

---

## Table of Contents

.....	1
Projet Télécommunications .....	1
Mahmoud LAANAIYA - Mohamed Hamza KADRI .....	1
Données .....	1
Sans Égalisation .....	1
Avec Égalisation .....	8

```
clear all;
close all;
clc;
```

## Projet Télécommunications

## Mahmoud LAANAIYA - Mohamed Hamza KADRI

## Données

Tests

```
Donnees = [1 1 1 0 0 1 0];
Dirac_test = [1 0 0 0 0 0 0];
% Donnees
Aleatoire = randi([0 1], 1, 10000);
Aleatoire_eg = randi([0 1], 1, 10000);
Dirac = [1 zeros(1, length(Aleatoire_eg)-1)];
Rb = 3000;
Fe = 24000;
Te = 1/Fe;
Ns = floor(Fe/Rb);
Ts = Ns*Te;
M = 2;
alpha0 = 1;
alpha1 = 0.5;
h = ones(1, Ns);
hc = alpha0*[1, zeros(1, Ns)] + alpha1*[zeros(1,Ns), 1];
eb_N0_db = [0 : 10];
precision = 100;
```

## Sans Égalisation

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Étude sans canal %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

---

```

[TEB_sans_canal, ~, ~, ~] = BPSK(Aleatoire, Ns, hc, h, 0); % TEB en
cas général
fprintf('\n***** Calcul du TEB sans canal *****\n');
fprintf('\nLa valeur du TEB sans canal est égale à : %d\n',
    TEB_sans_canal);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Étude sans bruit avec canal %%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
[TEB_canal, ~, ~, ~] = BPSK(Aleatoire, Ns, hc, h, 1); % Pour calculer
le TEB en cas général
fprintf('\n***** Calcul du TEB avec canal *****\n');
fprintf('\nLa valeur du TEB avec canal est égale à : %d\n',
    TEB_canal);

% Pour l'information binaire 1110010
[~, mapping, X, X_ech] = BPSK(Donnees, Ns, hc, h, 1);
% Constellations
scatterplot(mapping);
title("Les constellations en sortie du mapping");
scatterplot(X_ech);
title("Les constellations en sortie de l'échantillonneur");

% Les fonctions g(t) et z(t)
g = conv(h, conv(h, hc));
figure, plot(g);
title("La fonction g(t)");
figure, plot(X);
title("La fonction z(t)");
% Réponse impulsionnelle de la chaine de transmission
figure, plot(X_ech);
title("La réponse impulsionnelle de la chaine de transmission
    échantillonnée à N_{s}");
% Diagramme de l'Oeil
figure, plot(reshape(X, Ns, length(X)/Ns));
title("Diagramme de l'Oeil sans bruit");

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Étude avec bruit %%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% TEB simulée
[TEB_bruit_sans_canal, ~, ~] = Calcul_TEB_Bruit(precision, eb_N0_db,
    Aleatoire, Ns, hc, h, M, 0);
[TEB_bruit_avec_canal, mapping, X_ech] = Calcul_TEB_Bruit(precision,
    eb_N0_db, Aleatoire, Ns, hc, h, M, 1);
% TEB théorique
TEB_th = (1/2)*(qfunc(sqrt(10.^(eb_N0_db/10)/5)) +
    qfunc(sqrt(9*10.^(eb_N0_db/10)/5)));

% Comparaisons
figure();
semilogy(eb_N0_db, TEB_th, 'DisplayName', 'TEB théorique'); hold on
semilogy(eb_N0_db, TEB_bruit_avec_canal, 'DisplayName', 'TEB simulé');
legend;
title('Comparaison entre TEB simulée avec canal et TEB théorique');

```

---

---

```

hold off
figure();
semilogy(eb_N0_db, TEB_bruit_avec_canal, 'DisplayName', 'TEB avec
    canal');hold on
semilogy(eb_N0_db, TEB_bruit_sans_canal, 'DisplayName', 'TEB sans
    canal');
legend;
title('Comparaison entre TEB avec canal et TEB sans canal');
hold off

% Constellations
scatterplot(mapping);
title("Les constellations en sortie du mapping");
scatterplot(X_ech);
title("Les constellations en sortie de l'échantillonneur");

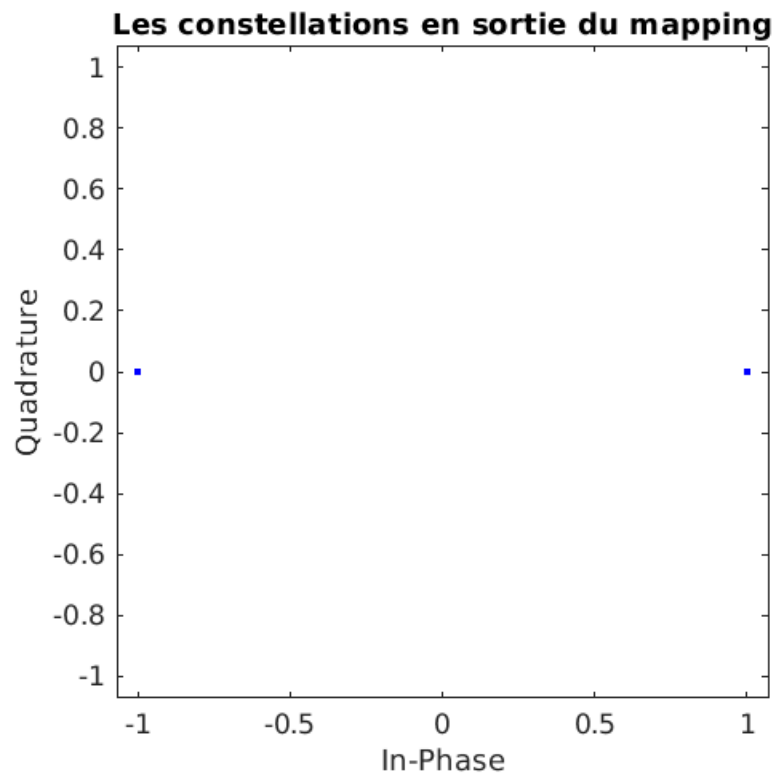
***** Calcul du TEB sans canal *****

La valeur du TEB sans canal est égale à : 0

***** Calcul du TEB avec canal *****

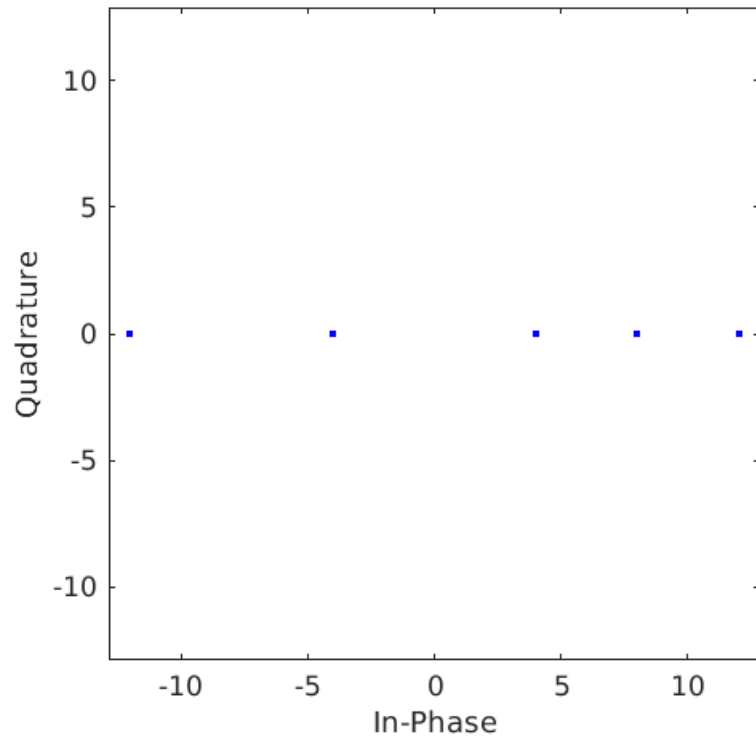
La valeur du TEB avec canal est égale à : 0

