Lab 09 – Discrete-Time Random Processes: 2

1. Objectives

This is the second lab in the series of two labs that cover some basic methods of analyzing random processes, culminating with the application of these methods to the radar detection problem. Note that these labs assume an introductory background in probability theory.

2. Autocorrelation of Discrete-Time Random Processes

In this section, we will generate *discrete-time random processes* and then analyze their behavior using the correlation measures introduced in the previous lab.

2.1 Background:

A *discrete-time random process* X_n can be considered a sequence of random variables. So, for each n, X_n is a random variable.

The *autocorrelation* is an important function for characterizing the behavior of random processes. If X is a *wide-sense stationary* (WSS) random process, the autocorrelation is defined by

$$r_{XX}(m) = E[X_n X_{n+m}] \qquad m = \ldots, -1, 0, 1, \ldots . \qquad (1)$$

Note that for a WSS random process, the autocorrelation does not vary with n. Also, since $E[X_n X_{n+m}] = E[X_{n+m} X_n]$, the autocorrelation is an even function of the "lag" value m.

Intuitively, the autocorrelation determines how strong a relation there is between random variables separated by a lag value of m. For example, if X is a sequence of independent identically distributed (i.i.d.) random variables each with zero mean and variance σ^2, then the auto-correlation is given by:

$$
\begin{aligned}
r_{XX}(m) &= E[X_n X_{n+m}] \\
&= \begin{cases} E[X_n]E[X_{n+m}] & \text{if } m \neq 0 \\ E[X_n^2] & \text{if } m = 0 \end{cases} \\
&= \sigma_X^2 \delta(m) \ .
\end{aligned}
$$

We use the term *white* or *white noise* to describe this type of random process. More precisely, a random process is called *white* if its values X_n and X_{n+m} are uncorrelated for every $m \neq 0$.

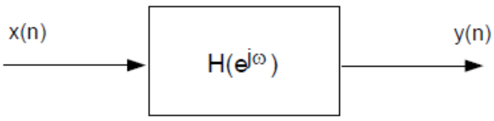


Figure 1: A new LTI system diagram

If we run a white random process X_n through an LTI filter as in figure 1, the output random variables Y_n may become correlated. In fact, it can be shown that the output autocorrelation $r_{YY}(m)$ is related to the input autocorrelation $r_{XX}(m)$ through the filter's impulse response $h(m)$.

$$r_{YY}(m) = h(m) * h(-m) * r_{XX}(m) \qquad (2)$$

2.2 Experiment:

Consider a white Gaussian random process X_n with mean 0 and variance 1 as input to the following filter.

$$y(n) = x(n) - x(n-1) + x(n-2) \qquad (3)$$

Calculate the theoretical autocorrelation of Y_n using (1) and (2). Show all of your work.

$We\ have:$

$h(n) = \delta(n) - \delta(n-1) + \delta(n-2)$

$r_{XX} = \sigma_X^2$

$Then:$

$$r_{YY} = \big(\delta(m) - \delta(m-1) + \delta(m-2)\big) * (\ \delta(-m) - \delta(-m-1) + \delta(-m-2)\ ) * \sigma_X^2\ \delta(m)$$

$$r_{YY} = \left(\sum_{\lambda=-\infty}^{\lambda=\infty} \big(\delta(m-\lambda) - \delta(m-1-\lambda) + \delta(m-2-\lambda)\big) \times (\ \delta(-m) - \delta(-m-1) + \delta(-m-2)\ ) \right) * \sigma_X^2\ \delta(m)$$

Performing convolution graphically, we get:

$$r_{YY} = \delta(n+2) - 2\,\delta(n+1) + 3\,\delta(n) - 2\delta(n-1) + \delta(n-2)$$

Generate 1000 independent samples of a Gaussian random variable X with mean 0 and variance 1. Filter the samples using (3). We will denote the filtered signal Y_i, i = 1, 2, . . . , 1000.

