ΤΥΧΑΙΕΣ ΔΙΑΔΙΚΑΣΙΕΣ ΚΑΙ ΘΟΡΥΒΟΣ

Σκοπός της εργαστηριακής άσκησης είναι

i. Να θυμηθούμε βασικές έννοιες από τη θεωρία πιθανοτήτων και να εξοικειωθούμε με την αναπαράστασή τους στο Matlab
ii. Να μάθουμε να δημιουργούμε τυχαία σήματα
iii. Να μάθουμε να δημιουργούμε θόρυβο.

Ανασκόπηση στη Θεωρία Πιθανοτήτων

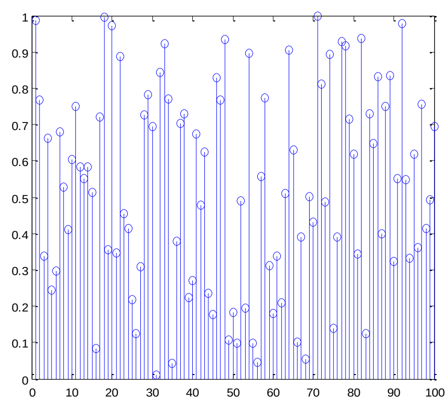
ΤΥΧΑΙΟΙ ΑΡΙΘΜΟΙ

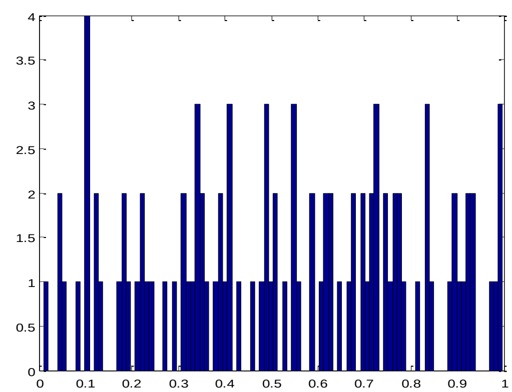
Με την εντολή rand(a,b) παίρνουμε έναν axb πίνακα με «ψευδοτυχαίους» αριθμούς στο διάστημα (0,1)

```
>> x=rand(100,1);

>> stem(x);
```

Μπορούμε να δούμε την κατανομή των τιμών αυτών με την εντολή hist . Η εντολή hist

```
>> hist(x,100)
```

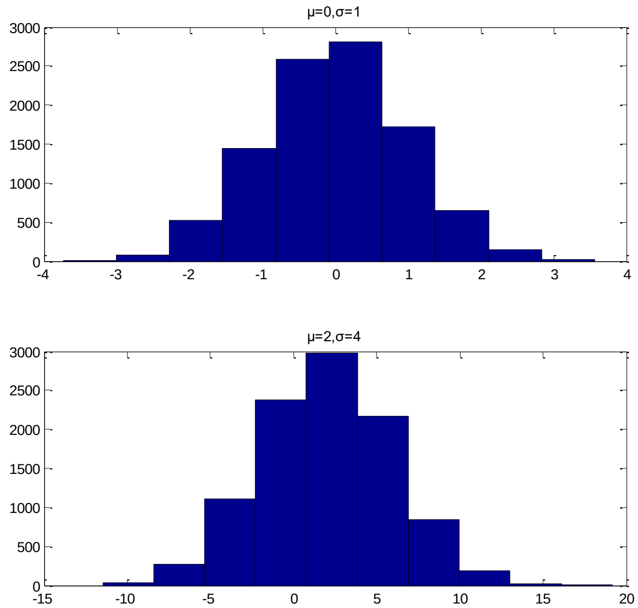




Το ιστόγραμμα υπολογίζει το πλήθος των αριθμών n_A που βρίσκονται σε κάθε διάστημα A.

Τυχαίοι αριθμοί από την κανονική κατανομή, με δύο τρόπους

```
>> x=randn(1,10000);

>> y=normrnd(2,4,1,10000); %R = normrnd(mu,sigma) generates random numbers from the normal distribution with mean parameter mu and standard deviation parameter sigma

>> subplot(211);hist(x);title('μ=0,σ=1');
```



ΜΕΣΗ ΤΙΜΗ, ΔΙΑΣΠΟΡΑ, ΤΥΠΙΚΗ ΑΠΟΚΛΙΣΗ.

Το Matlab μας παρέχει ένα σύνολο από εντολές για να υπολογίζουμε βασικά και γνωστά μεγέθη που αφορούν τις τυχαίες μεταβλητές.

Η εντολή `mean(x)` μας δίνει τη μέση τιμή για την τυχαία μεταβλητή x.
Η εντολή `var(x)` μας δίνει τη διασπορά (είναι η μέση τιμή του τετραγώνου της απόκλισης της τυχαίας μεταβλητής *X από τη μέση* της τιμής)

Τη διασπορά ως γνωστό θα μπορούσαμε να το υπολογίσουμε και ως εξής

```
mean((x-mean(x).*(x-mean(x)))
```

Η εντολή `std(x)` μας δίνει την τυπική απόκλιση

Στο παράδειγμα μας …

```
>> y=normrnd(2,4,1,10000);

>> mean(y)         % ans =   1.9967

>> std(y)   %ans =    3.9433

>> var(y)   %ans =   15.5494
```

ΣΥΜΜΕΤΑΒΟΛΗ (COVARIANCE)

Covariance των X,Y δίνεται από τη σχέση

$$C_{X,Y} = E[(X - m_x)(Y - m_y)]$$

Στο Matlab μπορούμε να γράψουμε μία δική μας συνάρτηση που την υπολογίζει

```
function C = covariance_xy(x,y)
  mx = mean(x);
  my = mean(y);
  C = mean((x-mx).*(y-my));
```

Ή να χρησιμοποιήσουμε την εντολή `cov(x,y)`

ΣΥΝΤΕΛΕΣΤΗΣ ΣΥΣΧΕΤΙΣΗΣ (CORRELATION COEFFICIENT)

Εκφράζεται από το πηλίκο

$$\rho_{X,Y} = \frac{C_{XY}}{\sigma_X \sigma_Y}$$

Άρα μπορούμε να χρησιμοποιήσουμε την συνάρτηση που γράψαμε πριν για το *covariance* και να την αλλάξουμε λίγο ώστε τώρα να μας δίνει *correlation coefficients*

```
function R = corr_xy(x,y)
  mx = mean(x);
  my = mean(y);
  C = mean((x-mx).*(y-my));
   R = C/(std(x)*std(y));
```

Ή να χρησιμοποιήσουμε την εντολή `R = corrcoef(x,y)`

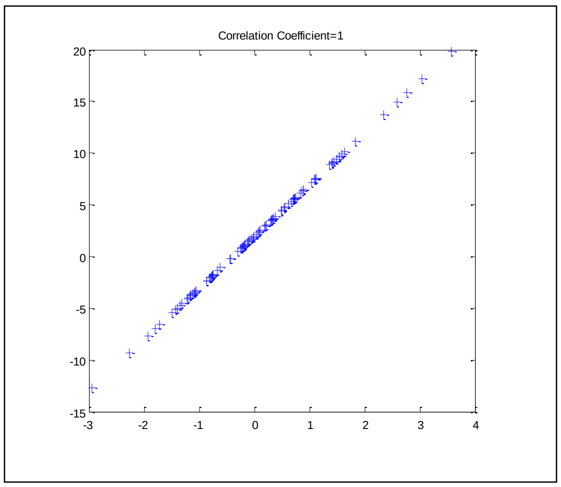
Η συσχέτιση μετρά το βαθμό συνάφειας ανάμεσα σε δύο ή περισσότερες μεταβλητές -1 <=ρ<=1