```

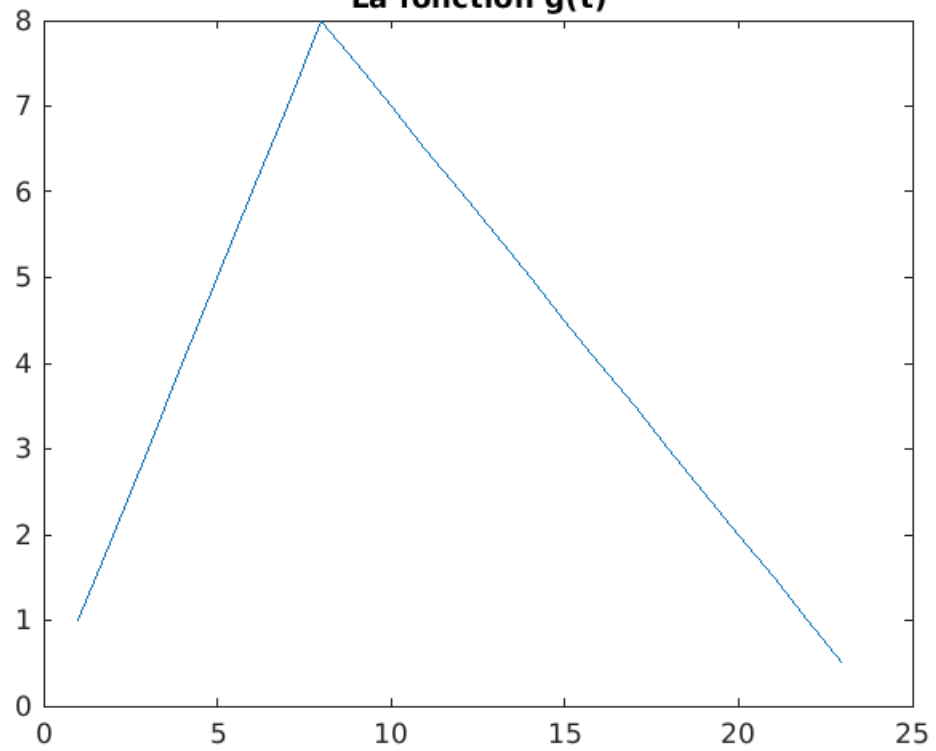


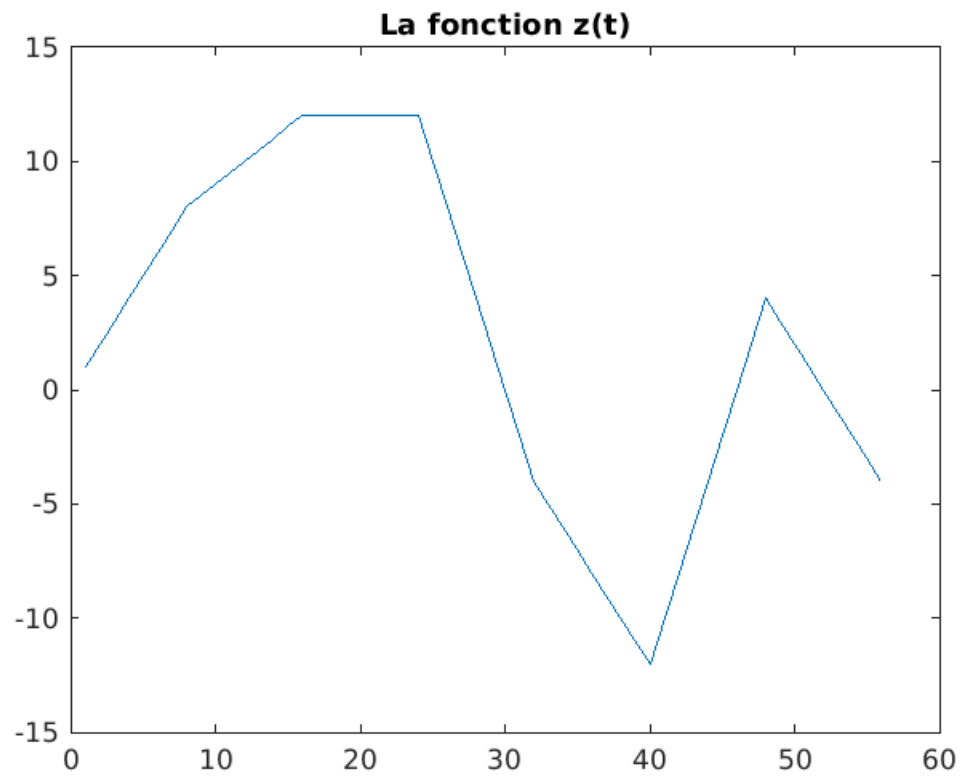
---

**Les constellations en sortie de l'échantillonneur**

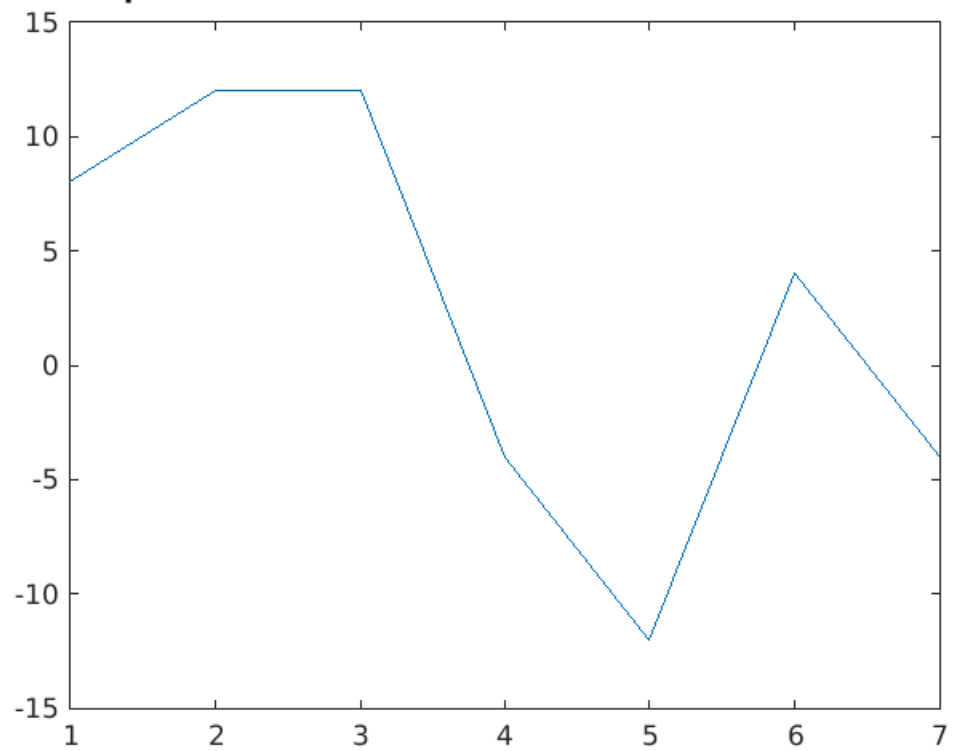


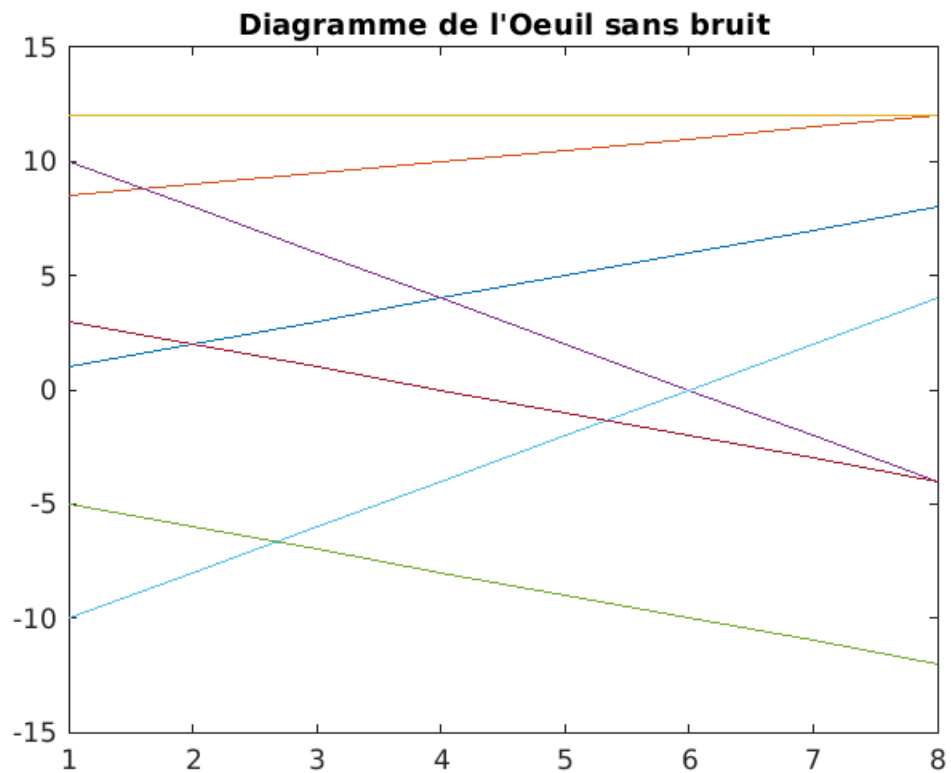
