

---

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

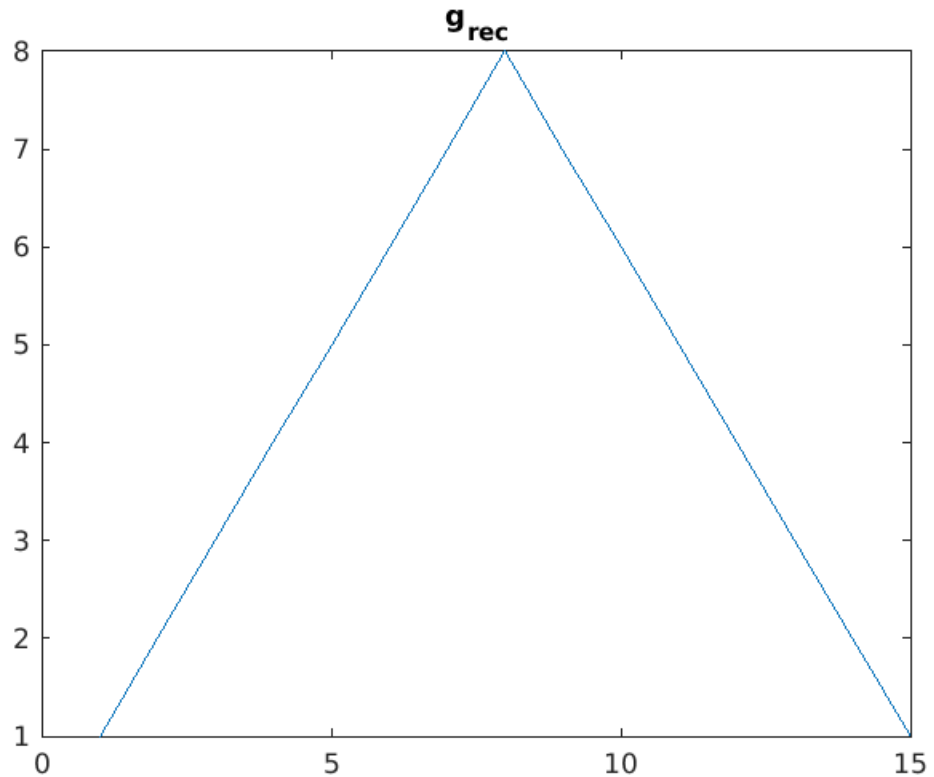
## Étude sans canal de propagation

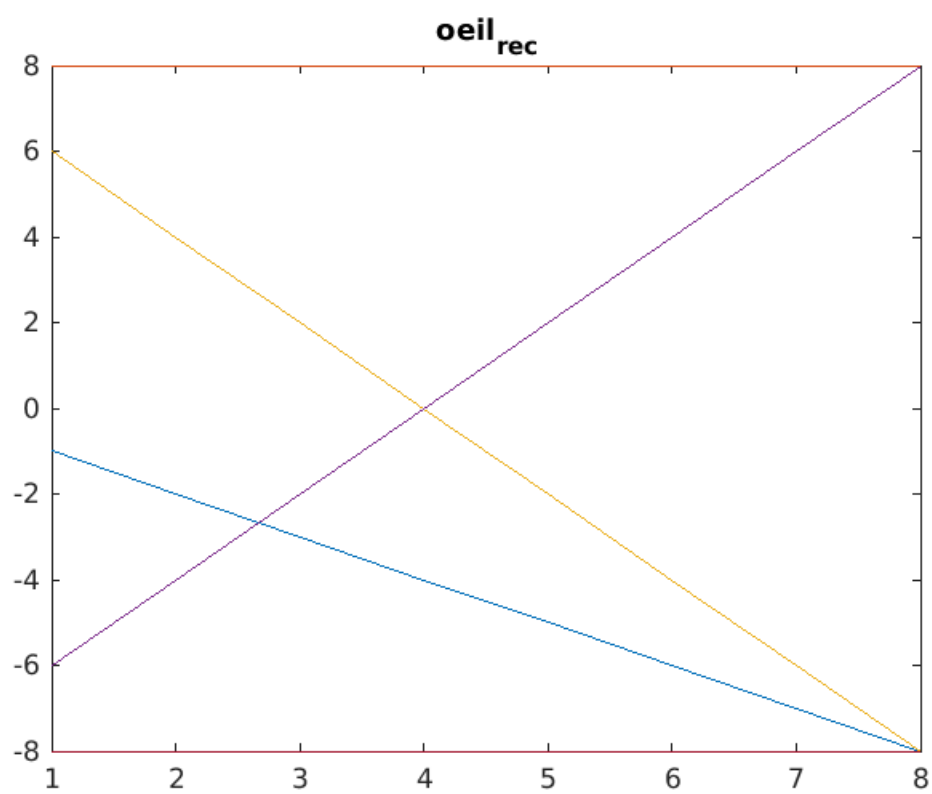
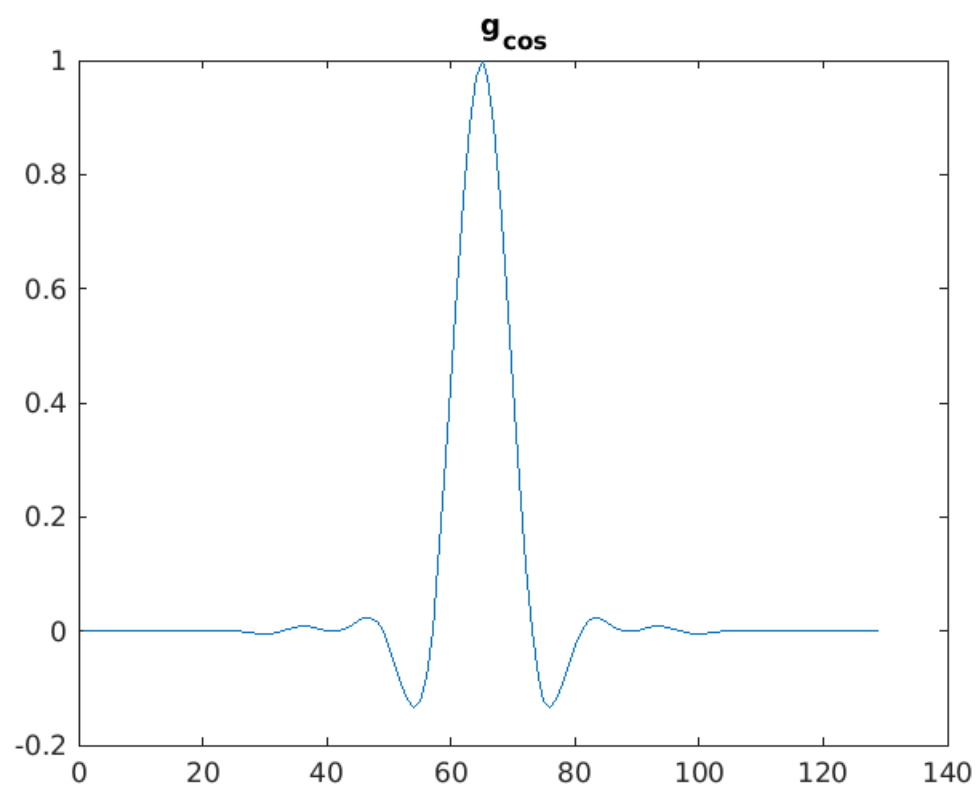
Bits