Draw 4 scatter plots using the form subplot(2,2,n), (n = 1, 2, 3, 4). The first scatter plot should consist of points, (Yi, Yi+1), (i = 1, 2, . . . , 900). Notice that this correlates samples that are separated by a lag of "1". The other 3 scatter plots should consist of the points (Yi, Yi+2), (Yi, Yi+3), (Yi, Yi+4), (i = 1, 2, . . . , 900), respectively. What can you deduce about the random process from these scatter plots?

For real applications, the theoretical autocorrelation may be unknown. Therefore, $r_{YY}(m)$ may be estimated by the *sample autocorrelation*, $r'_{YY}(m)$ defined by

$$r'_{YY}(m) = \frac{1}{N-|m|} \sum_{n=0}^{N-|m|-1} Y(n)Y(n+|m|) \quad -(N-1) \leq m \leq N-1 \tag{4}$$

where n is the number of samples of Y.

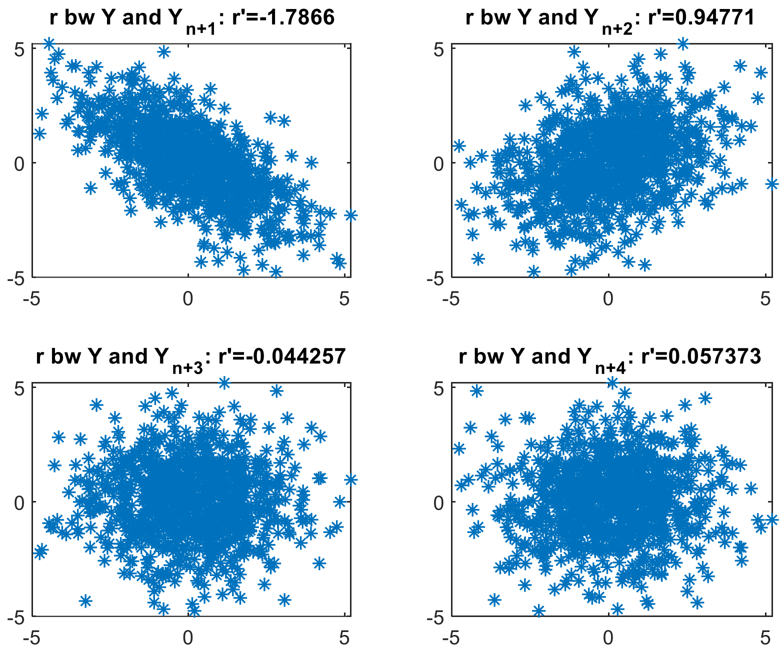
Use MATLAB to calculate the sample autocorrelation of Y using (4). Plot both the theoretical autocorrelation $r_{YY}(m)$ and sample autocorrelation $r'_{YY}(m)$ for -10 ≤ m ≤ 10. Use *subplot* to place them in the same figure. Does equation (4) produce a reasonable approximation of the true autocorrelation?

Task 1:

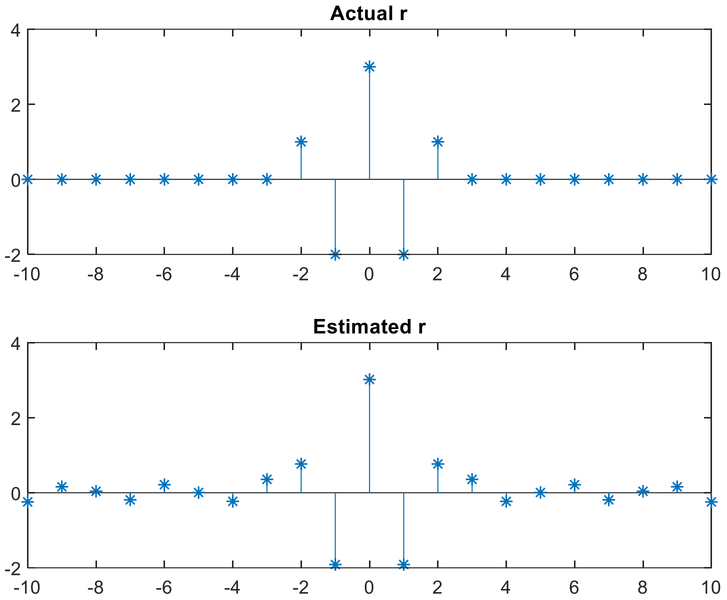
For the filter in Equation (3):

i. Show your derivation of the theoretical output autocorrelation, $r_{YY}(m)$.