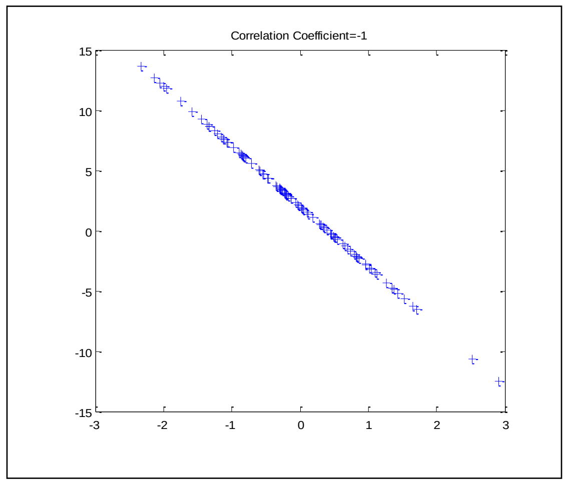
- Όταν παίρνει την τιμή -1, σημαίνει ότι υπάρχει τέλεια συσχέτιση (οι τιμές της μιας μεταβλητής αυξάνουν, ενώ οι τιμές της άλλης μειώνονται)
- Ομοίως η τιμή +1 σημαίνει τέλεια συσχέτιση (οι τιμές και των δύο αυξάνουν ή μειώνονται)
- Και στις δύο παραπάνω περιπτώσεις ισχύει η σχέση Y=αX+β μεταξύ των δύο μεταβλητών, όπου ρ=-1 όταν α>0 και ρ=1 όταν α<0
- Αν r=0 τότε οι μεταβλητές X και Y λέγονται ασυσχέτιστες

Παράδειγμα 1 (μ=0, σ=1)

```
%ρ=1
>> x=normrnd(0,1,1,100);
>> y=5*x+2;
>> r=corrcoef(x,y);
>> plot(x,y,'+')
```

```
%ρ=-1
>> x=normrnd(0,1,1,100);
>> y=-5*x+2;
>> r=corrcoef(x,y);
>> plot(x,y,'+')
```

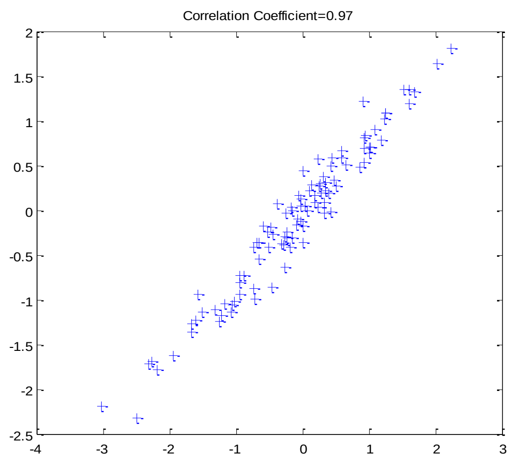


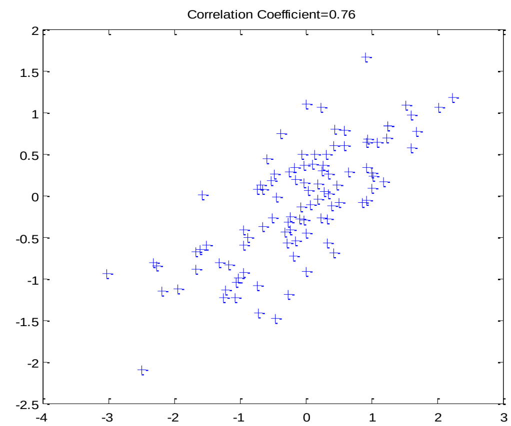


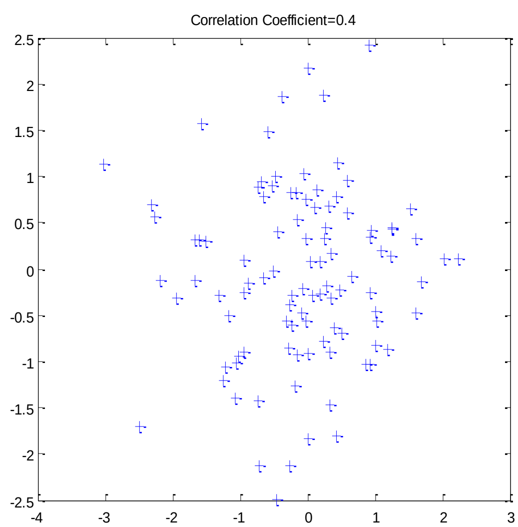
Παράδειγμα 2 με μ=0,σ=1

Δύο τυχαίες μεταβλητές, x,y και άλλη μία z , με z=(x+y)/2 και z=(4x+y)/5.

```
>> x=normrnd(0,1,1,100);
>> y=normrnd(0,1,1,100);
>> z=y;
>> r=corrcoef(x,z);
>> plot(x,z,'+')
>> z=(x+y)/2;
>> r=corrcoef(x,z);
>> figure;plot(x,z,'+')
>> z=(4*x+y)/5;
>> r=corrcoef(x,z);
>> figure;plot(x,z,'+')
```







Άλλες συναρτήσεις για τυχαίους αριθμούς

- use Matlab function unifrnd(a,b,M,N)
- use Matlab function exprnd(lambda,M,N)
- use Matlab function normrnd(mu,sigma,M,N)
- use Matlab function binornd(n,p,M,N)
- use Matlab function poissrnd(lambda,M,N)

Τυχαία σήματα

Τα τυχαία σήματα (Τ.Σ.) μπορούν να θεωρηθούν ως γενίκευση των τυχαίων διανυσμάτων, με τη διαφορά ότι αντί για ένα πεπερασμένο σύνολο τυχαίων μεταβλητών, εξετάσουμε ένα άπειρο σύνολο (πιθανά αριθμήσιμο) από Τ.Μ. Στις τηλεπικοινωνίες, δύο βασικές κατηγορίες τυχαίων σημάτων είναι:

- Το σήμα πληροφορίας που αποστέλλει ο πομπός
- Ο θόρυβος που εισάγει το τηλεπικοινωνιακό κανάλι

Πιο απλά ένα τυχαίο σήμα μεταβάλλεται χρονικά με ένα τρόπο που δεν είναι γνωστός με ακρίβεια , επομένως μπορούμε να κάνουμε πιθανο-θεωρητικούς υπολογισμούς για την μεταβολή του.

Ένα τυχαίο σήμα προκύπτει ως το αποτέλεσμα ενός τυχαίου πειράματος και καθορίζεται ως μια συλλογή (ensemble) **από διαφορετικές** εμφανίσεις (realizations) **κυματομορφών που αντιστοιχούν στα πιθανά αποτελέσματα του πειράματος.**

Στη συνέχεια θα δούμε πως μπορούμε στο Matlab να γράψουμε τέτοια τυχαία σήματα.

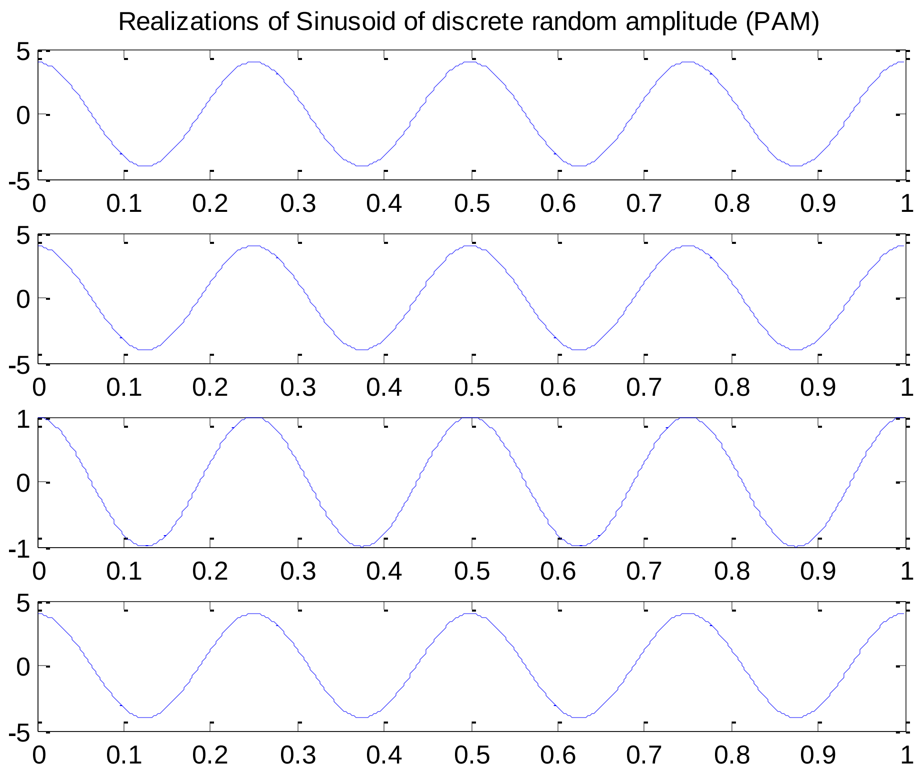
Ημιτονοειδής κυματομορφή με τυχαίο πλάτος