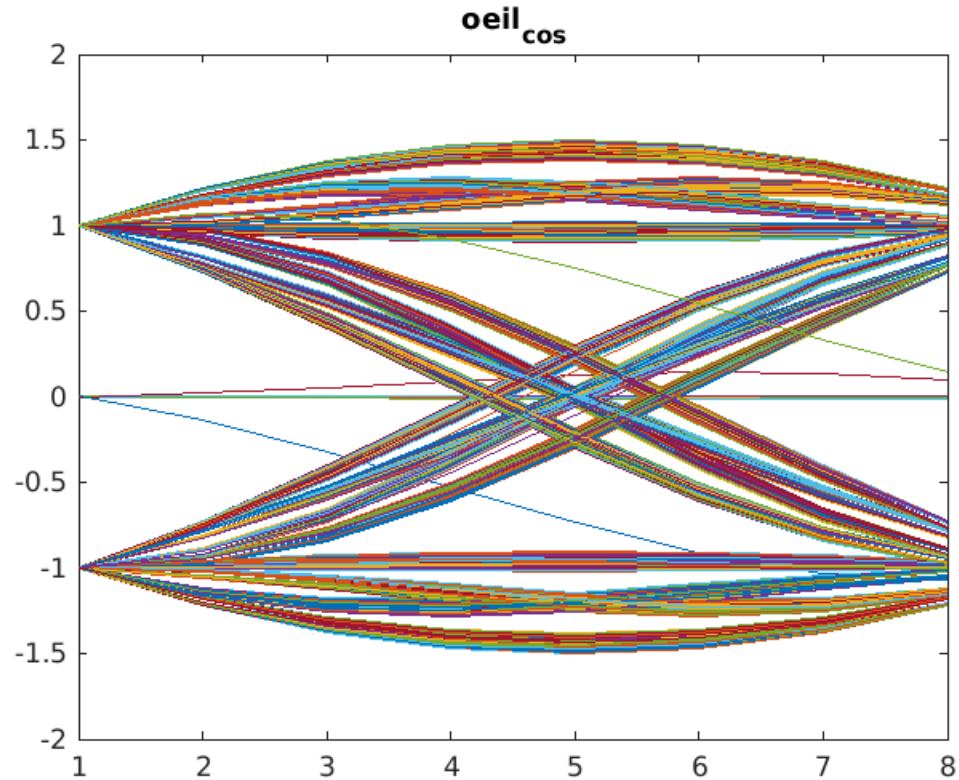
```
Donnees = randi([0 1], 1, 10000);
% Modulateur
% Mapping
mapping = Donnees - (Donnees == 0);
% Suréchantillonnage
Rb = 3000; % Débit binaire
M = 2; % M = 2^n (n=1)
Fe = 24000; % Fréquence d'échantillonnage
Te = 1/Fe;
Rs = Rb/log2(M); % Débit symbole
Ns = floor(Fe/Rs);
Ts = Ns*Te;
h_rec = ones(1, Ns); % Filtre rectangulaire
h_cos = rcosdesign(0.5,8,Ns); % Filtre en cosinus
% Filtrage
Kronr = kron(mapping, [1 zeros(1, Ns-1)]); % Positionnement des zéros
entre les ak
signalfiltre_rec = filter(h_rec, 1, Kronr); % Filtrage avec filtre
rectangulaire
signalfiltre_cos = filter(h_cos, 1, [Kronr zeros(1,Ns*8)]); % Filtrage
avec filtre en cosinus
% Demodulateur
% Filtrage de réception
hr_rec = fliplr(h_rec); % filtre rectangulaire
z_rec = filter(hr_rec, 1, signalfiltre_rec); % signal filtré par le
filtre rectangulaire
hr_cos = fliplr(h_cos); % filtre en cosinus
z_cos = filter(hr_cos, 1, signalfiltre_cos); % signal filtré par le
filtre en cosinus
% Echantillonnage
n0_rec = Ns; % n0 du rectangulaire
n0_cos = Ns*8+1; % n0 du cosinus
m = 1; % le pas
x_ech_rec = z_rec(n0_rec:Ns:end); % échantillonnage du rectangulaire
x_ech_cos = z_cos(n0_cos:Ns:end); % échantillonnage de celui en
cosinus
% Decision
Decision_rec = sign(x_ech_rec); % Décision du rectangulaire
Decision_cos = sign(x_ech_cos); % Décision de celui en cosinus
% Demapping
demapping_rec = (Decision_rec + 1)/2; % Démapping du rectangulaire
demapping_cos = (Decision_cos + 1)/2; % Démapping de celui en cosinus
% Convolution
```