**La fonction  $g(t)$**



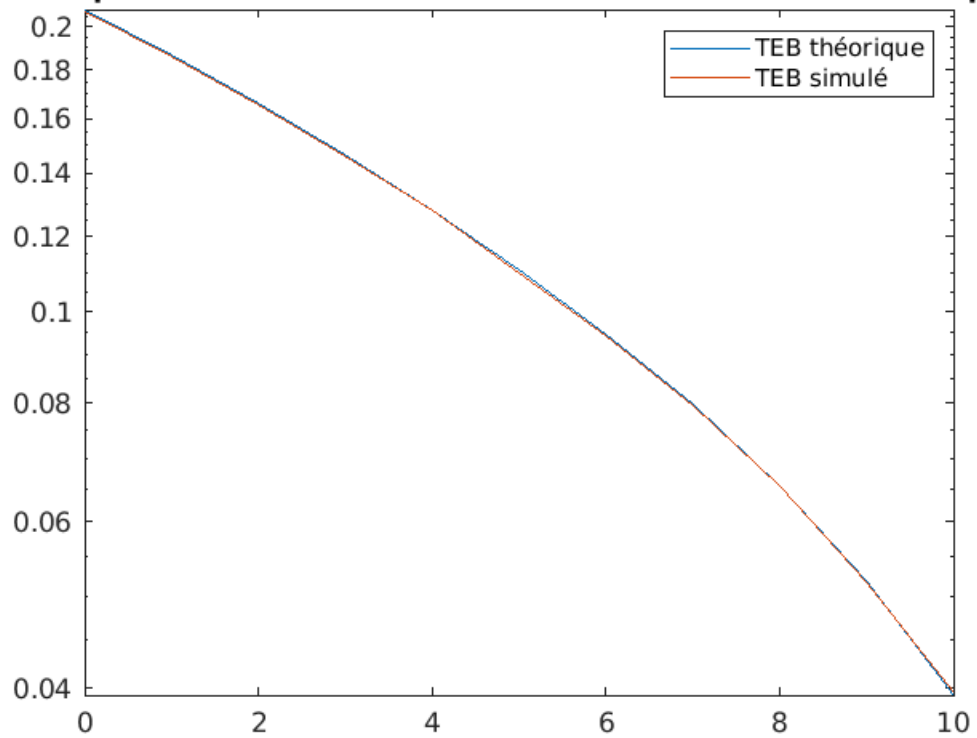


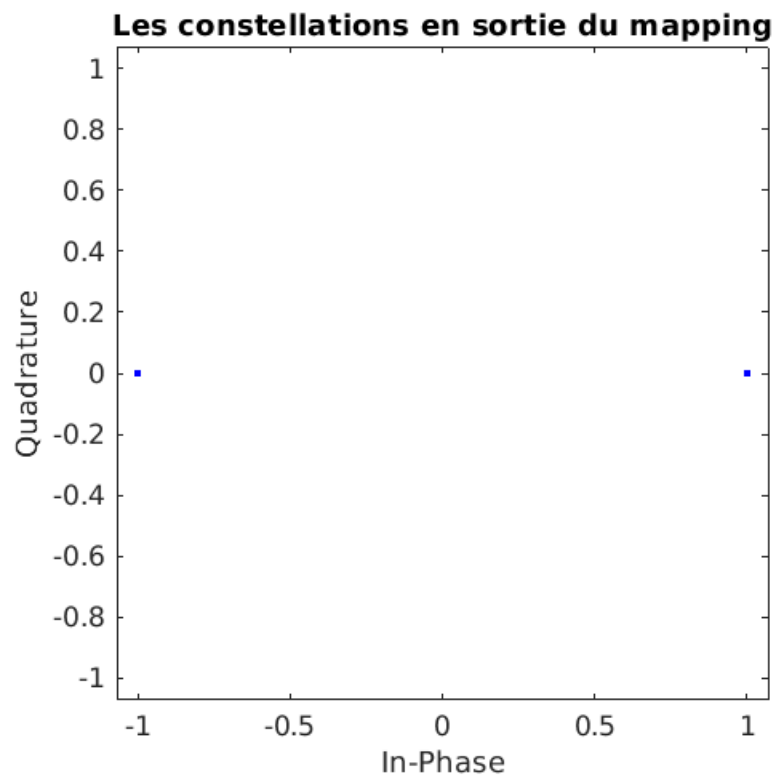
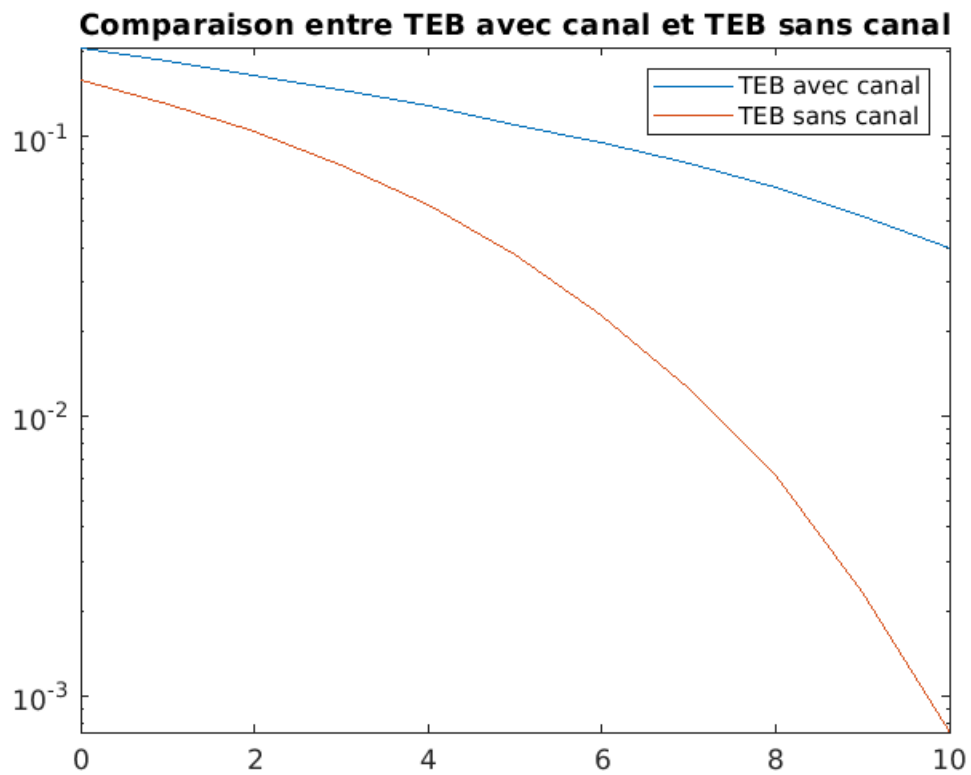
**réponse impulsionnelle de la chaine de transmission échantillonnée**