Έστω η παρακάτω ημ. κυματομορφή.

$$X(t) = A\cos(2p\,4t) \quad A \text{ uniform } [1,4]$$

Τι σημαίνει τυχαίο πλάτος; σημαίνει ότι η μεταβλητή Α είναι μία μεταβλητή που δε γνωρίζουμε αλλά ξέρουμε ότι παίρνει τιμές από μία κανονική κατανομή στο διάστημα [1,4] . Το σήμα αυτό επομένως είναι ένα τυχαίο σήμα. Θα χρησιμοποιήσουμε τη συνάρτηση `random` του Matlab, η οποία παράγει πίνακες με αριθμούς από δοσμένη κατανομή και αναλύουμε το τυχαίο σήμα στο χρόνο. Κάθε φορά καθορίζουμε την τιμή του πλάτους και λέμε ότι έχουμε μια «πραγματοποίηση» του τυχαίου σήματος. Η οικογένεια όλων αυτών των τυχαίων σημάτων ονομάζεται τυχαία διαδικασία.

```
>> NUM_REAL=4;
>> SIMULATION_LENGTH=1024;
>> t=0:(1/SIMULATION_LENGTH):(1-1/SIMULATION_LENGTH);
>> realizations=zeros(NUM_REAL,SIMULATION_LENGTH);
>> figure(1);
>> for n=1:NUM_REAL
     realizations(n,:)=random('unid',4,1,1)*cos(2*pi*4*t);
     subplot(NUM_REAL,1,n);
     plot(t,realizations(n,:));
>> end
>> subplot(NUM_REAL,1,1);
>> title('Realizations of Sinusoid of discrete random amplitude
     (PAM)')
```

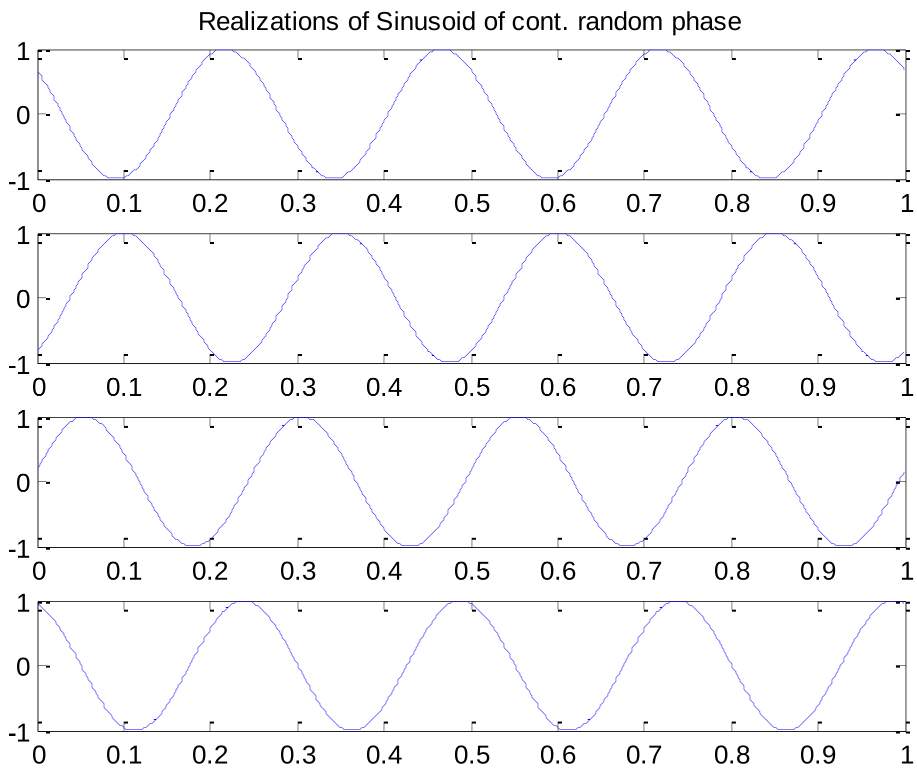


Σήμα με τυχαία φάση

$$X(t) = \cos(2p4t + Q) \quad Q \text{ uniform } [-p, p]$$

Με την ίδια λογική και εδώ, γνωρίζουμε ότι το Q είναι μία τυχαία μεταβλητή που παίρνει τιμές σύμφωνα με την συνάρτηση πυκνότητα πιθανότητας μίας από τις γνωστές κατανομές, και το σήμα εξαρτάται τόσο από το χρόνο όσο και από τη φάση.

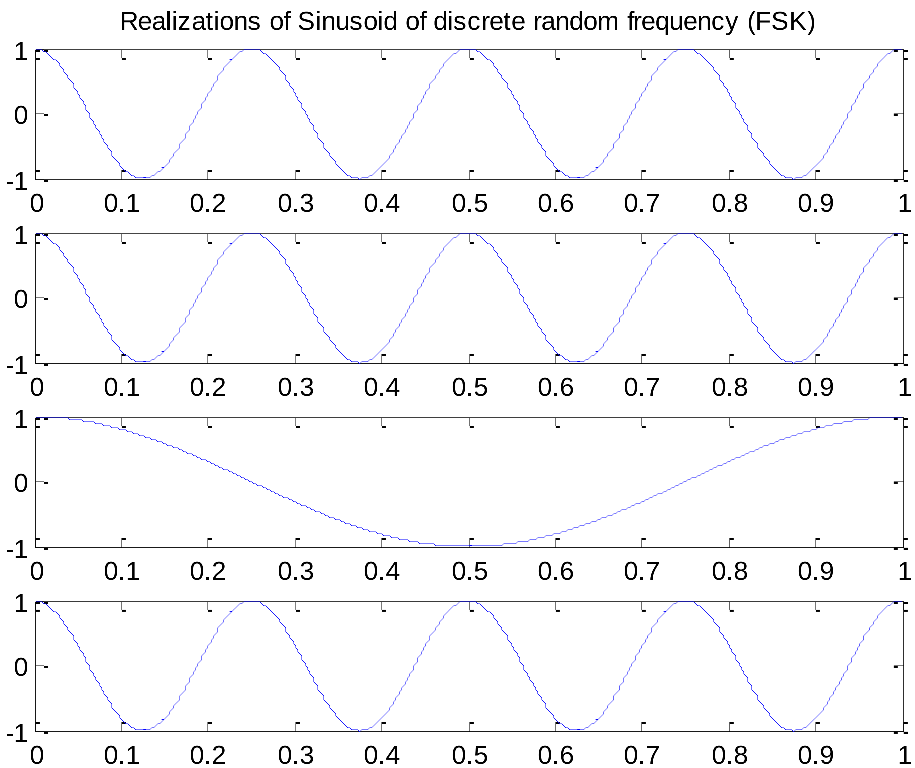
```
>> NUM_REAL=4;
>> SIMULATION_LENGTH=1024;
>> t=0:(1/SIMULATION_LENGTH):(1-1/SIMULATION_LENGTH);
>> realizations=zeros(NUM_REAL,SIMULATION_LENGTH);
>> figure(1);
>> clf;
>> for n=1:NUM_REAL
    realizations(n,:)=cos(2*pi*4*t+random('unif',-pi,pi,1,1));
    subplot(NUM_REAL,1,n);
    plot(t,realizations(n,:));
>> end
>> subplot(NUM_REAL,1,1);
>> title('Realizations of Sinusoid of cont. random phase')
```