---

```
g_rec = conv(h_rec, hr_rec); % Réponse impulsionnelle globale du
    réctangulaire
g_cos = conv(h_cos, hr_cos); % Réponse impulsionnelle globale du
    cosinus
figure,plot(g_rec);
title('g_{rec}');
figure,plot(g_cos);
title('g_{cos}');
figure,plot(reshape(z_rec,Ns,length(z_rec)/Ns));
title('oeil_{rec}');
figure,plot(reshape(z_cos,Ns,length(z_cos)/Ns));
title('oeil_{cos}');
% erreur binaire
erreur_rec = sum(abs(demapping_rec-Donnees))/length(Donnees); % Erreur
    du réctangulaire
erreur_cos = sum(abs(demapping_cos-Donnees))/length(Donnees); % Erreur
    du cosinus
```







## Étude avec canal de propagation

BW = 4000

```

BW_1 = 4000; % Fréquence de coupure
N_rec = length(hr_rec); % L'ordre du filtre rectangulaire
N_cos = length(hr_cos); % L'ordre du filtre en cosinus
I_rec = [(-N_rec+1)*Te/2 :Te: (N_rec-1)*Te/2]; % Interval discret
I_cos = [(-N_cos+1)*Te/2 :Te: (N_cos-1)*Te/2]; % Interval discret
hc_rec_1 = (2*BW_1/Fe)*sinc(2*BW_1*I_rec); % Réponse impulsionnelle du
    filtre passe-bas rectangulaire
hc_cos_1 = (2*BW_1/Fe)*sinc(2*BW_1*I_cos); % Réponse impulsionnelle de
    celui en cosinus
Normalise_produit_rec1 = abs(fft(conv(h_rec,hr_rec),2^11)); % |
H(f)*Hr(f)|
Normalise_produit_rec1 = Normalise_produit_rec1/
max(Normalise_produit_rec1); % Normaliser
Normalise_produit_cos1 = abs(fft(conv(h_cos,hr_cos),2^11)); % |
H(f)*Hr(f)|
Normalise_produit_cos1 = Normalise_produit_cos1/
max(Normalise_produit_cos1); % Normaliser
Normalise_hc_rec1 = abs(fft(hc_rec_1,2^11)); % |Hc(f)|
Normalise_hc_rec1 = Normalise_hc_rec1/max(Normalise_hc_rec1); %
    Normaliser
Normalise_hc_cos1 = abs(fft(hc_cos_1,2^11)); % |Hc(f)|