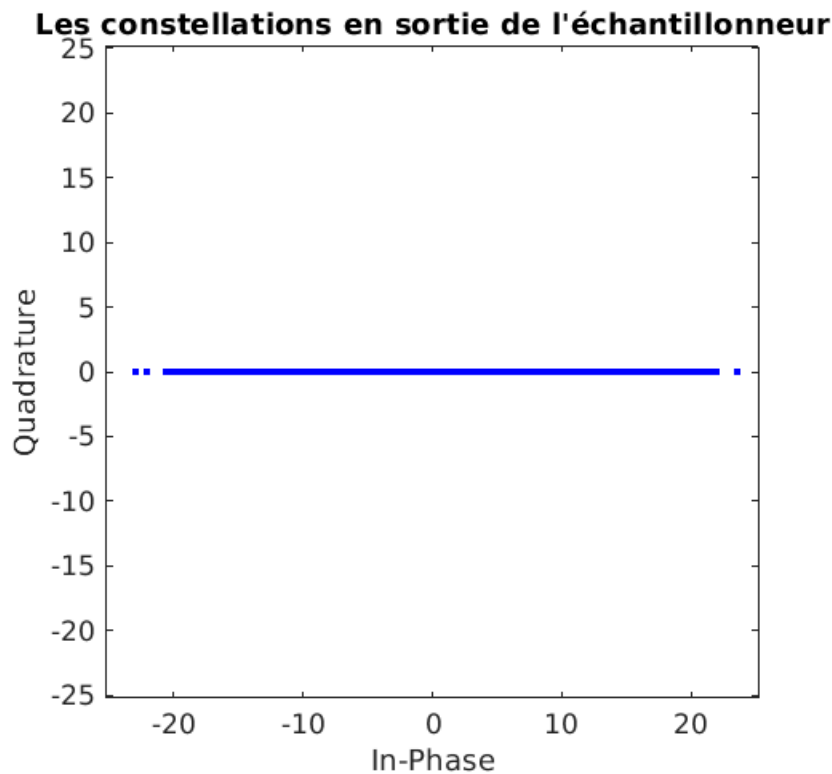




**Comparaison entre TEB simulée avec canal et TEB théorique**







## Avec Égalisation

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Sans Bruit %%%%%%%%%
%%%%%%%%

