

A large, abstract 3D surface graphic that resembles a mountain or a wave. It has a color gradient from teal on the left to bright yellow and orange on the right, with a dark red peak. The surface is smooth and curved.

# Atelier Communications Numériques

TP2 (05/03/2018)

Oueslati Mohamed melek  
Master SmartCom

## Détails du TP

### TP :

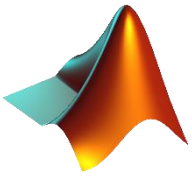
TP numéro 2 de la date: 05-03-2018

### Objectifs du TP :

On va générer un signal  $Y = X$  (un signal aléatoire BPSK constitué de  $n$  éléments binaires) +  $N$  (un signal aléatoire gaussien de moyenne nulle et d'écart-type  $\sigma$  constitué de  $n$  éléments).  
Puis on va essayer de calculer le nombre d'erreurs BER en fonction du rapport signal à bruit SNR.

### Logiciel utilisé :

Matlab version 2014



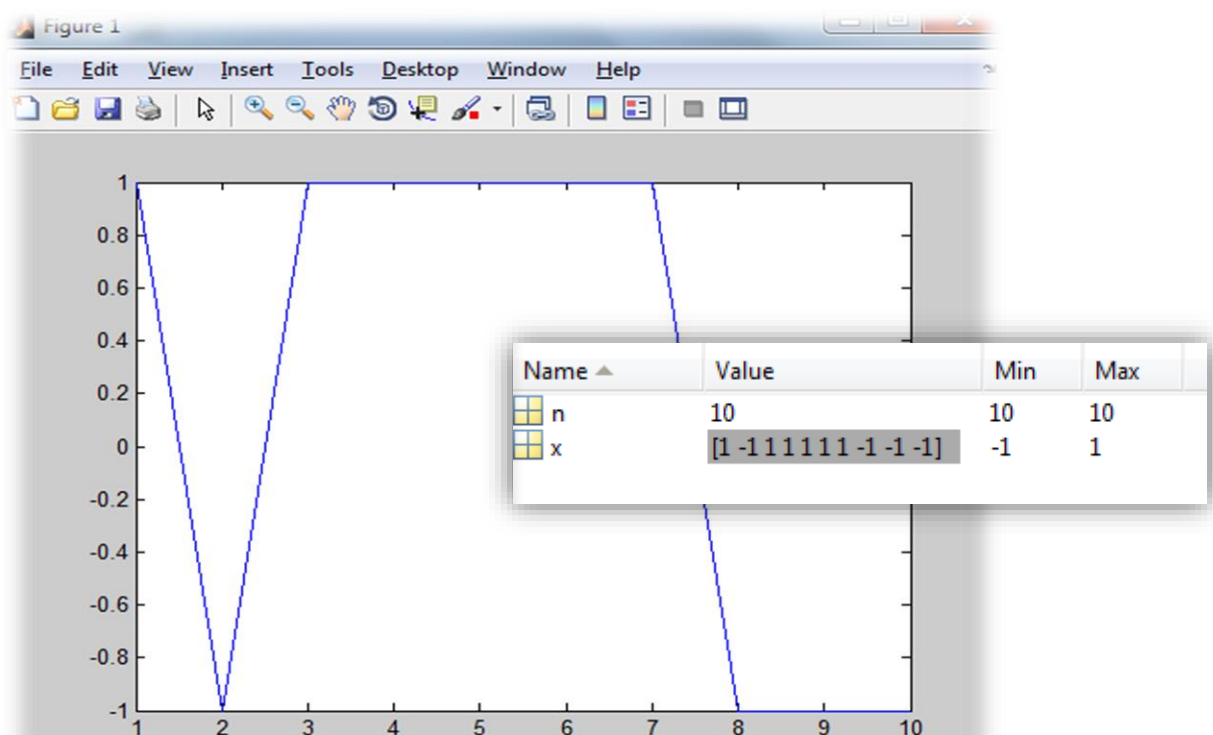
## Questions 1 et 2

```
tp2.m  x  +  
- n=10 ;  
- x=2*round(rand(1,n))-1 ;  
- plot(x)
```

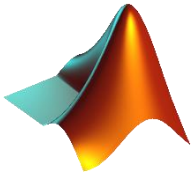
Code qui permet de générer un signal aléatoire BPSK constitué de N=(10) éléments binaires.