```

---

```

Normalise_hc_cos1 = Normalise_hc_cos1/max(Normalise_hc_cos1); %
    Normaliser
figure();
title("Comparaison de  $|H(f)H_r(f)|$  et  $|H_c(f)|$  rectangulaire avec
    BW = 4000");
hold on
I = linspace(-Fe/2,Fe/2,length(Normalise_produit_rec1));
plot(I,fftshift(Normalise_produit_rec1), "DisplayName", " $|H(f)H_r(f)|$ ");
plot(I, fftshift(Normalise_hc_rec1), "DisplayName", " $|H_c(f)|$ ");
hold off
legend;
figure();
title("Comparaison de  $|H(f)H_r(f)|$  et  $|H_c(f)|$  en cosinus avec BW
    = 4000");
hold on
I = linspace(-Fe/2,Fe/2,length(Normalise_produit_cos1));
plot(I,fftshift(Normalise_produit_cos1), "DisplayName", " $|H(f)H_r(f)|$ ");
plot(I, fftshift(Normalise_hc_cos1), "DisplayName", " $|H_c(f)|$ ");
hold off
legend;

% BW = 1000
BW_2 = 1000; % Fréquence de coupure
hc_rec_2 = (2*BW_2/Fe)*sinc(2*BW_2*I_rec); % Réponse impulsionnelle du
    filtre passe-bas rectangulaire
hc_cos_2 = (2*BW_2/Fe)*sinc(2*BW_2*I_cos); % Réponse impulsionnelle de
    celui en cosinus
Normalise_produit_rec2 = abs(fft(conv(h_rec,hr_rec),2^11)); % |
H(f)*Hr(f)|
Normalise_produit_rec2 = Normalise_produit_rec2/
max(Normalise_produit_rec2); % Normaliser
Normalise_produit_cos2 = abs(fft(conv(h_cos,hr_cos),2^11)); % |
H(f)*Hr(f)|
Normalise_produit_cos2 = Normalise_produit_cos2/
max(Normalise_produit_cos2); % Normaliser
Normalise_hc_rec2 = abs(fft(hc_rec_2,2^11)); %  $|H_c(f)|$ 
Normalise_hc_rec2 = Normalise_hc_rec2/max(Normalise_hc_rec2); %
    Normaliser
Normalise_hc_cos2 = abs(fft(hc_cos_2,2^11)); %  $|H_c(f)|$ 
Normalise_hc_cos2 = Normalise_hc_cos2/max(Normalise_hc_cos2); %
    Normaliser
figure();
title("Comparaison de  $|H(f)H_r(f)|$  et  $|H_c(f)|$  rectangulaire avec
    BW = 1000");
hold on
I = linspace(-Fe/2,Fe/2,length(Normalise_produit_rec2));
plot(I,fftshift(Normalise_produit_rec2), "DisplayName", " $|H(f)H_r(f)|$ ");
plot(I, fftshift(Normalise_hc_rec2), "DisplayName", " $|H_c(f)|$ ");
hold off
legend;
figure();