ii. Submit the four scatter plots. Label each plot with the corresponding theoretical correlation, using $r_{YY}(m)$. What can you conclude about the output random process from these plots?



iii. Submit your plots of $r_{YY}(m)$ and $r'_{YY}(m)$ for $-10 \leq m \leq 10$. Does equation (4) produce a reasonable approximation of the true autocorrelation? For what value of m does $r_{YY}(m)$ reach its maximum? For what value of m does $r'_{YY}(m)$ reach its maximum.



At m = 0, both are maximum.

iv. Submit your MATLAB code.

```
ITER = 1000;

x = randn(1, ITER);
a = 1;
b = [1 -1 1];
y_delay = zeros(4, ITER);
y = filter(b, a, x);
m = [1 2 3 4];

%estimate auto correlation
r_est = autocorrelation_(y,m, ITER);

close all;
for ii=1:4
    subplot(2, 2, ii)
    y_delay(ii, :) = [y(1, m(ii)+1:end) zeros(1, m(ii))];
    plot(y,y_delay(ii, :) , '*')
    title('r bw Y and Y_{n+'+string(ii)+'}'+": r'="+string(r_est(1, ii)))
end

%actual r:
n = -10:10;
r = [zeros(1, 8) 1 -2 3 -2 1 zeros(1, 8)];
m = -10:10;
r_est = autocorrelation_(y,m, ITER);

subplot(2, 1, 1)
stem(n, r, '*')
title('Actual r')
subplot(2, 1, 2)
stem(n, r_est, '*')
title('Estimated r')
```

```
function [r_est] = autocorrelation_(y,m, ITER)
r_est = zeros(1, length(m));
for ii = 1:length(m)
    for jj = 1:ITER-abs(m(ii))
        r_est(1, ii) = r_est(1, ii) + y(jj)*y(jj+abs(m(ii)));
    end
    r_est(1, ii) = (1/(ITER-abs(m(ii))))*r_est(1, ii);
end

end
```

3. Correlation of Two Discrete-Time Random Processes

In this section, we will study the concept of cross-correlation of two discrete-time random processes and its application in a radar detection scenario.

3.1 Background

The *cross-correlation* is a function used to describe the correlation between two separate random processes. If X and Y are jointly WSS random processes, the cross-correlation is defined by

$$c_{XY}(m) = E[X_n Y_{n+m}] \quad m = \ldots, -1, 0, 1, \ldots . \tag{5}$$

Similar to the definition of the sample autocorrelation introduced in the previous section, we can define the *sample cross-correlation* for a pair of data sets. The *sample cross-correlation* between two finite random sequences X_n and Y_n is defined as:

$$c'_{XY}(m) = \frac{1}{N-m} \sum_{n=0}^{N-m-1} X(n)Y(n+m) \quad 0 \leq m \leq N-1 \tag{6}$$

$$c'_{XY}(m) = \frac{1}{N-|m|} \sum_{n=|m|}^{N-1} X(n)Y(n+m) \quad 1-N \leq m < 0 \tag{7}$$

where N is the number of samples in *each* sequence. Notice that the cross-correlation is not an even function of m. Hence a two-sided definition is required.