### Interprétation:

On choisit d'abord la valeur de  $n=10$ . La commande `rand(1,n)` va ensuite générer des valeurs aléatoires entre  $[0,1]$ , l'arrondissement de ces valeurs se font par la commande « round » qui choisit la valeur entière pour le réel donné. On va ensuite obtenir la courbe ci-dessous :



Représentation graphique du signal  $X f(x)$



### Question 3

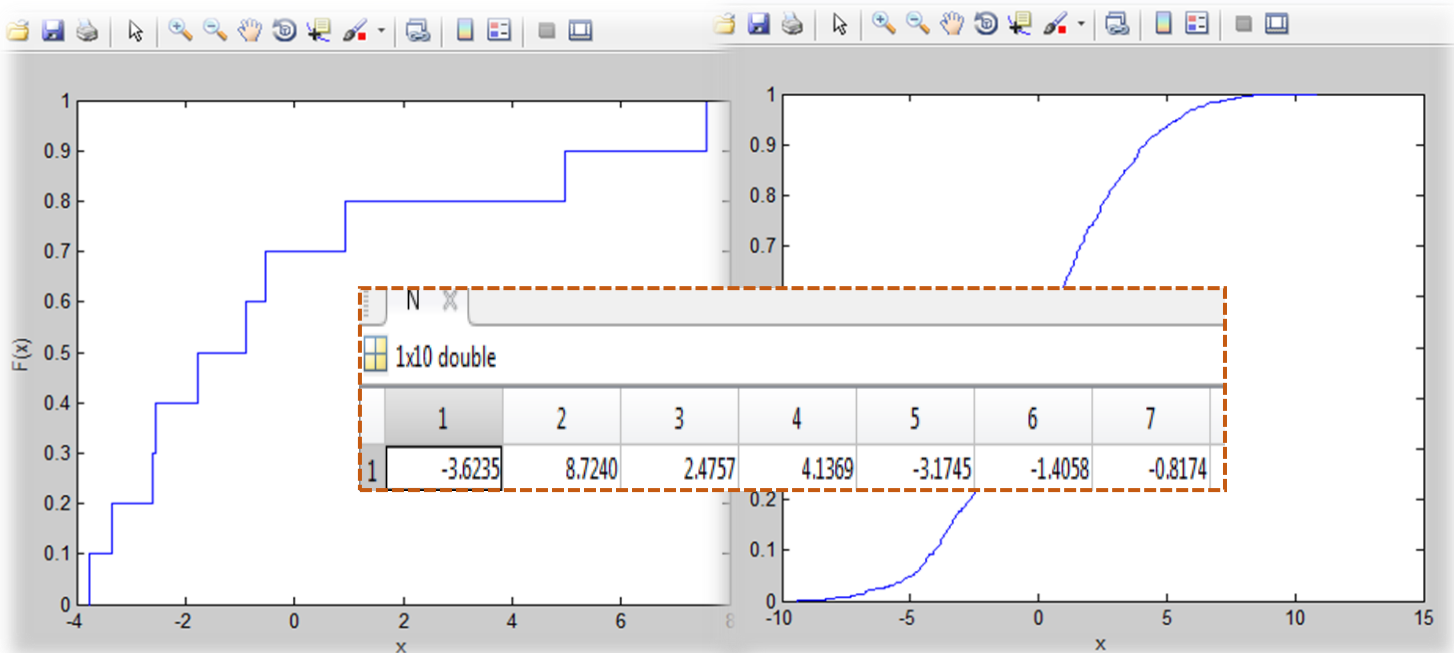
```
Editor - C:\Users\Administrateur\Documents\MATLAB\tp2\tp2.m
tp2.m x +
1 - n=10 ;
2 - x=2*round(rand(1,n))-1 ;
3 - plot(x);
4 - sigma=3 ;
5 - N=sigma*randn(1,n);
6 - plot(N);
7 - ecdf(N)
```

```
sigma=3 ;
N=sigma*randn(1,n);
plot(N);
ecdf(N)
```

Code qui permet de générer un signal nommer **N**  $f(n)$  aléatoire gaussien de moyenne nulle et d'écart-type  $\sigma$  constitué de **n (=10)** éléments.