% Calcul des coefs de l'égaliseur

[~, ~, ~, X_ech] = BPSK(Dirac_test, Ns, hc, h, 1);
% Création de la matrice
Z = toeplitz(X_ech);
K = length(X_ech);
Y0 = [1;zeros(K-1,1)];

% Coefs de l'égaliseur
% N = K, matrice carré, on utilise l'inverse
C_test = Z\Y0; % Z\Y0 = inv(Z)*Y0

% Réponses en fréquences des filtres
Normalise_hc = abs(fft(hc,2^11)); % |Hc(f)|
Normalise_hc = Normalise_hc/max(Normalise_hc); % Normaliser
Normalise_hcg = abs(fft(C_test,2^11)); % |Heg(f)|
Normalise_hcg = Normalise_hcg/max(Normalise_hcg); % Normaliser
Normalise_produit = abs(fft(conv(hc,C_test),2^11)); % |Heg(f)*Hc(f)|
Normalise_produit = Normalise_produit/max(Normalise_produit); %
Normaliser

```



---

```

I = linspace(-Fe/2,Fe/2,length(Normalise_produit));
figure();
plot(I,fftshift(Normalise_produit), "DisplayName", "|H_{c}(f)H_{eg}(f)|");
title("|H_{c}(f)H_{eg}(f)|");
figure();
plot(I, fftshift(Normalise_hc), "DisplayName", "|H_{c}(f)|");
title("|H_{c}(f)|");
figure();
plot(I, fftshift(Normalise_hcg), "DisplayName", "|H_{eg}(f)|");
title("|H_{eg}(f)|");

% Les réponses sur une même figure
figure();
hold on
plot(I,fftshift(Normalise_produit), "DisplayName", "|H_{c}(f)H_{eg}(f)|");
plot(I, fftshift(Normalise_hc), "DisplayName", "|H_{c}(f)|");
plot(I, fftshift(Normalise_hcg), "DisplayName", "|H_{eg}(f)|");
hold off
legend;

% Étude avec une information binaire sans bruit avec égalisation
[~, mapping, X, X_ech] = BPSK(Donnees, Ns, hc, h, 1);
X_ech = filter(C_test', 1, X_ech);
decision = sign(X_ech);
demapping = (decision + 1)/2;
test = Donnees - demapping ;
TEB_eg = length(find(test~=0))/length(Donnees);
fprintf('\n***** Calcul du TEB avec égaliseur *****\n');
fprintf('\nLa valeur du TEB avec égaliseur est égale à : %d\n',
    TEB_eg);

scatterplot(mapping);
title("Les constellations en sortie du mapping avec égalisation");
scatterplot(X_ech);
title("Les constellations en sortie de l'échantillonneur avec
    égalisation");

% Réponse impulsionnelle de la chaine de transmission
figure,plot(X_ech);
title("La réponse impulsionnelle de la chaine de transmission
    échantillonnée à N_{s} avec égalisation");
% Diagramme de l'Oeil
figure,plot(reshape(X,Ns,length(X)/Ns));
title("Diagramme de l'Oeil avec égalisation");

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Avec bruit %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Calcul des coefs de l'égaliseur

[~, ~, ~, X_ech] = BPSK(Dirac, Ns, hc, h, 1);

```

---

---

```

% Création de la matrice
Z = toeplitz(X_ech);
K = length(X_ech);
Y0 = [1;zeros(K-1,1)];

% Coefs de l'égaliseur
% N = K, matrice carré, on utilise l'inverse
C = Z\Y0; % Z\Y0 = inv(Z)*Y0

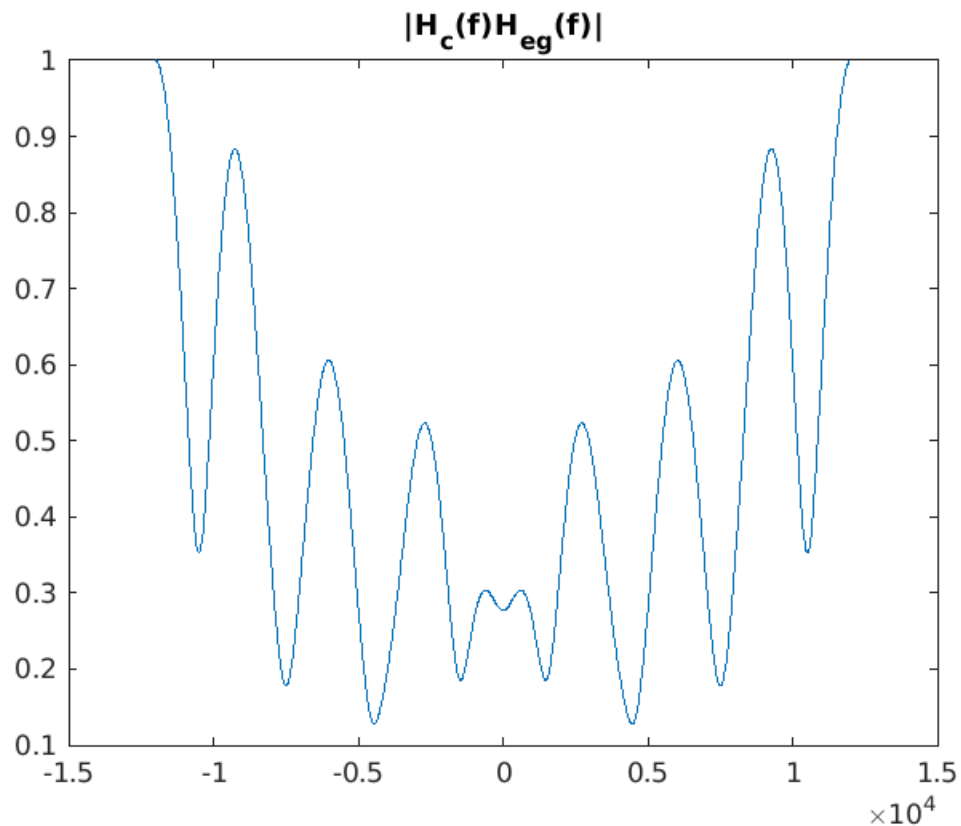
% Calcul du TEB avec bruit
[TEB_bruit_avec_egaliseur, ~, ~] = Calcul_TEB_Egaliseur(precision,
    eb_N0_db, Aleatoire_eg, Ns, hc, h, M, C');