```

---

---

```

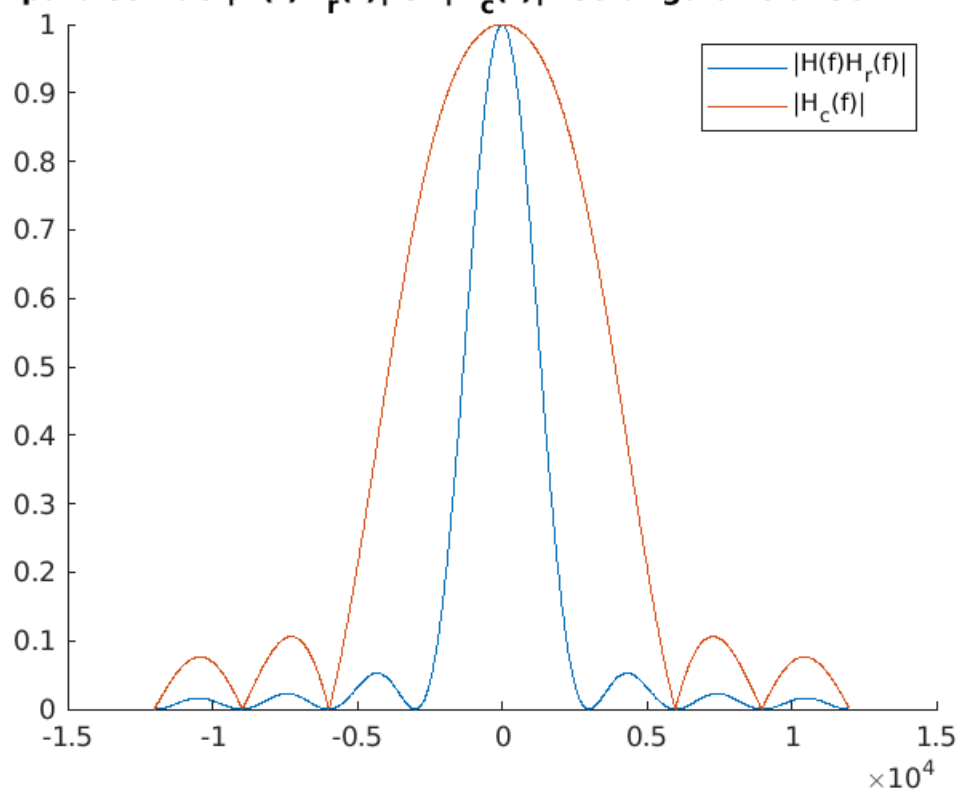
title("Comparaison de  $|H(f)H_r(f)|$  et  $|H_c(f)|$  en cosinus avec BW
      = 1000");
hold on
I = linspace(-Fe/2,Fe/2,length(Normalise_produit_cos2));
plot(I,fftshift(Normalise_produit_cos2), "DisplayName", " $|H(f)H_r(f)|$ ");
plot(I, fftshift(Normalise_hc_cos2), "DisplayName", " $|H_c(f)|$ ");
hold off
legend;

% Oeil
% BW = 4000
h_rec_canal1 = conv(hr_rec, hc_rec_1);
z_rec_canal1 = filter(h_rec_canal1, 1, signalfiltre_rec);
figure,plot(reshape(z_rec_canal1,Ns,length(z_rec_canal1)/Ns));
title('oeil rectangle 1');
h_cos_canal1 = conv(hr_cos, hc_cos_1);
z_cos_canal1 = filter(h_cos_canal1, 1, signalfiltre_cos);
figure,plot(reshape(z_cos_canal1,Ns,length(z_cos_canal1)/Ns));
title('oeil cosinus 1');

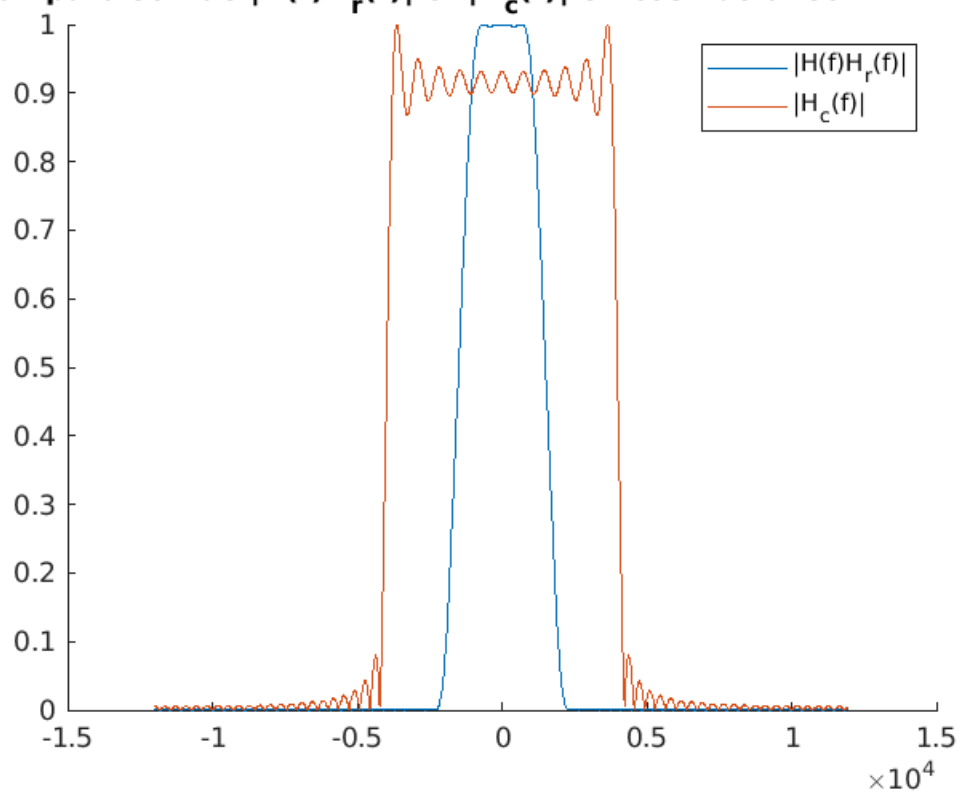
% BW = 1000
h_rec_canal2 = conv(hr_rec, hc_rec_2);
z_rec_canal2 = filter(h_rec_canal2, 1, signalfiltre_rec);
figure,plot(reshape(z_rec_canal2,Ns,length(z_rec_canal2)/Ns));
title('oeil rectangle 2');
h_cos_canal2 = conv(hr_cos, hc_cos_2);
z_cos_canal2 = filter(h_cos_canal2, 1, signalfiltre_cos);
figure,plot(reshape(z_cos_canal2,Ns,length(z_cos_canal2)/Ns));
title('oeil cosinus 2');