3.2 Application to Radar/Sonar Scenario

Cross-correlation of signals is often used in applications of sonar and radar, for example to estimate the distance to a target. In a basic radar set-up, a zero-mean signal $X(n)$ is transmitted, which then reflects off a target after traveling for $D/2$ seconds. The reflected signal is received, amplified, and then digitized to form $Y(n)$. If we summarize the attenuation and amplification of the received signal by the constant α, then

$$Y(n) = \alpha X(n-D) + W(n) \qquad (8)$$

where $W(n)$ is additive noise from the environment and receiver electronics.

In order to compute the distance to the target, we must estimate the delay D. We can do this using the cross-correlation. The cross-correlation c_{XY} can be calculated by substituting (8) into (5).

$$\begin{aligned} c_{XY}(m) &= E[X(n)Y(n+m)] \\ &= E[X(n)(\alpha X(n-D+m) + W(n+m))] \\ &= \alpha E[X(n)X(n-D+m)] + E[X(n)]E[W(n+m)] \\ &= \alpha E[X(n)X(n-D+m)] \end{aligned}$$

Here, we have used the assumptions that $X(n)$ and $W(n+m)$ are uncorrelated and zero-mean. By applying the definition of autocorrelation, we see that

$$c_{XY}(m) = \alpha r_{XX}(m-D) \qquad (9)$$

Because $r_{XX}(m-D)$ reaches its maximum when $m = D$, we can find the delay D by searching for a peak in the cross correlation $c_{XY}(m)$. Usually the transmitted signal $X(n)$ is designed so that $r_{XX}(m)$ has a large peak at $m = 0$.

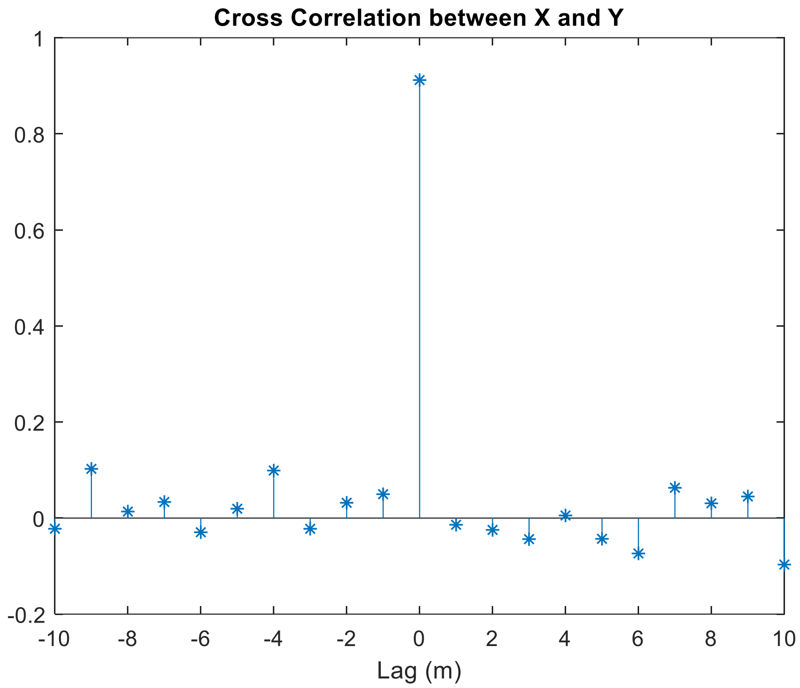
3.3 Experiment:

Using (6) and (7), write a MATLAB function C=CorR(X,Y,m) to compute the sample cross-correlation between two discrete-time random processes, X and Y, for a single lag value m.

To test your function, generate two length 1000 sequences of zero-mean Gaussian random variables, denoted as X_n and Z_n. Then compute the new sequence $Y_n = X_n + Z_n$. Use *CorR* to calculate the sample cross-correlation between X and Y for lags $-10 \le m \le 10$. Plot your cross-correlation function.

Task 2:

i. Submit your plot for the sample cross-correlation between *X* and *Y*. Label the *m*-axis with the corresponding lag values.



ii. Which value of *m* produces the largest cross-correlation? Why?