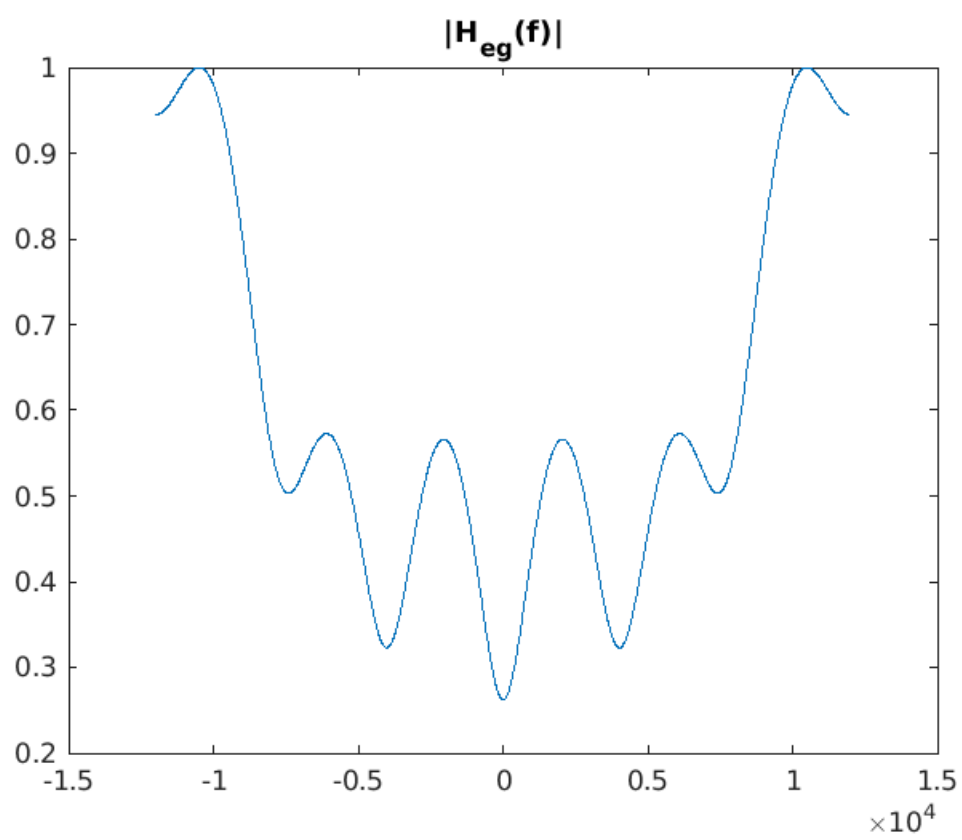
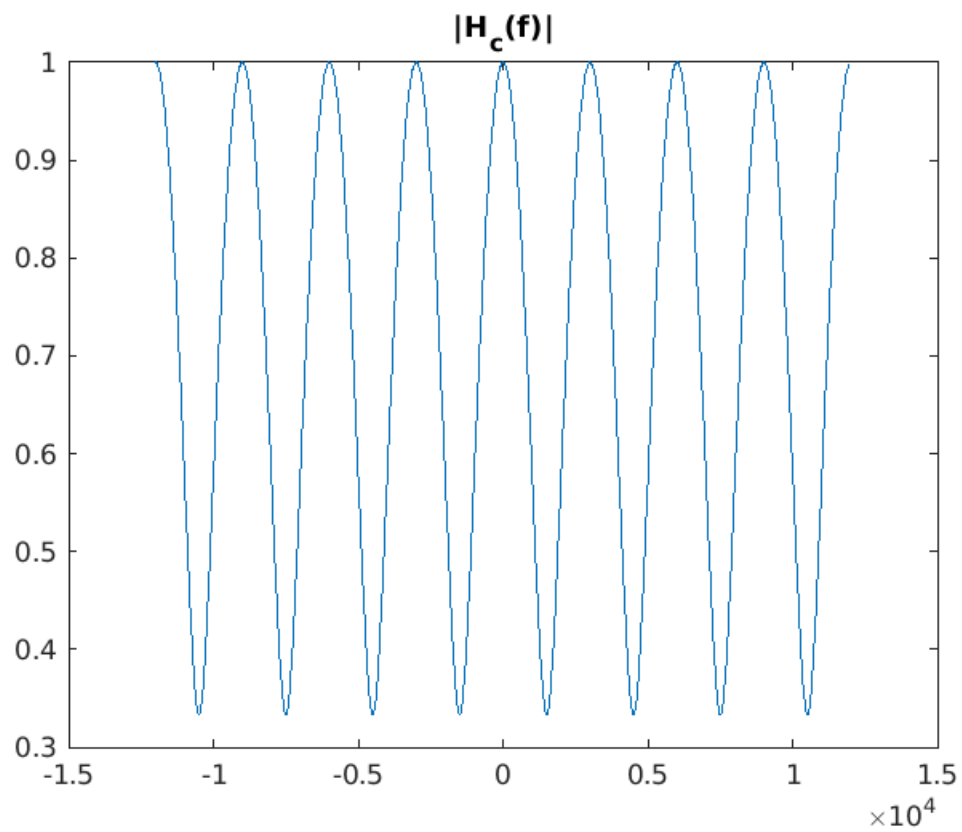
figure();
semilogy(eb_N0_db, TEB_bruit_avec_canal, 'DisplayName', 'TEB sans
    égaliseur'); hold on
semilogy(eb_N0_db, TEB_bruit_avec_egaliseur, 'DisplayName', 'TEB avec
    égaliseur');
legend;
title('Comparaison TEB');
hold off