```

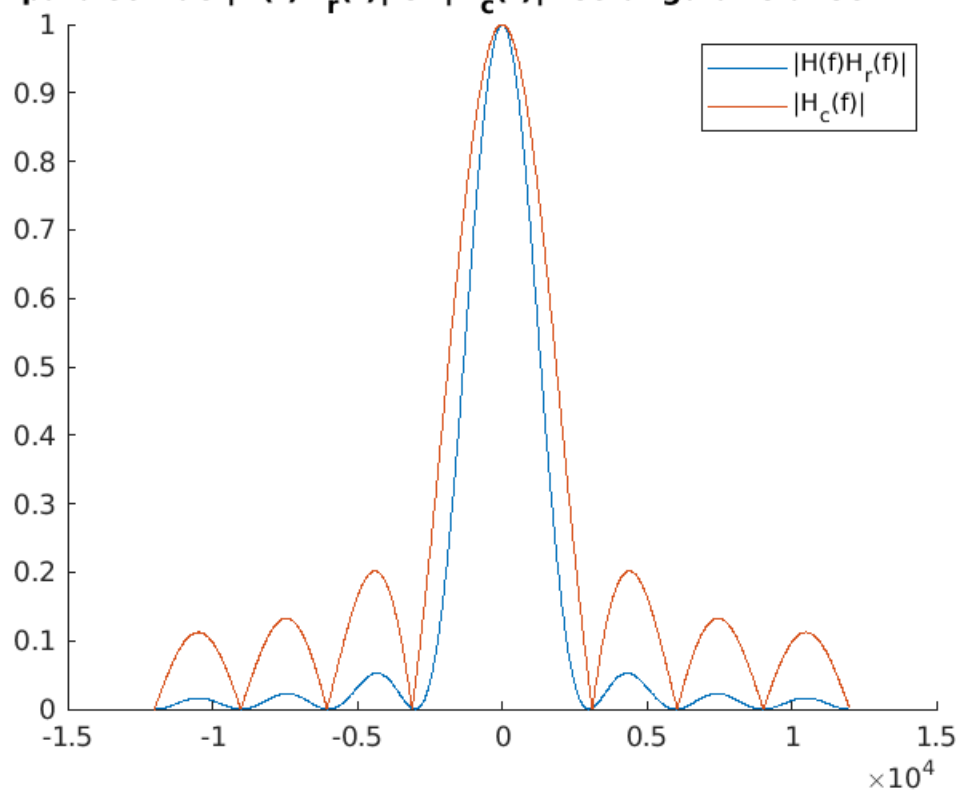
**Comparaison de  $|H(f)H_r(f)|$  et  $|H_c(f)|$  rectangulaire avec  $BW = 4000$**



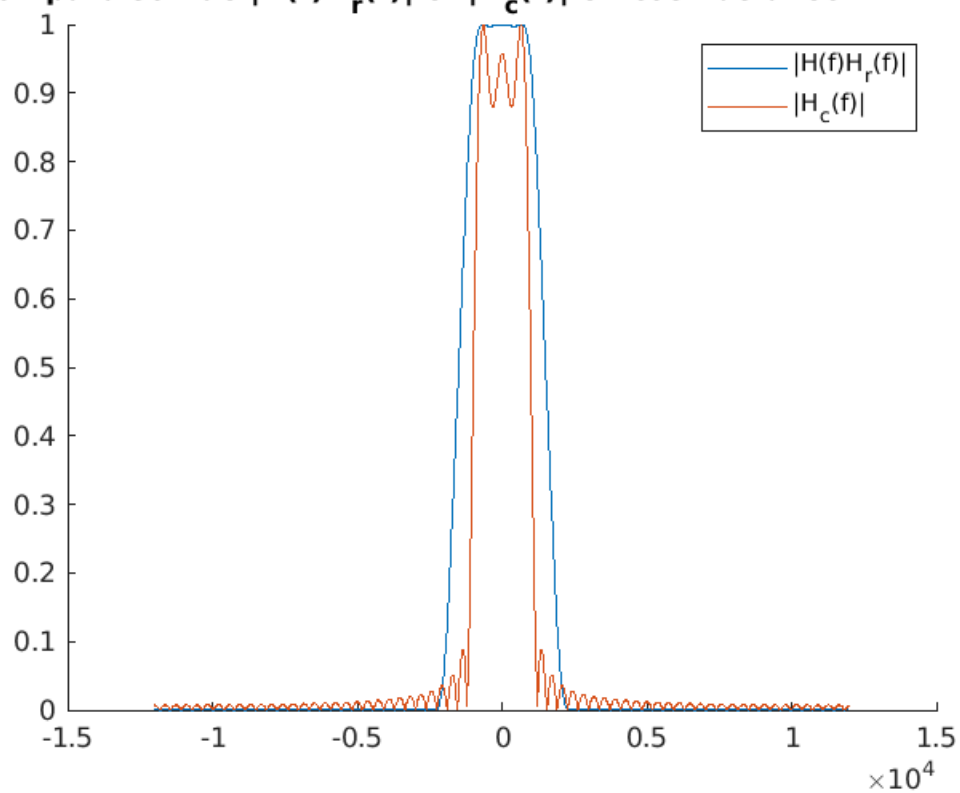
**Comparaison de  $|H(f)H_r(f)|$  et  $|H_c(f)|$  en cosinus avec  $BW = 4000$**