Σήμα με τυχαία συχνότητα

$$X(t) = \cos(2\pi f t) \quad f \text{ uniform } [1,4]$$

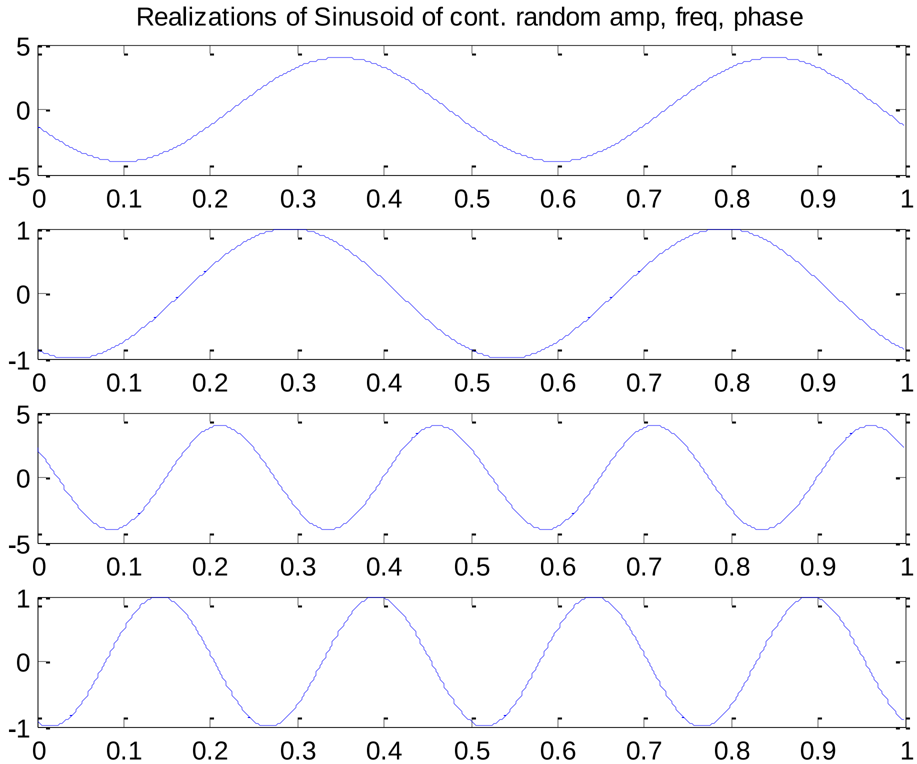
```
>> NUM_REAL=4;
>> SIMULATION_LENGTH=1024;
>> t=0:(1/SIMULATION_LENGTH):(1-1/SIMULATION_LENGTH);
>> realizations=zeros(NUM_REAL,SIMULATION_LENGTH);
>> figure(1);
>> for n=1:NUM_REAL
    realizations(n,:)=cos(2*pi*random('unid',4,1,1)*t);
    subplot(NUM_REAL,1,n);
    plot(t,realizations(n,:));
>> end
>> subplot(NUM_REAL,1,1);
>> title('Realizations of Sinusoid of discrete random frequency
    (FSK)')
```



Τυχαίο σήμα , με ΤΜ πλάτος, φάση, συχνότητα

$$X(t) = A\cos(2\pi ft + \Theta) \quad A \text{ uniform } [1,4] \quad f \text{ uniform } [1,4] \quad \Theta \text{ uniform } [-\pi,\pi]$$

```
>> NUM_REAL=4;
>> SIMULATION_LENGTH=1024;
>> t=0:(1/SIMULATION_LENGTH):(1-1/SIMULATION_LENGTH);
>> realizations=zeros(NUM_REAL,SIMULATION_LENGTH);
>> figure(1);
>> for n=1:NUM_REAL
    realizations(n,:)=random('unid',4,1,1)*cos(2*pi*random('unid',4,
    1,1)*t+random('unif',-pi,pi,1,1)); subplot(NUM_REAL,1,n);
    plot(t,realizations(n,:));
>> end
>> subplot(NUM_REAL,1,1);
>> title('Realizations of Sinusoid of cont. random amp, freq,
    phase')
```


Realizations of Sinusoid of cont. random amp, freq, phase
5
0
-5
1
0
-1
0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1

Η πυκνότητα φάσματος ισχύος (Power Spectral Density – PSD)

Μέχρι εδώ εξετάσαμε τυχαία σήματα στο πεδίο του χρόνου. Στη συνέχεια θα μελετήσουμε τα τυχαία σήματα στο πεδίο της συχνότητας. Πολλές από τις έννοιες που θα χρησιμοποιήσουμε σε αυτό το μέρος της εργαστηριακής άσκησης σας είναι γνωστές από τη θεωρία. Η ανάλυση Fourier παίζει σημαντικό ρόλο και στην μελέτη των τυχαίων διαδικασιών. Όπως είδαμε η τυχαία διαδικασία είναι ένα σύνολο σημάτων διακριτού –χρόνου και άρα δεν μπορούμε να υπολογίσουμε τον μετασχηματισμό Fourier της διαδικασίας. Παρόλα αυτά όπως θα δούμε παρακάτω είναι δυνατόν να αναπτύξουμε μια αναπαράσταση της διαδικασίας στο χώρο των συχνοτήτων αν εκφράσουμε το μετασχηματισμό Fourier με όρους ενός μέσου συνόλου (ensemble average).

Η πυκνότητα φάσματος ισχύος (Power Spectral Density – PSD) για ένα σήμα ορίζεται ως ο μετασχηματισμός Fourier της συνάρτησης αυτοσυσχέτισης. Για μια στατική υπό την ευρεία έννοια στοχαστική ανέλιξη διακριτού χρόνου μπορεί να υπολογισθεί από τη σχέση

$$S_X(f) = \lim_{N\to\infty} E\left\{ \frac{1}{2N+1} \left| \sum_{n=-N}^{N} x[n] \exp(-j2\pi n f) \right|^2 \right\} \quad (\text{εξ.}10)$$

που προκύπτει από το θεώρημα Wiener-Khinchin.

Η προηγούμενη σχέση (εξ. 10) υποθέτει ότι λαμβάνουμε την αναμενόμενη τιμή στο χώρο όλων των πιθανών εκδοχών (ensemble average). Στην πράξη όμως διαθέτουμε μόνο ένα δείγμα της ανέλιξης. Επίσης, ο εμπλεκόμενος μετασχηματισμός Fourier έχει άπειρο μήκος, ενώ διαθέτουμε πεπερασμένα το πλήθος δείγματα $x[n]$ της ανέλιξης. Επομένως, για την εκτίμηση της PSD μιας στατικής υπό την ευρεία έννοια στοχαστικής ανέλιξης, μπορούμε είτε να υπολογίσουμε τον DFT του σήματος και μετά να λάβουμε κάποια μορφή μέσης τιμής είτε, εναλλακτικά, να εκτιμήσουμε τη συνάρτηση αυτοσυσχέτισης χρησιμοποιώντας κάποια μορφή μέσης τιμής και μετά να υπολογίσουμε τον DFT. Αμφότερες οι προσεγγίσεις οδηγούν στους κλασικούς «μη παραμετρικούς» αλγόριθμους εκτίμησης της PSD.