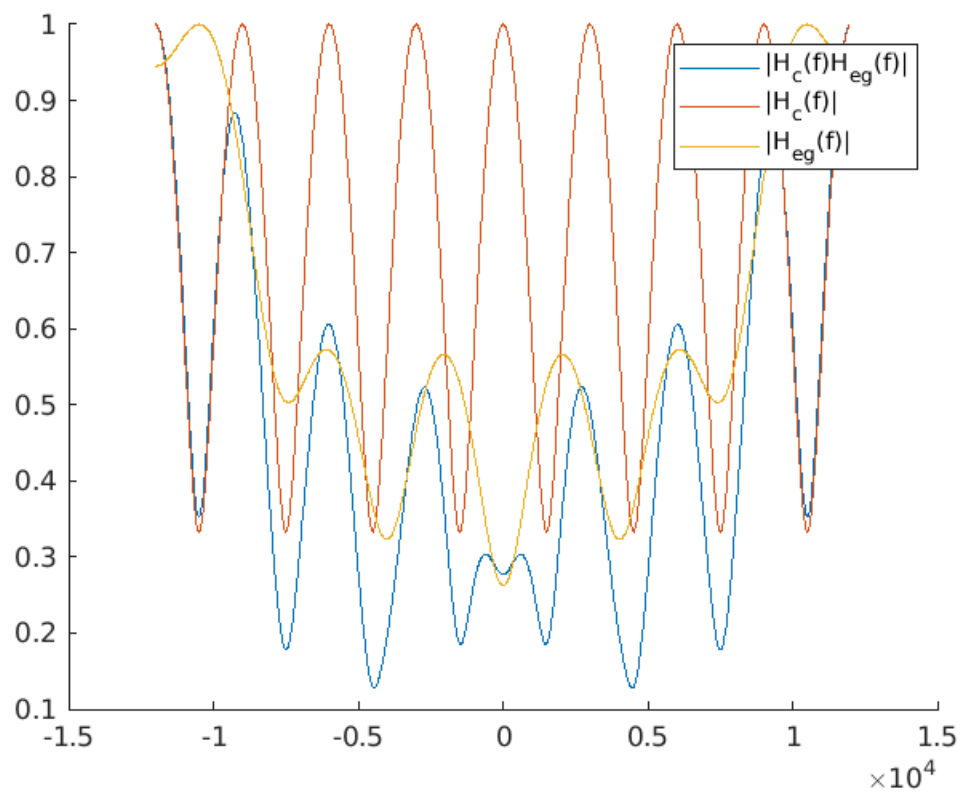
```

\*\*\*\*\* Calcul du TEB avec égaliseur \*\*\*\*\*

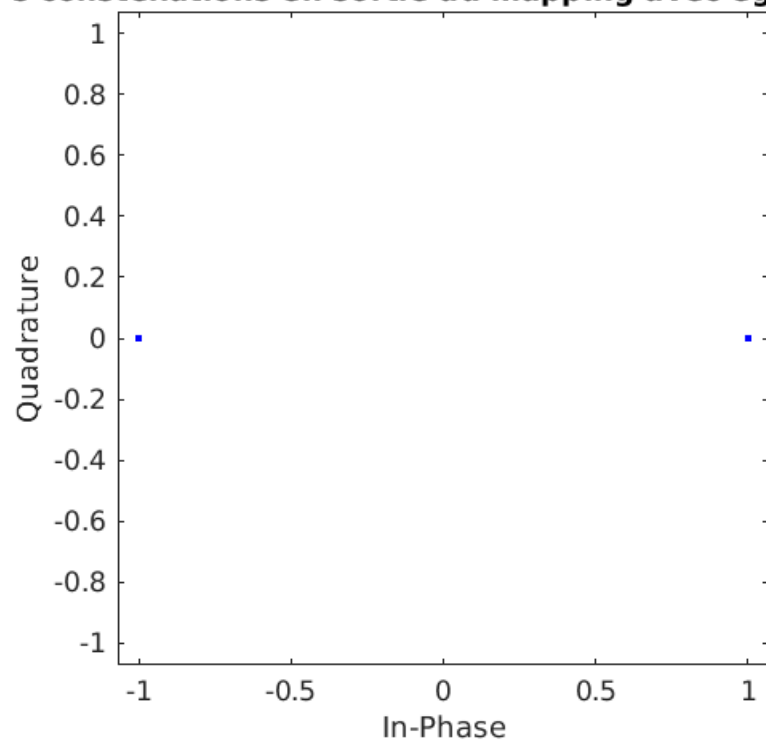
La valeur du TEB avec égaliseur est égale à : 0



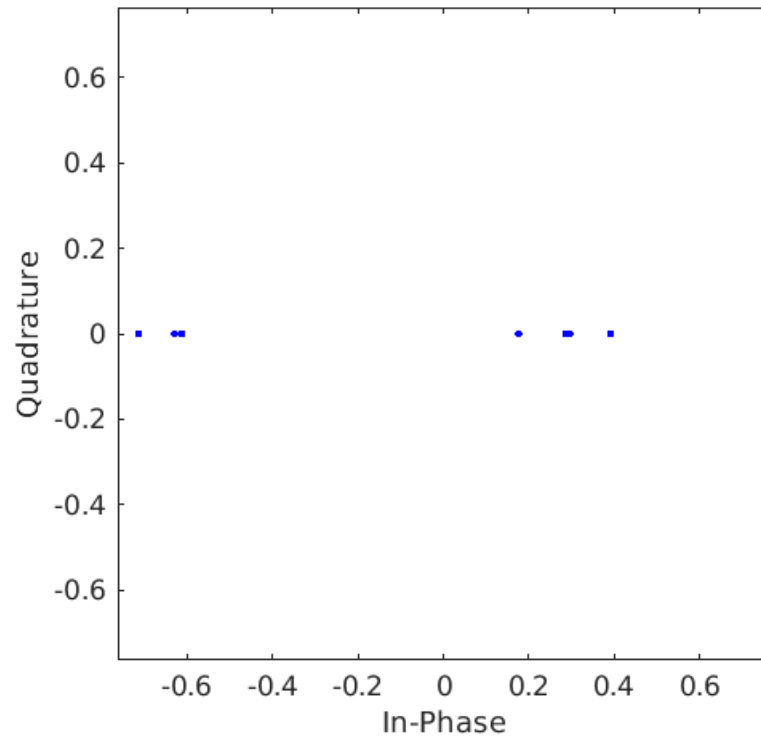




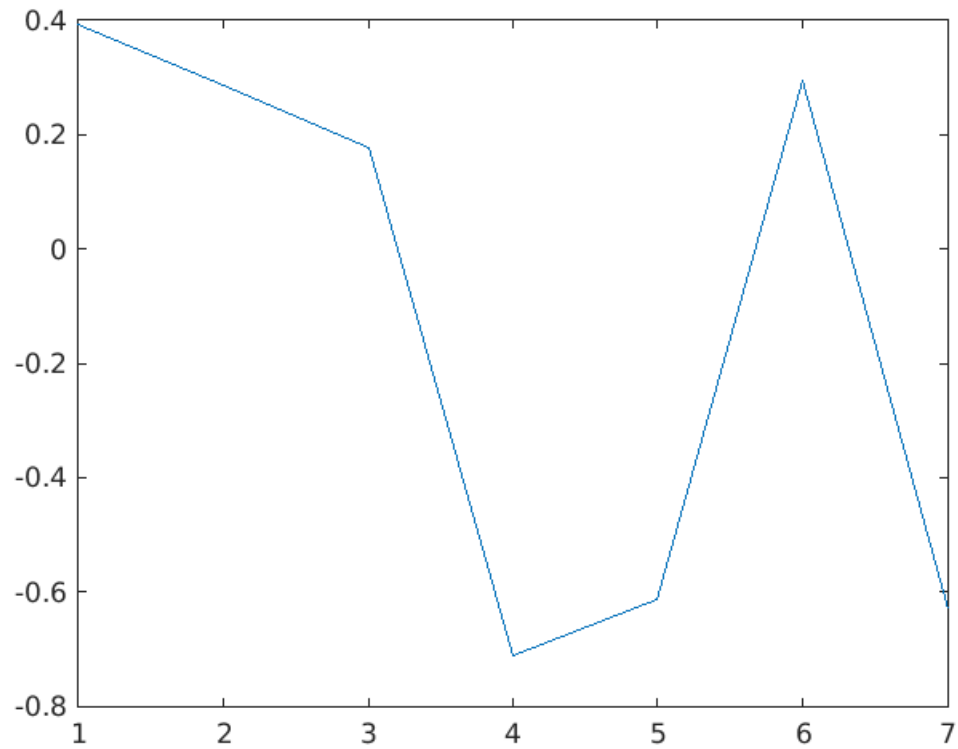