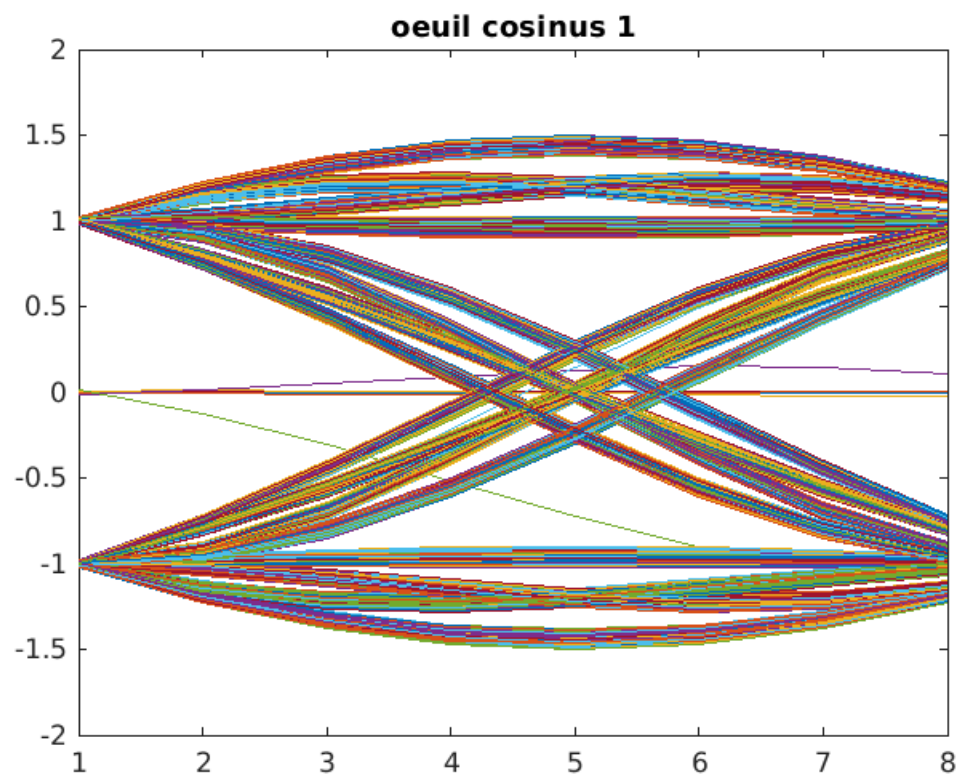
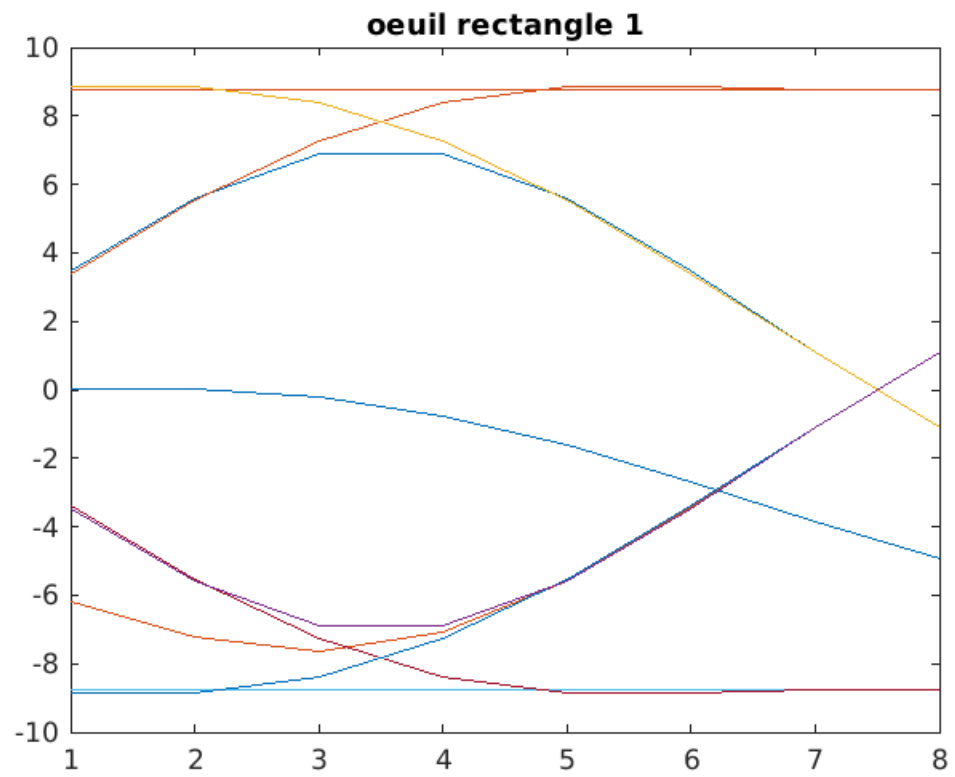
**Comparaison de  $|H(f)H_r(f)|$  et  $|H_c(f)|$  rectangulaire avec  $BW = 1000$**