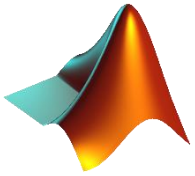
#### Interprétation:

On doit d'abord initialiser le sigma (dans ce cas,  $\sigma=3$ ). On commence ensuite à générer le signal **N** avec `randn`. Le résultat obtenu est représenté dans la figure ci-dessous :



Représentation graphique  $f(N)$  pour  $n=10$

Représentation graphique  $f(N)$  pour  $n=1000$



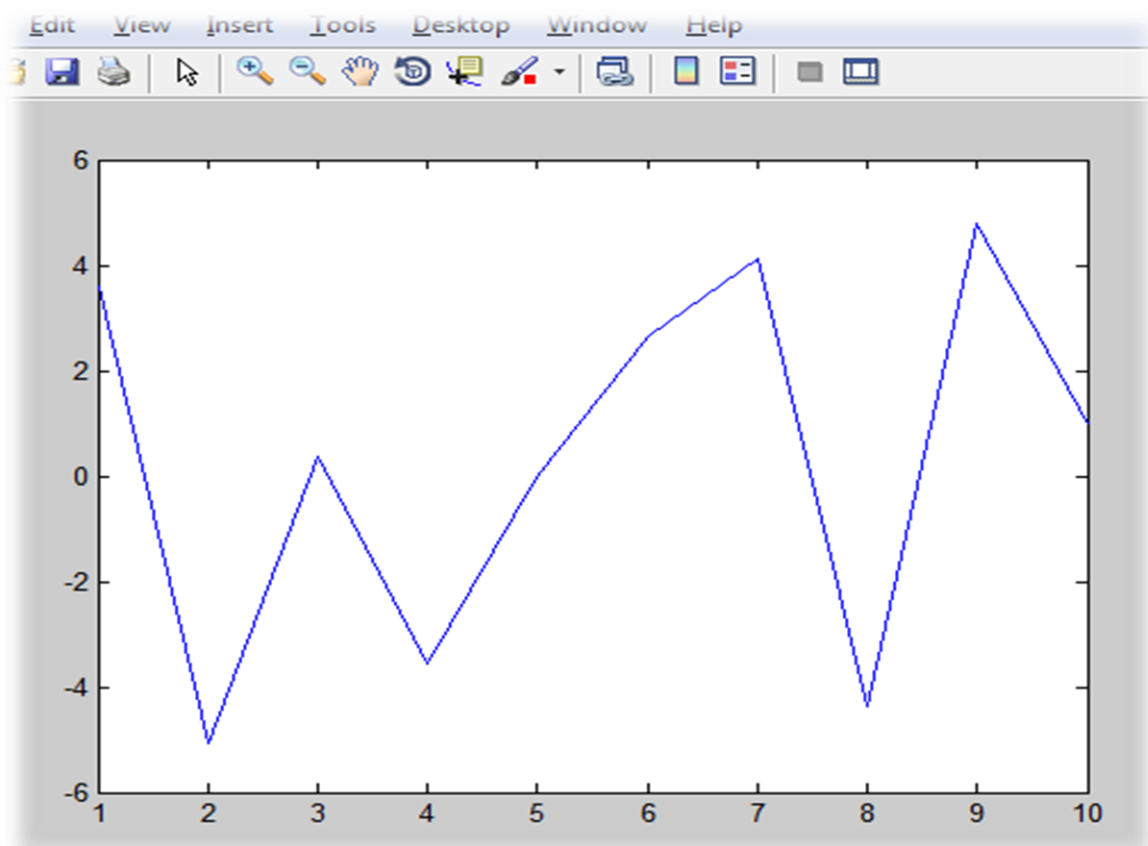
## Question 4

```
tp2.m*  x  +  
9 -      Y=N+X ;  
10 -     plot(Y) ;  
11 -     ecdf(Y)
```

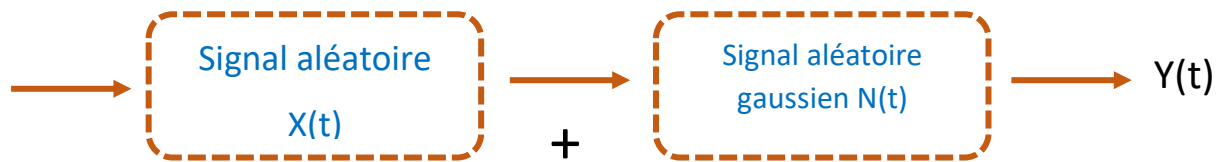
Code qui permet de générer un signal aléatoire  $Y = X + N$

### Interprétation:

On va tout simplement additionner notre signal aléatoire  $X$  + le deuxième signal  $N$ , on les affecter à  $Y$ , le résultat obtenu est représenté dans la figure ci-dessous :



Représentation graphique du signal  $Y f(y)$



### Règle de décision

Puisque le signal émis  $X$  et le signal reçu bruité  $Y$  sont complexes donc chacun a une partie imaginaire et une partie réelle. Afin de décider si le signal reçu est correcte ou bien il présente une erreur il suffit de savoir le signe de la partie réelle (éventuellement la partie imaginaire) du signal émis et du signal reçu ; s'ils sont de même signe alors il n'y a pas d'erreurs.

```
tp2.m*  ✕  +  
9 - SNRdb=0:2:10;  
10 - SNR=10.^(SNRdb/10);  
11 - S=1./sqrt(SNR);  
    plot(SNR);
```