**s constellations en sortie du mapping avec égalisa**

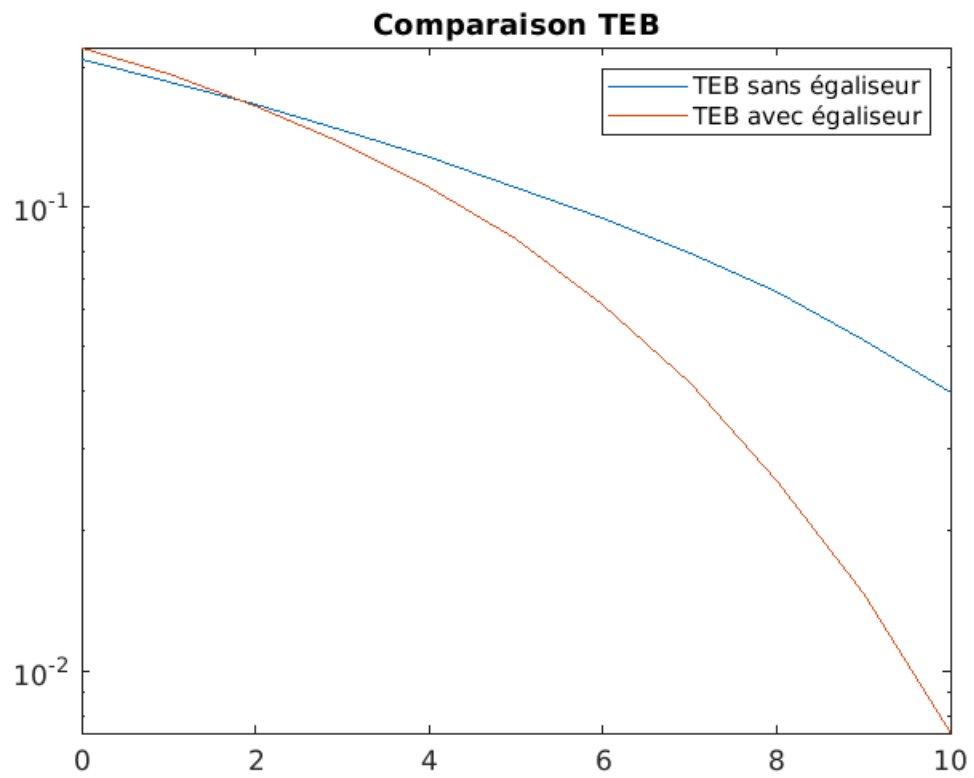
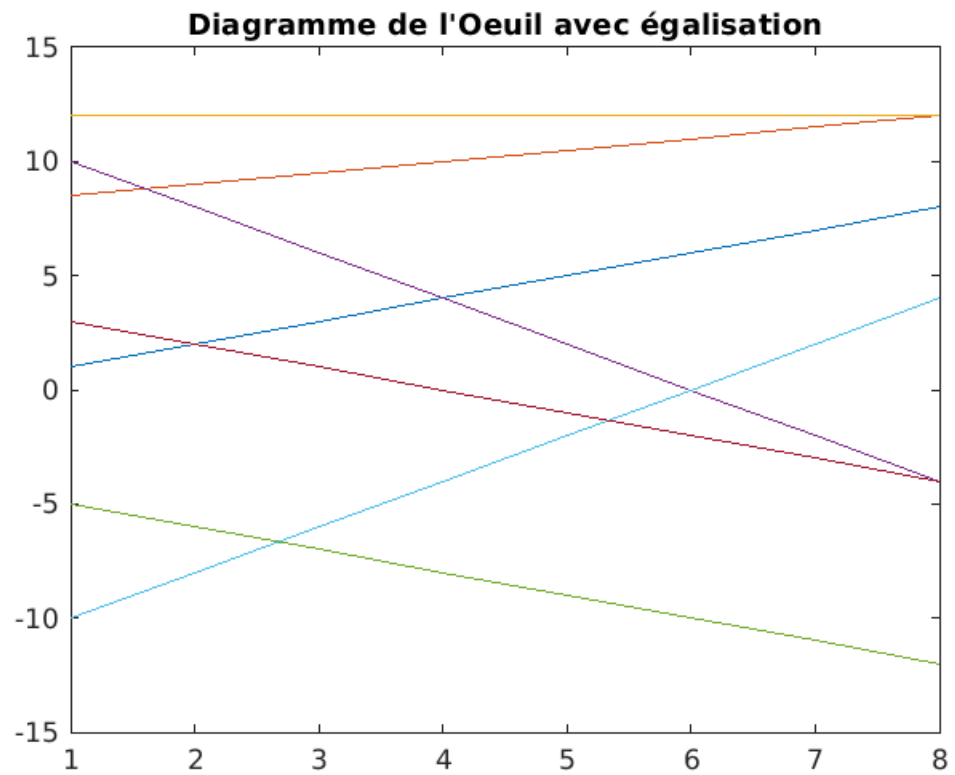


**Constellations en sortie de l'échantillonneur avec éga**



**Impulsionnelle de la chaine de transmission échantillonnée à  $N_s$  avec**





---

*Published with MATLAB® R2020a*