Lag of 0 produces the largest correlation as both X and Y are dependent on each other. Hence max correlation occurs when they overlap without any delay.

iii. Is the cross-correlation function an even function of *m*? Why or why not?

As is being observed, the function is not symmetric about y axis. Hence it is not even. This is due to the fact that the cross correlation itself is not symmetric.

iv. Submit the MATLAB code for your CorR function.

```
function [C] = CorR(x,y, m)
    ITER = length(x);
    C = 0;
    if (m<0)
        for jj = abs(m)+1:ITER
            C = C + x(jj)*y(jj+m) ;
        end
        C = (1/(ITER-abs(m)))*C;
    else
        for jj = 1:ITER-m
            C = C + x(jj)*y(jj+m) ;
        end
        C = (1/(ITER-m))*C;
    end
```

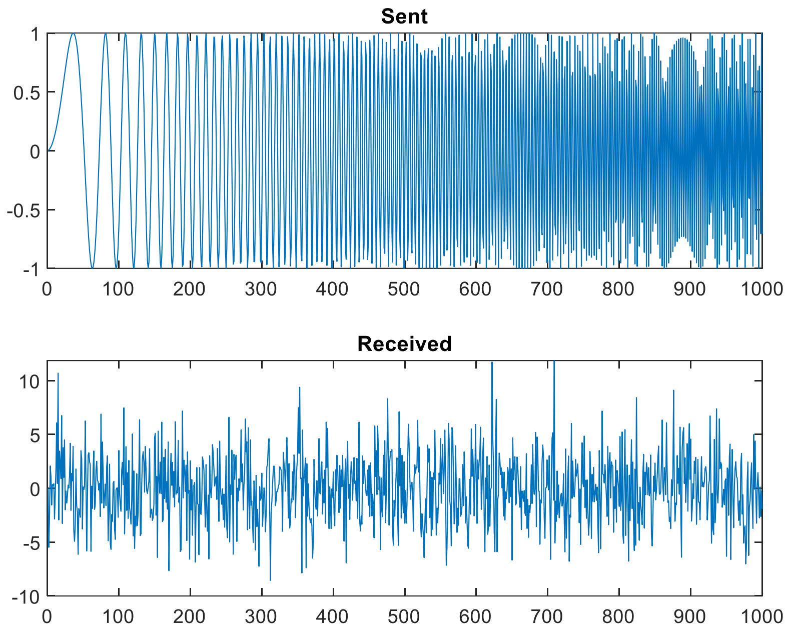
```
end
```

Next, we will illustrate how cross-correlation can be used to measure time delay in radar applications. Download the MAT file *radar.mat* from CANVAS assignment for this lab and load it using the MATLAB command *load radar*. The vectors *trans* and *received* (in the loaded *radar.mat* file) contain two signals corresponding to the transmitted and received signals for a radar system. First, compute the sample autocorrelation of the signal *trans* for the lags $-100 \leq m \leq 100$. (Hint: Use your *CorR* function.)

Next, compute the sample cross-correlation between the signal *trans* and *received* for the range of lag values $-100 \leq m \leq 100$, using your *CorR* function. Determine the delay D.

