ECE 437 Lab #7(MATLAB problem) Winter 2009

The purpose of this lab is to explore two examples of applications of cross-correlation function, $\rho_{xy}(k)$, which is simply the cross-correlation of two signals, x(n) and y(n-k), where k is called the lag value. For detailed explanation, please read the attached description of cross-correlation.

1. Range estimation of targets (using a Radar)

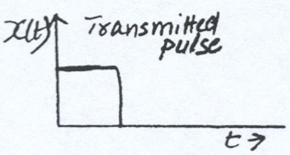
The basic idea of range estimation involves transmitting a pulse, detecting an echo from a target using cross-correlation technique, and estimating its range as,

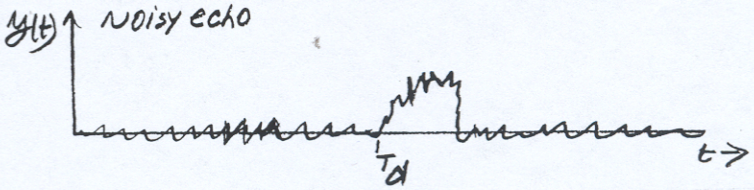
$$R = cT_d/2$$

where

c = velocity of electromagnetic waves = $3\text{x}10^8$ m/sec, and
T_d = Time delay between the transmitted pulse and the received echo.

The problem of range detection becomes a challenging one in presence of random noise, which is always encountered (due to both natural and man-made causes). Most of the signal samples received in a Radar receiver are actually random noise samples, whereas the true echoes from targets are the desired signals buried in noise. Since a deterministic signal, such as a transmitted Radar pulse, and random noise are uncorrelated with each other, their cross-correlation will be zero for most of the samples, except at locations where true echoes are present. Thus, the cross-correlation of the received signal and the transmitted pulse can reveal the locations of the true echoes. The following picture illustrates the idea.





<u>What you need to do</u>

1. Assume a 11-point Barker sequence is used as the transmitted pulse:

B(n) = [+1 +1 +1 -1 -1 -1 +1 -1 -1 +1 -1].

Generate a vector, x, of length 2000 that contains the above sequence at sample locations, $1 \leq n \leq 11$, and zeros elsewhere.

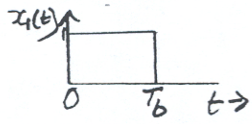
2. Now load the given data file, lab709dat1.mat, that contains a noisy received signal, y(n), of length 2000.

3. First compute the cross-correlation of lag = 0 between x(n) and y(n). Next, slide x(n) over y(n) one sample at a time to simulate different lag values, $1 \leq k \leq 1999$, and for each k, compute the cross-correlation, corr(k), between x(n-k) and y(n).

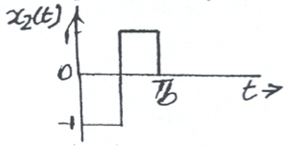
4. Plot corr(k) as a function of k, and locate the targets.

2. Detection of digital data using a Correlation Receiver

Consider the baseband transmission of a binary bit stream consisting of 1s and 0s, where each bit is represented by a pulse of duration, T_b. Suppose a "1" is represented by the following pulse, x_1:



and a "0" is represented by the following one, x_2:



Since the received signal is usually noisy, we can compute cross-correlation between the received signal, y(t) and the above templates at intervals of T_b. Suppose r_{1y} denotes the cross-correlation between x_1 and y(t), and r_{2y} denotes the cross-correlation between x_2 and y(t). A correlation receiver decides between 0 or 1 by examining the values of r_{1y} and r_{1y} as follows:

If $r_{1y} > r_{2y}$, decide the bit to be a "1", and

if $r_{2y} > r_{1y}$, decide the bit to be a "0".

The above decision rule is called an optimum detector that minimizes the probability of error. This will be discussed later in the class.

<u>What you need to do</u>

1. Assume that each 1 and 0 is represented by the following discretized pulses of duration 4:

A "1" is represented by x_1 = [1 1 1 1], and
a "0" is represented by x_2 = [-1 -1 1 1];

Also, assume T_b = 4 samples.

2. Load the data file lab709dat2.mat that contains two noisy received pulse sequences, y1(n), y2(n), one of which more noisy than the other. For each pulse sequence, compute cross-correlations, r_{1y} and r_{1y}, at sample intervals of 4, and decode the received bit stream using the decision rule mentioned above. You should find two bit streams, each containing 25 bits.