Η διαφορετικά, η πυκνότητα φάσματος ισχύος και η συνάρτηση αυτοσυσχέτισης αποτελούν ζευγάρι μετασχηματισμού Fourier.

$$S_x(f) = \int_{-\infty}^{\infty} R_x(\tau) \exp(-j2\pi f\tau)\, d\tau$$

$$R_x(\tau) = \int_{-\infty}^{\infty} S_x(f) \exp(j2\pi f\tau)\, df$$

Οι σχέσεις αυτές δείχνουν ότι εάν είτε η συνάρτηση αυτοσυσχέτισης είτε η πυκνότητα φάσματος ισχύος μιας στοχαστικής ανέλιξης είναι γνωστές, τότε η άλλη μπορεί να βρεθεί ακριβώς. Είναι κοινά αποδεκτό ότι η πυκνότητα φάσματος ισχύος είναι πιο κρίσιμη παράμετρος.
Αυτό το αποτέλεσμα , πέραν της σημασίας του για την ανάλυση τυχαίων σημάτων είναι πολύ σημαντικό για ομαδοποιεί όλα τα σήματα τυχαία και μη, έτσι ώστε η φασματική πυκνότητα ενός οποιουδήποτε σήματος, υπολογίζεται από τον μετασχηματισμό Fourier της αυτοσυσχέτισης του.

Στη συνέχεια θα δούμε το πρόβλημα της εκτίμησης της πυκνότητας φάσματος ισχύος, μίας στάσιμης υπό της ευρεία έννοια τυχαίας διαδικασίας μέσα από μη παραμετρικές μεθόδους, και ο απλούστερος εκτιμητής είναι το περιοδόγραμμα (*periodogram*) που βασίζεται αποκλειστικά στους ορισμούς πυκνότητας φάσματος ισχύος που είδαμε πριν. Στην πράξη η εκτίμηση με το περιοδόγραμμα είναι καλή εφόσον το μέγεθος της τυχαίας διαδικασίας είναι επαρκώς μεγάλο (για να ικανοποιηθεί η συνθήκη της στασιμότητα) , κάτι που δε είναι πάντα εφικτό. Ένα άλλο πρόβλημα είναι ότι συχνά το σήμα είναι παραμορφωμένο από θόρυβο.

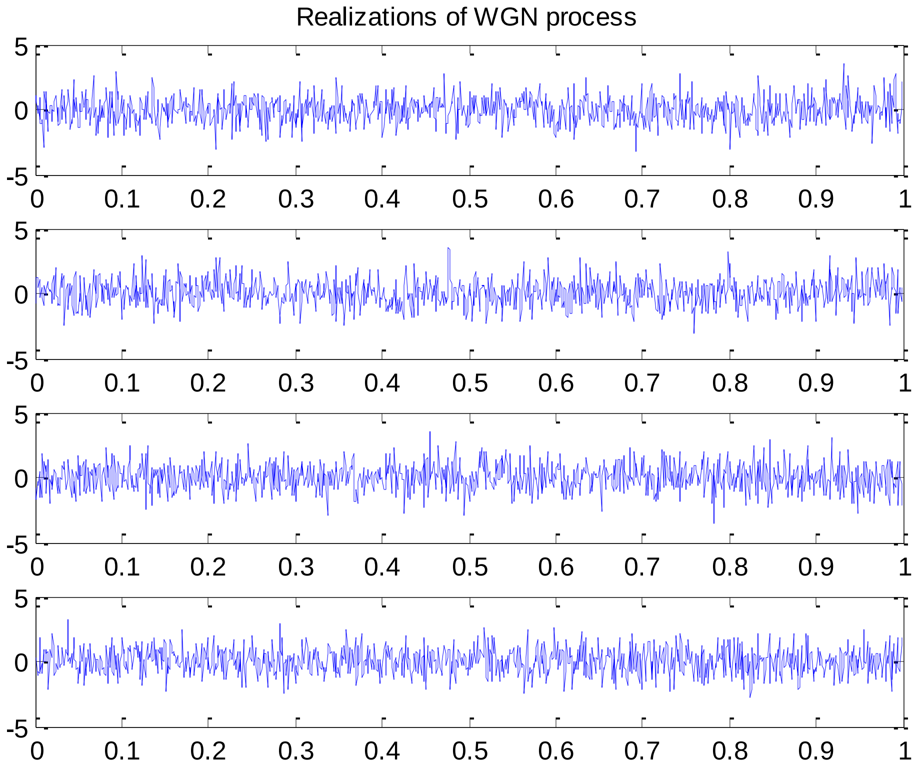
Το περιοδόγραμμα βασίζεται στον ορισμό της εξ.10 , και ευτυχώς για εμάς υπάρχει αντίστοιχη συνάρτηση στο Matlab!

White Gaussian Random Process

Ένα άλλο παράδειγμα τυχαίου σήματος είναι ο λευκός θόρυβος. Λευκός θόρυβος ονομάζεται το σήμα στο οποίο η ισχύς του κατανέμεται ομοιόμορφα σε όλες τις συχνότητες. Δηλαδή είναι ένα σταθερό σήμα στο χώρο της συχνότητας. Ο αντίστροφος μετασχηματισμός Fourier ενός τέτοιου σήματος είναι η συνάρτηση δ ενώ ο αντίστροφος μετασχηματισμός Fourier της PSD είναι όπως είδαμε η συνάρτηση αυτοσυσχέτισης. Επομένως η συνάρτηση αυτοσυσχέτισης ενός τέτοιου σήματος είναι μία συνάρτηση δ. Αυτό πρακτικά σημαίνει ότι δείγματα από ένα τέτοιο σήμα είναι ασυσχέτιστα μεταξύ τους.

Στη συνέχεια βλέπουμε ένα παράδειγμα από λευκό θόρυβο που απαρτίζεται από Gaussian TM.

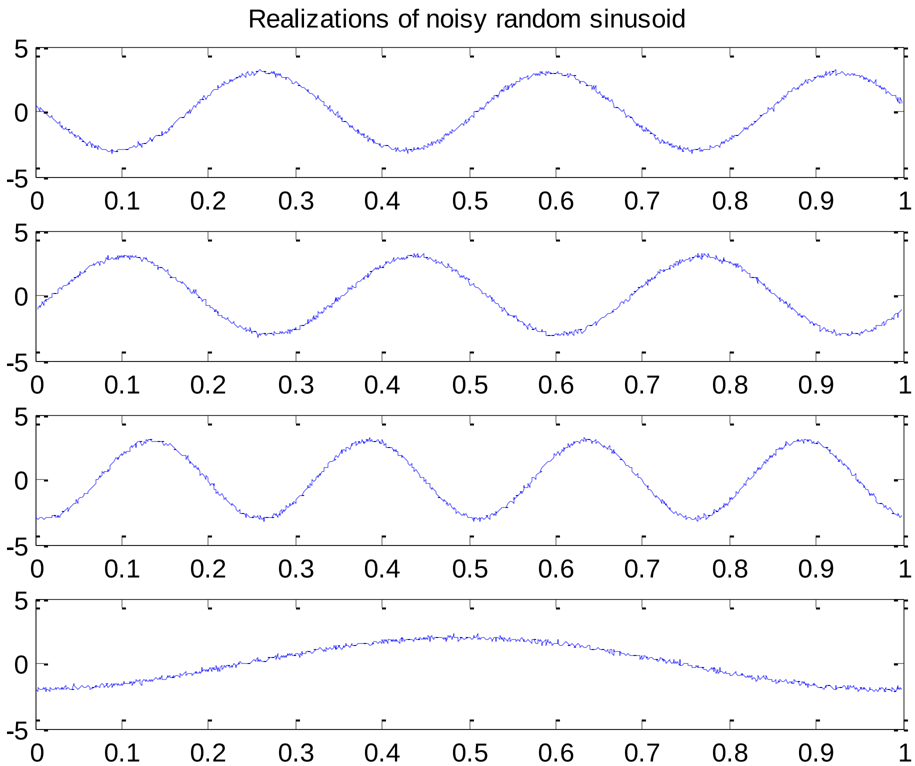
```
>> NUM_REAL=4;
>> SIMULATION_LENGTH=1024;
>> t=0:(1/SIMULATION_LENGTH):(1-1/SIMULATION_LENGTH);
>> realizations=zeros(NUM_REAL,SIMULATION_LENGTH);
>> figure(1);
>> for n=1:NUM_REAL
     realizations(n,:)=randn(1,SIMULATION_LENGTH);
     subplot(NUM_REAL,1,n);
     plot(t,realizations(n,:));
>> end
>> subplot(NUM_REAL,1,1);
>> title('Realizations of WGN process')
```



