronic copy of the Matlab function you wrote. (I will test your algorithm by running it on a data set different from the one provided on the course website.)

Background

Please review Project I in which you implemented a complex demodulator and matched filter for linear frequency modulated pulses. You will need the code your wrote for that project to preprocess the simulated data for this project.

Project Definition

In this project you will implement a time-delay beamformer for the linear frequency-modulated (LFM) signals contained in the Matlab file on the course website. The file contains the following variables:

`p`	`Ntimes` $\times$ `Lrcvr` matrix of array data
`fs`	Sample frequency (Hz)
`z`	Vector of depths of the receivers (length = `Lrcvr`)
`taxis`	Time axis corresponding to `p` (length = `Ntimes`)
`c0`	Sound speed of medium (m/s)

Since the file is rather large, I have also made it available as 5 sub-files (which each contain part of the data set). These sub-files contain the channels of the `p` matrix, *e.g.*, `lfmsim_ch1_10.mat` contains the first 10 channels of the data set. The LFM signals in this data set have parameters identical to that in Project I. Each signal is 135 seconds long. The source sweeps from 225 Hz to 325 Hz, thus the center frequency is 275 Hz, and the total bandwidth is 100 Hz.

A. Preprocess the signals using matched filter function you developed for Project I. The arriving signals should be visible on the array after demodulation and matched filtering. Note the matched-filtered signals will occupy a relatively small time window (as compared the original signal). I recommend saving the data in that narrow time window (2 seconds should be sufficient) in a separate file for processing. This will save you from having to beamform the whole data set.

B. Write a program to compute the beampattern for the array as a function of frequency (over the bandwidth of the source). Plot the beampattern. Please include several a 3-d beampattern as function of frequency and angle in your report, along with several "cuts", *i.e.*, beampatterns at a single frequency. Your program should allow for the use of any of the standard Matlab window functions. You should be able to do some simple analytical calculations to predict what the beampattern looks like, and the resolution of the array at different frequencies. Make sure your Matlab beampatterns agree with your analytical calculations.

C. Write a program to implement a time-delay beamformer for this array using quantized delays. This function should allow the user to specify any of the standard window functions to use as a taper. Which beams can be computed exactly? Which beams will be affected by quantization of the delays?

D. Apply your beamforming program to the data contained in the file `lfmsim.mat`. What are the arrival times and angle-of-arrivals of the signals incident on this array? Note: when computing the arrival time you should account for any delay through the filters you used in the pre-processing stage.

E. Suppose that you only use every other sensor of the array to do the the beamforming. Based on the beampattern and your analytical calculations, what do you expect the results will be? Does your Matlab result agree with your predictions? Explain your results.

F. After demodulation the signal occupies a smaller bandwidth, thus it could be decimated (converted to a lower sample rate) without significant loss of information. In this part you will explore what effect a lower sample rate has on your time-delay beamformer. Use the `decimate` command to lower the sample rate of your data and then process the new data with your time-delay beamformer. What, if any, effects do you observe?