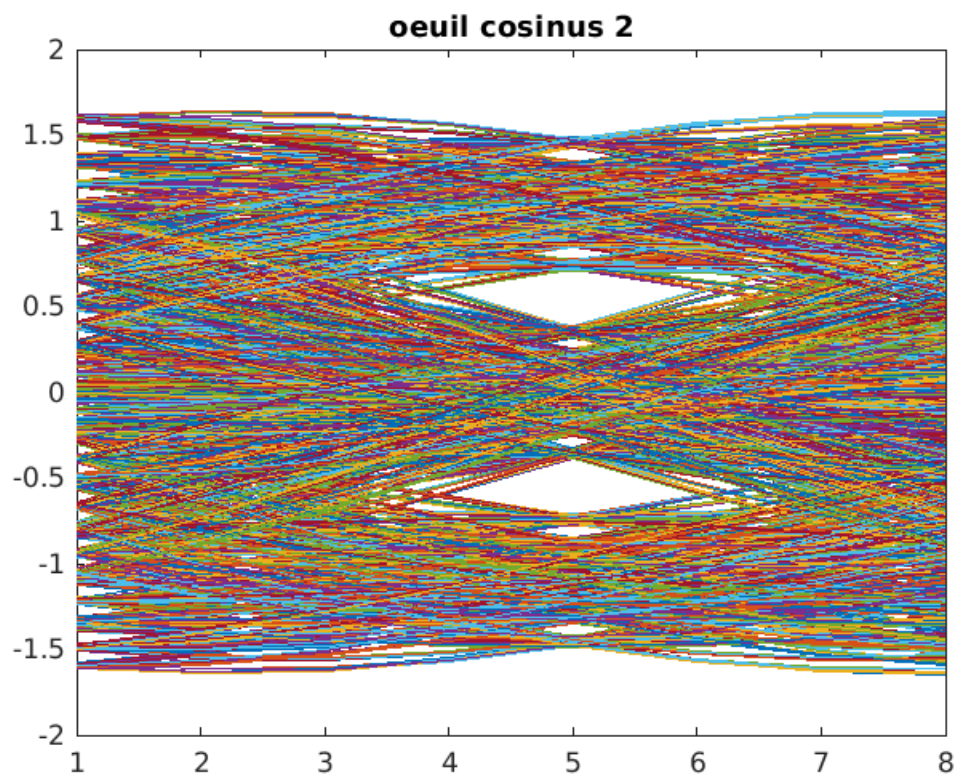
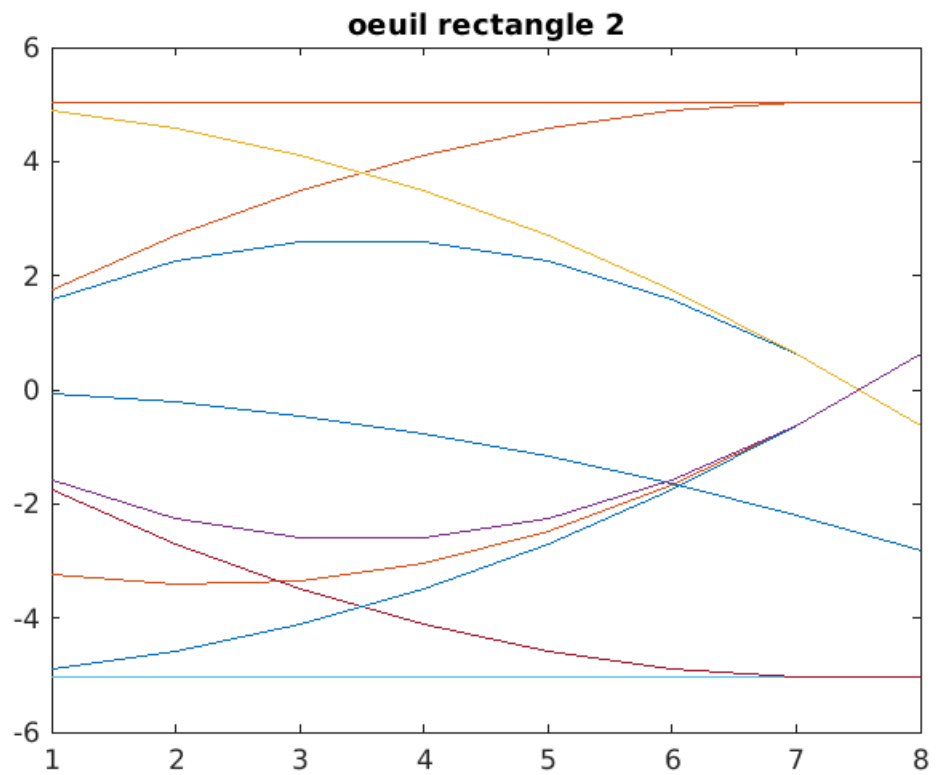


**Comparaison de  $|H(f)H_r(f)|$  et  $|H_c(f)|$  en cosinus avec  $BW = 1000$**









---

*Published with MATLAB® R2020a*