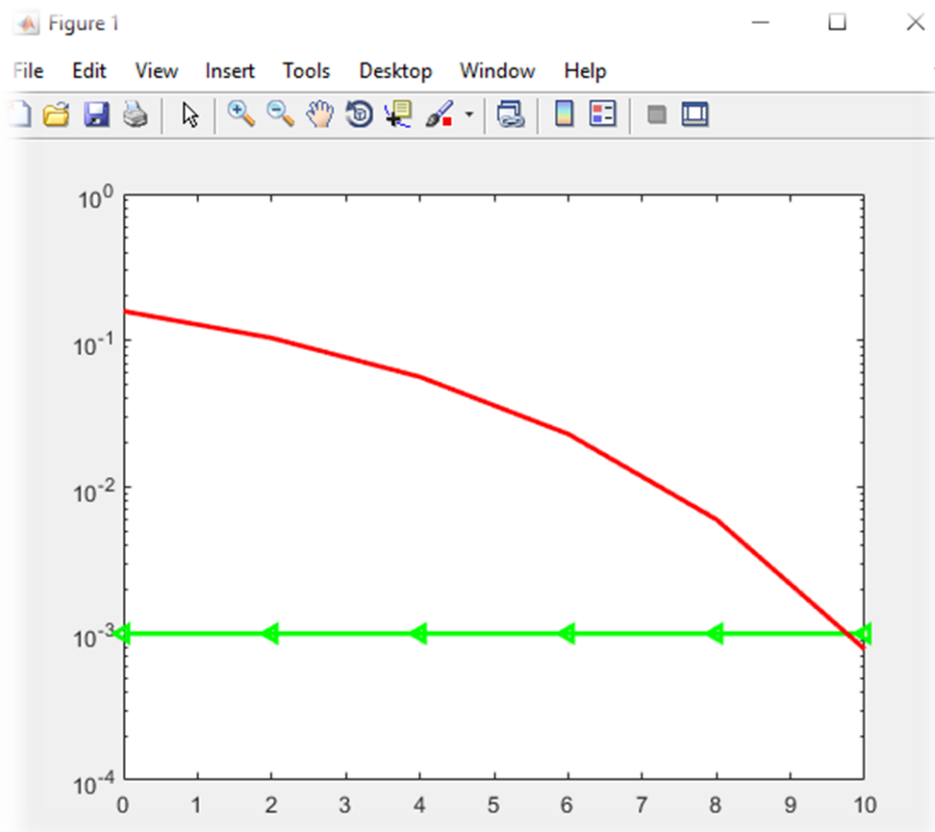
Code permettant de calculer le nombre d'erreurs BER en fonction du rapport signal à bruit SNR.

```

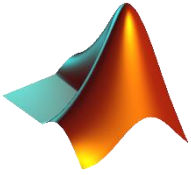
16 - SNRdb=0:2:10;
17 - SNR=10.^(SNRdb/10);
18 - S=1./sqrt(SNR);
19 - semilogy(SNRdb,SER,'-g.','LineWidth',0.5)
20 - hold on
21 - semilogy(SNRdb,2*qfunc(sqrt(SNR)),'-b.','LineWidth',0.5)

```

Représentation du BER en fonction du SNR variant de 0dB à 10dB  
avec un pas de 2 dB



Représentation graphique de SER en fonction de SNR



## Code complet

```
n=10^3;
X =2*(round(rand(1,n)))-1;
plot(X) ecdf(X);
sigma=5;
N=sigma*randn(1,n);
plot (N);
ecdf(N);
Y = X+N ;
plot (Y);
ecdf (Y);
BER = sum((Y.*X)<0)/n ;
SNR = 1 ./ (sigma^2) ;
SNRdb=0:2:10;
NR=10.^(SNRdb/10);
S=1./sqrt(SNR);
plot(SNR);
for j=1:length(SNR);
    x=2*round(rand(1,n))-1;
    y=(sqrt(SNR(j))*x)+randn(1,n);
    BER(j)=length(sum((Y.*X)<0)/n) ;
end
BER=BER/n;
semilogy(SNRdb,BER,'g-<', 'linewidth',2.0 );
hold on;
semilogy(SNRdb,qfunc(sqrt(SNR)), 'r-', 'linewidth',2.0);
```