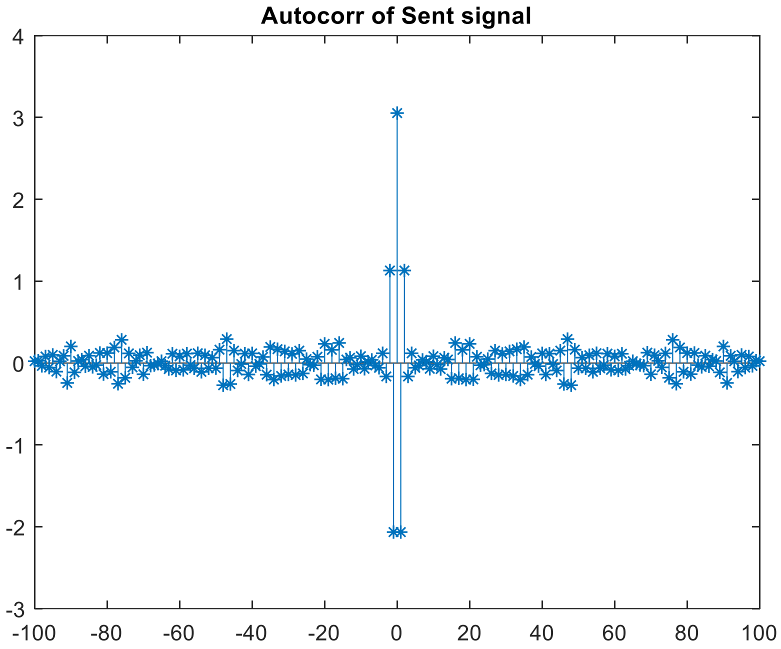
Task 3:

i. Plot the transmitted signal and the received signal on a single figure using *subplot*. Can you estimate the delay D by a visual inspection of the received signal?

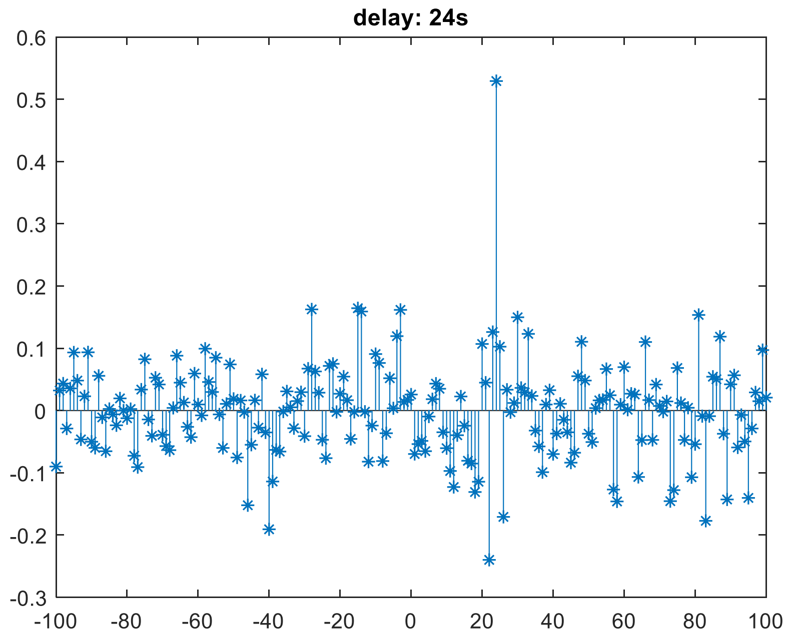


The received signal is too much distorted and transformed by noise to be comparable with the original sent signal. Hence I cannot predict by mere visual inspection.

ii. Plot the sample autocorrelation of the transmitted signal, $r'_{xx}(m)$ for $-100 \leq m \leq 100$



iii. Plot the sample cross-correlation of the transmitted signal and the received signal $c'_{XY}(m)$ for $-100 \le m \le 100$



iv. Determine the delay D from the sample correlation. How did you determine this? Since the maximum auto correlation of the sent signal is at m=0. Hence we just search for the maximum value of cross correlation between the sent and received signal. This max is at m=24, hence the time it took for the signal to return was 24 sec. Assuming total traversing distance=d, the speed of EM wave = c:

From Newton's law:

$$d = c * 24$$

$$d = 48 * 3 * 10^8\ m$$

(assuming time is in seconds)

References

[1] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, *3rd ed.*, McGraw-Hill, New York, 1991.

[2] C. Bouman, Lab Assignment for the Purdue University Course: "Digital Signal Processing with Applications"

[3] D. Berstekas, J, Tsitsiklis, *Introduction to Probability*, *2nd ed.*, Athena Scientific, 2008.