Noisy Random Sinusoid

$$X(t) = A\cos(2\pi ft + \Theta) + N$$

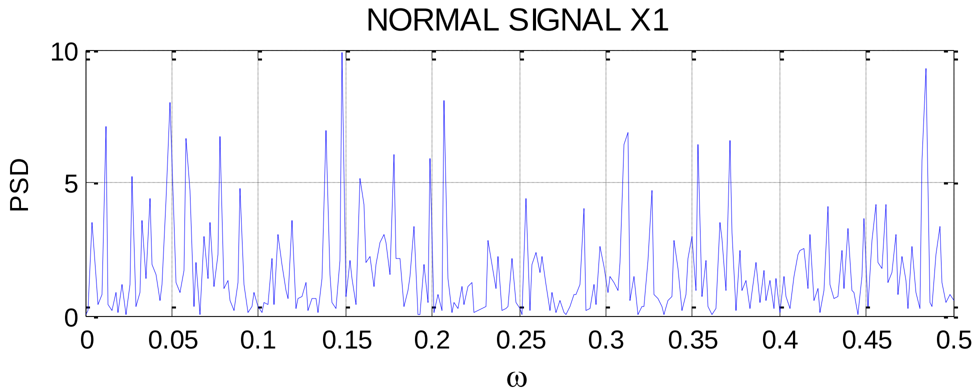
```
>> NUM_REAL=4;
>> SIMULATION_LENGTH=1024;
>> t=0:(1/SIMULATION_LENGTH):(1-1/SIMULATION_LENGTH);
>> realizations=zeros(NUM_REAL,SIMULATION_LENGTH);
>> figure(1);
>> for n=1:NUM_REAL
     realizations(n,:)=random('unid',4,1,1)*cos(2*pi*random('unid',4,1,1)*t+
     random('unif',-pi,pi,1,1))+0.1*randn(1,SIMULATION_LENGTH);
>> subplot(NUM_REAL,1,n);
>> plot(t,realizations(n,:));
>> end
>> subplot(NUM_REAL,1,1);
>> title('Realizations of noisy random sinusoid')
```

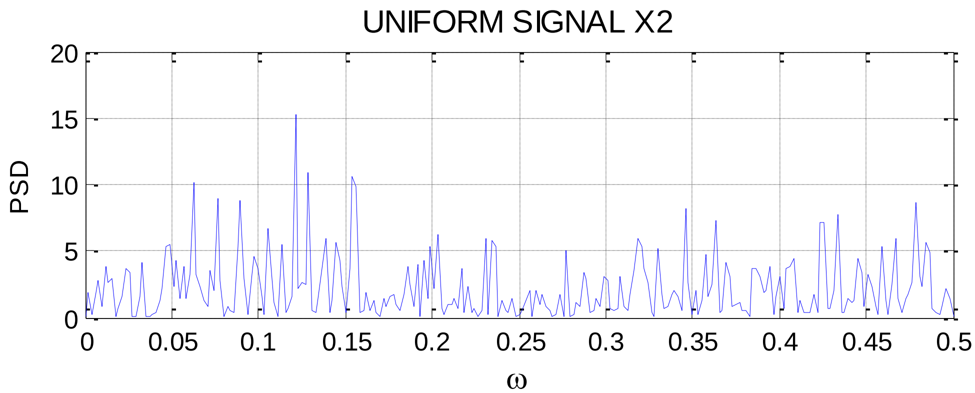

Realizations of noisy random sinusoid
5
0
-5
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1

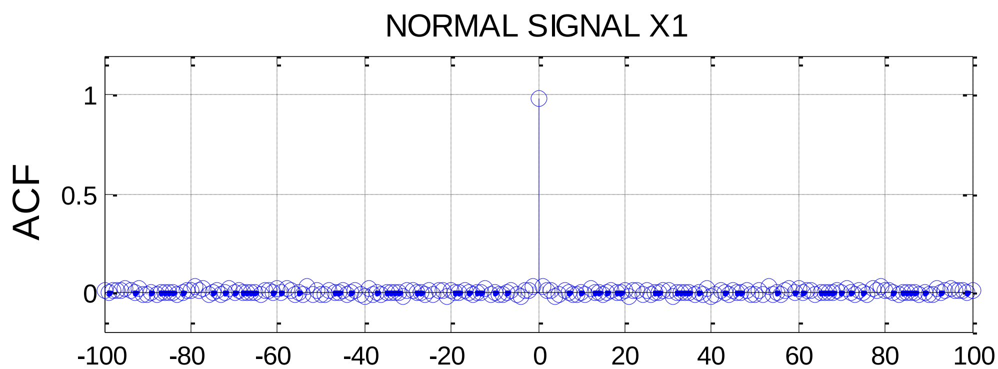
Παράδειγμα

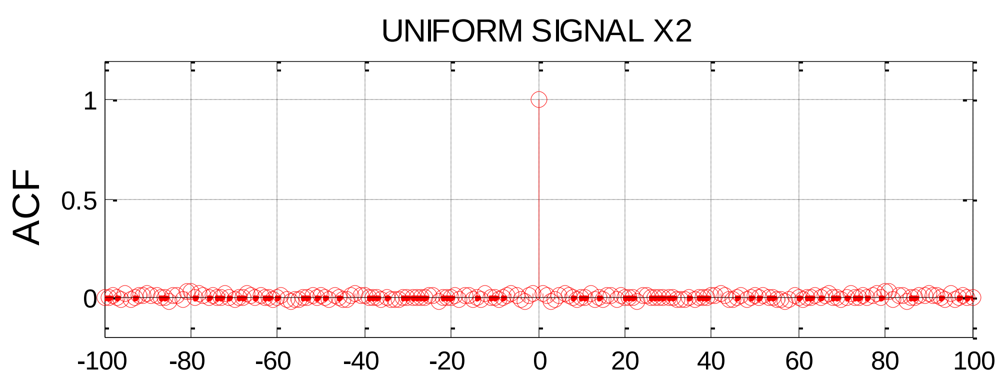
Χρησιμοποιώντας την συνάρτηση periodogram υπολογίστε και σχεδιάστε την πυκνότητα φάσματος ισχύος για δύο σήματα λευκού θορύβου , X1(normal) και X2 (uniform) για τα οποία μ=0 και σ=1.

```
>> %   GENERATION OF SIGNALS:
>> N1=10000;
>> X1=randn(1,N1);
>> X2=3.4641*rand(1,N1)-1.7321;
%%   PSD estimation

>> [S1,w]=periodogram(X1,[],512,1);
>> [S2,w]=periodogram(X2,[],512,1);
>> figure

>> subplot(2,1,1),plot(w,S1),grid, ylabel('PSD'),title('NORMAL SIGNAL
X1','FontSize',[12]),xlabel('\omega','FontSize',[12])
>> subplot(2,1,2),plot(w,S2),grid, ylabel('PSD'), title ('UNIFORM SIGNAL
X2','FontSize',[12]),xlabel('\omega','FontSize',[12])

%Autocorrelation functions:
>> r1=xcorr(X1,'biased');
>> r1=r1';
>> r2=xcorr(X2,'biased');
>> r2=r2';
%  ZOOM
>> N=100;
>> figure
>> subplot(2,1,1),stem(-N:N,r1((N1-N):N1+N),'b'),ylabel('ACF
','FontSize',[14]),title('NORMAL SIGNAL X1','FontSize',[12]),grid, axis([-
N,N,-0.2,1.2]);

>> subplot(2,1,2),stem(-N:N,r2((N1-N):N1+N),'r'),ylabel('ACF
','FontSize',[14]),title('UNIFORM SIGNAL X2','FontSize',[12]),grid,axis([-
N,N,-0.2,1.2]);
```


