L.EEC025 - FUNDAMENTALS OF SIGNAL PROCESSING

Academic year 2023-2024, week 2
P2P exercises

Illustrative exercises related to "Peer-to-peer learning/assessment" (P2P L/A)

NOTE: these illustrative P2P exercises will be explained during the lectures.

P2P Exercise 1

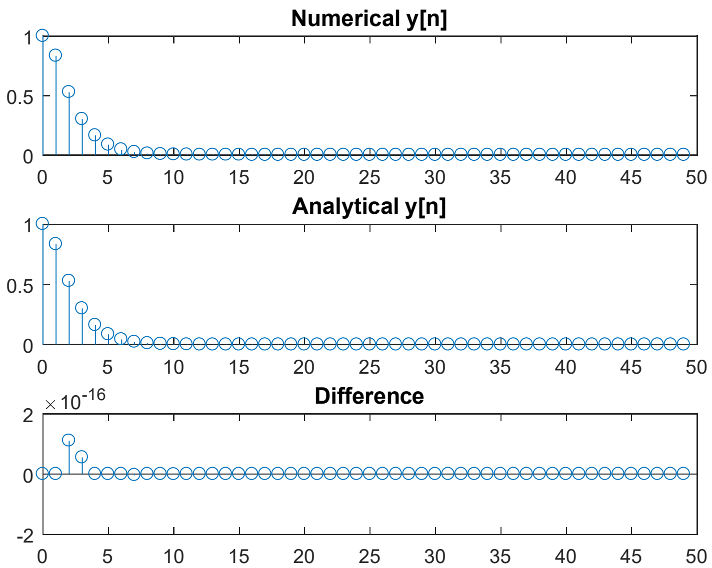
A discrete-time system has impulse response $h[n] = b^n u[n]$ and is excited with the input sequence $x[n] = a^n u[n]$. Using the convolution sum (and not the Fourier Transform), show that the system output is given by $y[n] = \frac{a^{n+1} - b^{n+1}}{a-b} u[n]$.

P2P assessment: **3pt /5** if demonstration is complete and without errors

Validate this result in Matlab using $a = 1/2$, $b = 1/3$, and using $N = 50$ samples. Show that your analytical solution is consistent with the Matlab-based numerical solution by completing the following set of Matlab commands.

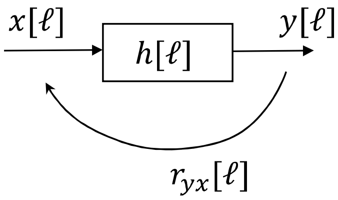
```
a=1/2; b=1/3;
N=50; n=[0:N-1];
% input sequence
x=a.^n;
% impulse response
h=b.^n;
% To be completed...
```

P2P assessment: **2pt /5** if code is completed in such a way as to generate a figure similar to:



P2P Exercise 2

Consider a linear and time-invariant (LTI) system whose impulse response is given by $h[\ell]$, and whose input and output sequences are given, respectively, by $x[\ell]$ and $y[\ell]$. As discussed already in a recent lecture, the cross-correlation between output and input is given by $r_{yx}[\ell] = y[\ell] * x^*[-\ell]$, as the following block diagram illustrates.



The usual definition of impulse response is the output sequence of a system when it is excited with a special (and simple) deterministic signal: $\delta[\ell]$. In other words, $h[\ell] = y[\ell]|_{x[\ell]=\delta[\ell]}$.

This exercise shows that the impulse response of an LTI system may also be found using a random sequence as excitation.

Consider that, according to the definition, the cross-correlation and the discrete-time convolution are given, respectively, by Equations (1) and (2).

$$r_{yx}[\ell] = y[\ell] * x^*[-\ell] = \sum_{k=-\infty}^{+\infty} y[k]\, x^*[k-\ell] \tag{1}$$

$$y[k] = h[k] * x[k] = \sum_{m=-\infty}^{+\infty} h[m]\, x[k-m] \tag{2}$$

Show that when the right-hand side of Eq. (2) replaces $y[k]$ in the right-hand side of Eq. (1), and admitting that ℓ is finite, then Eq. (3) is obtained.

$$r_{yx}[\ell] = h[\ell] * r_x[\ell] \tag{3}$$

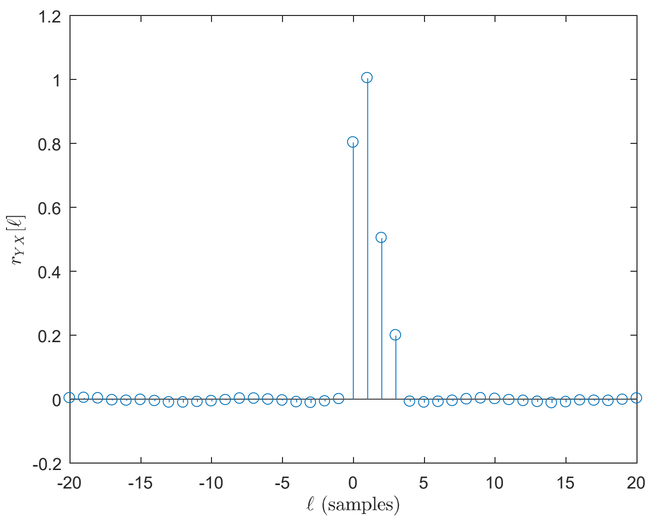
P2P assessment: **4pt /5** if demonstration is complete and without errors

In particular, if $r_x[\ell] = \delta[\ell]$, then $r_{yx}[\ell] = h[\ell]$, which means that the impulse response of the system may be found by exciting the discrete-time system with random noise having an auto-correlation that corresponds to the unit impulse.