NORMAL SIGNAL X1
PSD
10
5
0
0
0.05
0.1
0.15
0.2
0.25
0.3
0.35
0.4
0.45
0.5
ω


UNIFORM SIGNAL X2
PSD
20
15
10
5
0
0
0.05
0.1
0.15
0.2
0.25
0.3
0.35
0.4
0.45
0.5
ω

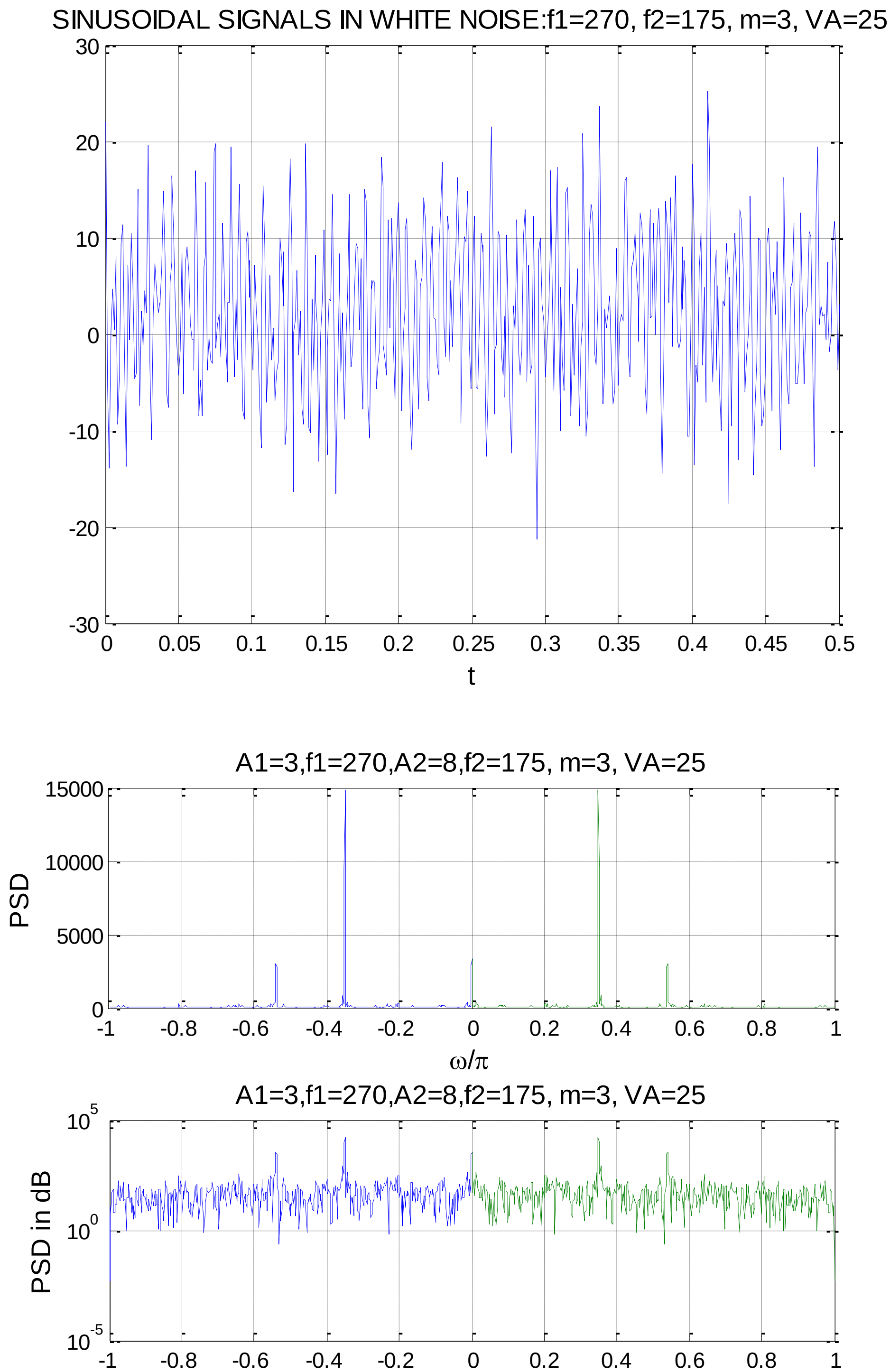

NORMAL SIGNAL X1
ACF
1
0.5
0
-100
-80
-60
-40
-20
0
20
40
60
80
100


UNIFORM SIGNAL X2
ACF
1
0.5
0
-100
-80
-60
-40
-20
0
20
40
60
80
100

Παράδειγμα

Χρησιμοποιώντας την συνάρτηση periodogram υπολογίστε και σχεδιάστε την πυκνότητα φάσματος ισχύος για ένα σήμα το οποίο αποτελείται από το άθροισμα δύο ημιτονοειδών σημάτων με πλάτος A1=3 και A2=8 αντίστοιχα και με f1=270Hz, f2=175Hz στα οποία έχει επιπλέον προστεθεί λευκός θόρυβος με μέση τιμή μ=3 και διασπορά σ^2 =25.

```
>> %   GENERATION OF SIGNALS:
>> Fs=1000;
>> t=0:1/Fs:.5;
>> A1=3;
>> A2=8;
>> f1=270;
>> f2=175;
>> m=3;
>> VA=25;
>> x1=A1*cos(2*pi*t*f1);
>> x2=A2*cos(2*pi*t*f2);
>> X=5*randn(size(t))+3+x1+x2;
%Plot
>> figure

>> plot(t,X), xlabel('t','FontSize',[12]),title('SINUSOIDAL SIGNALS IN
WHITE NOISE:f1=270, f2=175, m=3, VA=25','FontSize',[12]),grid

%   PSD estimation
>> [S,w]=periodogram(X,[],1024,1);

>> figure

>> subplot(2,1,1),plot(-2*w,S,2*w,S),grid,
ylabel('PSD','FontSize',[12]),title('A1=3,f1=270,A2=8,f2=175, m=3,
VA=25','FontSize',[12]),xlabel('\omega/\pi','FontSize',[12]),
>>subplot(2,1,2),semilogy(-2*w,S,2*w,S),grid, ylabel('PSD in
dB','FontSize',[12]),title('A1=3,f1=270,A2=8,f2=175, m=3,
VA=25','FontSize',[12])
```


SINUSOIDAL SIGNALS IN WHITE NOISE:f1=270, f2=175, m=3, VA=25
30
20
10
0
-10
-20
-30
0
0.05
0.1
0.15
0.2
0.25
0.3
0.35
0.4
0.45
0.5
t
A1=3,f1=270,A2=8,f2=175, m=3, VA=25
PSD
15000
10000
5000
0
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8
1
ω/π
A1=3,f1=270,A2=8,f2=175, m=3, VA=25
PSD in dB
10^5
10^0
10^{-5}
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8
1

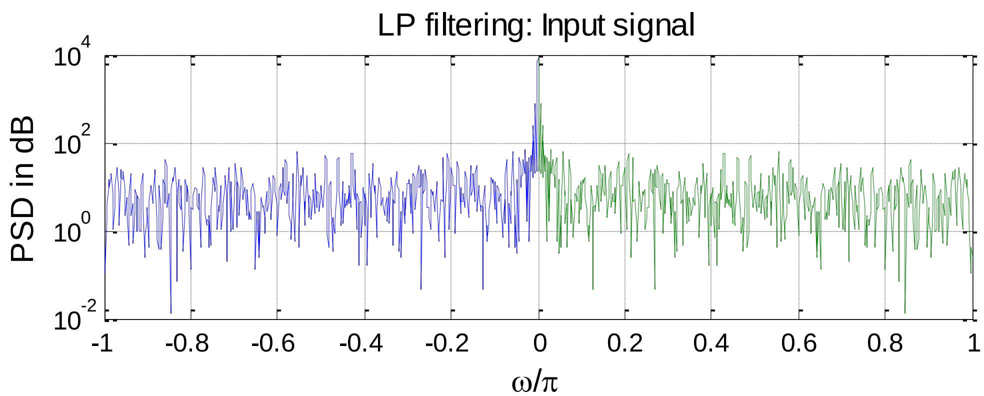
Παράδειγμα

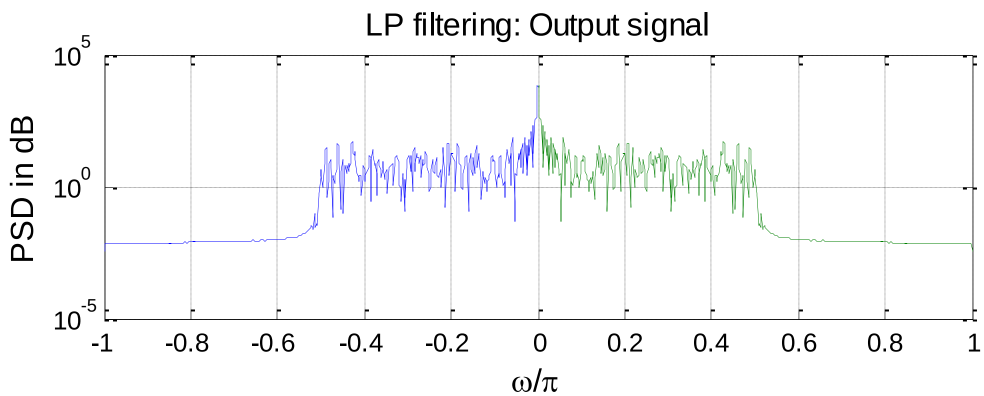
Δίνεται ένα σήμα θορύβου με μέση τιμή μ=4 και διασπορά $\sigma^2 = 4$.

Α. Φιλτράρουμε αρχικά το σήμα με ένα χαμηλοπερατό φίλτρο τάξης N=150 και συχνότητας αποκοπής Wn=0.5, (χρησιμοποιώντας τη συνάρτηση fir1) και στη συνέχεια χρησιμοποιώντας το περιοδόγραμμα υπολογίζουμε την πυκνότητα φάσματος ισχύος στην έξοδο του φίλτρου.

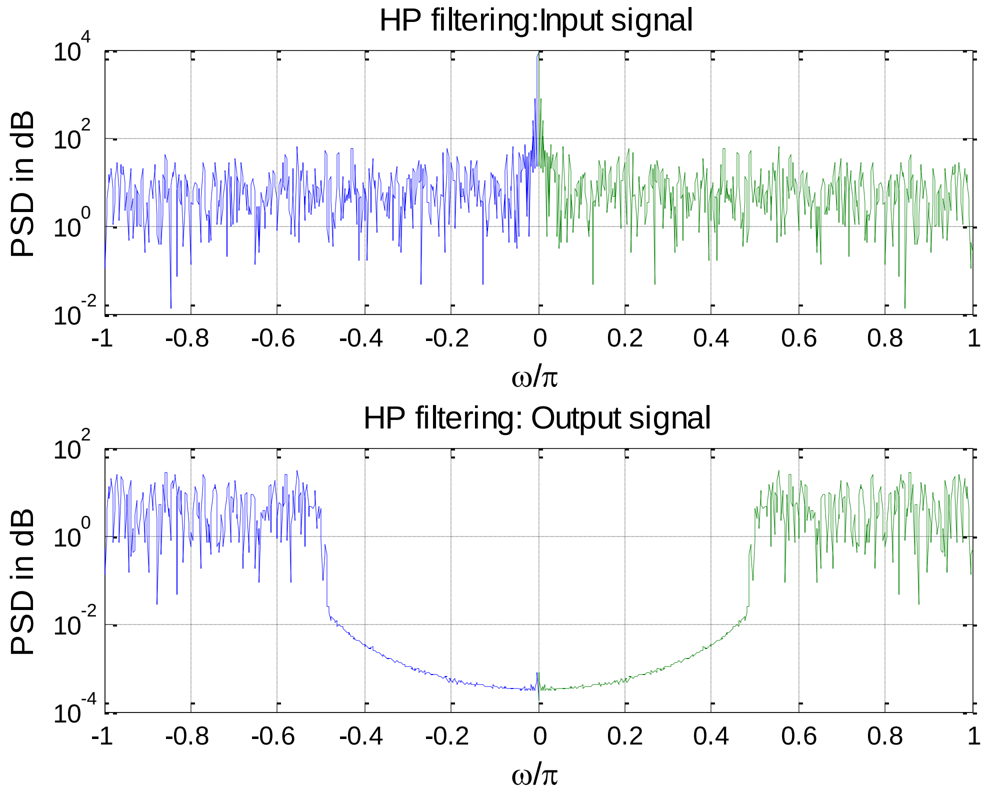
Β. Φιλτράρουμε τώρα το σήμα με υψηπερατό φίλτρο, με τα ίδια χαρακτηριστικά με το Α, και σχεδιάζουμε την πυκνότητα φάσματος ισχύος στην έξοδο του φίλτρου.

```
>> %   GENERATION OF SIGNALS:
>> Fs=1000;
>> t=0:1/Fs:.5;
>> m=4;
>> VA=4;
>> X=2*randn(size(t))+4;

>> %     1.  LP FILTERING

>> h=fir1(150,.5);
>> Y=filter(h,1,X);
>> [SX,w]=periodogram(X,[],1024,1);
>> [SY,w]=periodogram(Y,[],1024,1);
>> figure

>> subplot(2,1,1),plot(-2*w,SX,2*w,SX),
grid,ylabel('PSD','FontSize',[12]),title('LP filtering:Input signal',
'FontSize',[12]),
xlabel('\omega/\pi','FontSize',[12]),
>> subplot(2,1,2),plot(-2*w,SY,2*w,SY), grid, ylabel('PSD','FontSize',[12])
,title('LP filtering:Output signal', 'FontSize' ,[12]),
xlabel('\omega/\pi', 'FontSize',[12]),
>> figure
>> subplot(2,1,1),semilogy(-2*w,SX,2*w,SX),grid, ylabel('PSD in
dB','FontSize',[12]),title('LP filtering: Input signal','FontSize',[12]),
xlabel('\omega/\pi','FontSize',[12]),
>> subplot(2,1,2),semilogy(-2*w,SY,2*w,SY),grid, ylabel('PSD in
dB','FontSize',[12]),title('LP filtering: Output signal','FontSize',[12]),
xlabel('\omega/\pi','FontSize',[12]),
```






```
%2.  HP FILTERING)

>> h=fir1(150,.5,'high');
>> Y=filter(h,1,X);
%   PSD estimation

>> [SX,w]=periodogram(X,[],1024,1);
>> [SY,w]=periodogram(Y,[],1024,1);
>> figure
>> subplot(2,1,1),plot(-2*w,SX,2*w,SX),grid,
ylabel('PSD','FontSize',[12]),title('HP filtering:Input
signal','FontSize',[12]),
xlabel('\omega/\pi','FontSize',[12]),
>> subplot(2,1,2),plot(-2*w,SY,2*w,SY),grid,
ylabel('PSD','FontSize',[12]),title('HP filtering: Output
signal','FontSize',[12])
xlabel('\omega/\pi','FontSize',[12]),
>> figure
>> subplot(2,1,1),semilogy(-2*w,SX,2*w,SX),grid, ylabel('PSD in
dB','FontSize',[12]),title('HP filtering:Input signal','FontSize',[12]),
xlabel('\omega/\pi','FontSize',[12]),
>> subplot(2,1,2),semilogy(-2*w,SY,2*w,SY),grid, ylabel('PSD in
dB','FontSize',[12]),title('HP filtering: Output signal','FontSize',[12])
xlabel('\omega/\pi','FontSize',[12]),
```


HP filtering:Input signal
PSD in dB
ω/π
HP filtering: Output signal
PSD in dB
ω/π