Now, illustrate this result in Matlab. Let us assume that the LTI system is a black box. We generate a (zero-mean) random vector having a Gaussian PDF and 1E5 samples. For test purposes, we take the impulse response of the black box system as $h[\ell] = 0.8\delta[\ell] + \delta[\ell-1] + 0.5\delta[\ell-2] + 0.2\delta[\ell-3]$. Complete the following Matlab set of commands showing that when the system is excited with the generated random sequence $x[\ell]$, and when the cross-correlation between output and input is found, then, it delivers $h[\ell]$ very approximately.

```
N=1E5;
maxlag=20;
h=[0.8 1 0.5 0.2]; % our system impulse response
x=randn(N,1); sigma=sqrt(x.'*x); x=x/sigma;
% to confirm that rx[ell]=DELTA[ell]
[rx, lag]=xcorr(x, maxlag);
stem(lag, rx)
xlabel('$$\ell$$ (samples)','Interpreter', 'Latex');
ylabel('$$r_{X}[\ell]$$','Interpreter', 'Latex'); pause
% To be completed...
%
% Note that in Matlab Eq. (2) is computed as y=conv(h,x);
% Note that in Matlab Eq. (1) is computed as:
%                               [ryx, lag]=xcorr(y,x, maxlag);
```

P2P assessment: **1pt /5** if code is completed in such a way as to generate a figure similar to:



Extra question just to think about it: if the noise has a uniform PDF (instead of Gaussian), would that modify the result ?

Extra exercises

Topics: Experiments with the PDF and auto-correlation of deterministic and random signals

Exercise 3

This exercise involves Matlab experimentation and analysis/discussion of the results.

a) Let us create in Matlab a sinusoidal signal containing 1E6 samples using the following code:

```
N=1E6; n=[0:N-1].';
omega=0.0123; A=1; x=A*sin(omega*n);
```

Is this signal periodic ?

What do you expect its PDF to be ? Estimate it using the following code:

```
[H X]=hist(x,50); equalize=50/(max(x)-min(x));
bar(X, H/sum(H)*equalize, 0.5);
ylabel('PDF'); xlabel('x[n] amplitude');
```

Discuss whether or not the PDF of this deterministic signal is deterministic.

What do you expect its auto-correlation function to look like ? Represent its auto-correlation using the following code:

```
maxlag=700;
[rd, lag]=xcorr(x, maxlag);
stem(lag, rd);
xlabel('$$\ell$$ (samples)','Interpreter', 'Latex');
ylabel('$$r_X[\ell]$$','Interpreter', 'Latex');
```

Based on this representation, how can you estimate the *approximate* period (in samples) of the sinusoid ?

In the above code, change the value of omega to `omega=2*pi/80;` . Explain the changes that you observe in the results.

b) Let us now create and represent graphically another deterministic signal in Matlab:

```
N=100; n=[0:N-1]; alfa=0.9;
xd=alfa.^n*(1-alfa^2)/alfa; xd(1)=xd(1)-1/alfa;
stem(n,xd); xlabel('n \rightarrow');
ylabel('Amplitude \rightarrow'); pause
```

This is a specially designed signal (you will be able to design it later on during the semester) whose auto-correlation is peculiar as the following Matlab code allows to observe:

```
[rd, lag]=xcorr(xd); stem(lag, rd);
xlabel('$$\ell$$ (samples)','Interpreter', 'Latex');
ylabel('$$r_X[\ell]$$','Interpreter', 'Latex');
```

This result represents a desired/useful feature in many situations. Can you name one ?

(**NOTE**: Random sequences may also exhibit the same feature as we show next)

Argue if it makes sense or not to find the PDF of this deterministic signal.

c) Let us create a random signal in Matlab containing 1E6 samples and having a uniform PDF between -0.5 and +0.5. The following Matlab code helps to estimate its PDF:

```
N=1E6; x=rand(1,N)-0.5;
[H X]=hist(x,50); equalize=50/(max(x)-min(x));
bar(X, H/sum(H)*equalize, 0.5);
ylabel('PDF'); xlabel('x[n] amplitude'); pause
```

Check also its auto-correlation function using the function Matlab code:

```
maxlag=100;
[rn, lag]=xcorr(x, maxlag);
stem(lag, rn);
xlabel('$$\ell$$ (samples)','Interpreter', 'Latex');
ylabel('$$r_X[\ell]$$','Interpreter', 'Latex');
```

d) Repeat the previous two observations in the case of a random signal having a Gaussian PDF. In this case, it suffices that you change `x=rand(1,N)-0.5;` to `x=randn(1,N);`. Are the new observations as expected ?

e) Let us now create two different random sequences both containing 1E6 samples and having a uniform PDF between -0.5 and +0.5. We also combine them additively.

```
N=1E6; x1=rand(1,N)-0.5; x2=rand(1,N)-0.5;
x=x1+x2;
```

Add code to analyse the PDF and the autocorrelation of `x`. Are those functions as expected ? If not, provide a possible explanation.

[**Note**: an extension of this last experiment can be used to show the Central Limit Theorem]

Discuss whether or not the PDF of the sum of these two random sequences is deterministic.

Exercise 4

Replicate the Matlab code in slide 35 of the first FPS lecture slides addressing the generation of a sinusoidal sequence with a prescribed SNR.

a) Running the code several times produces an SNR that is not exactly 30.00 dB, why ?

b) Adapt the code such that the noise has a uniform PDF (instead of Gaussian) and that the obtained SNR result is fairly close to 30 dB.