

# Calcul distance radar

```
% Commandes de remise à zéro de Matlab
close all; % Ferme toutes les fenêtres
clear;     % Efface toutes les variables
clc;       % Efface le contenu de la console
```

## 1 - Chargement des valeurs des signaux

```
Donnees_emission = load('z_SignalRadarEMIS.txt'); % Chargments des valeurs du signal émis
Donnees_reception = load('z_SignalEchoRadar2.txt'); % Chargments des valeurs du signal reçu
```

## 2 - Affectations des valeurs

### Création des vecteurs de données

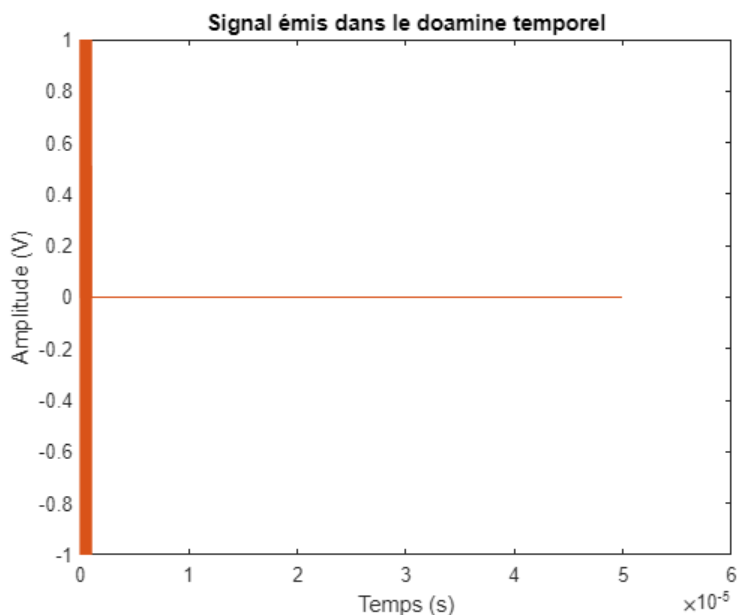
```
% Vecteur Temps
Temps = Donnees_emission(:,1); % Toutes les valeurs de la colonne 1

% Vecteur Signal radar émis
Signal_emis = Donnees_emission(:, 2);

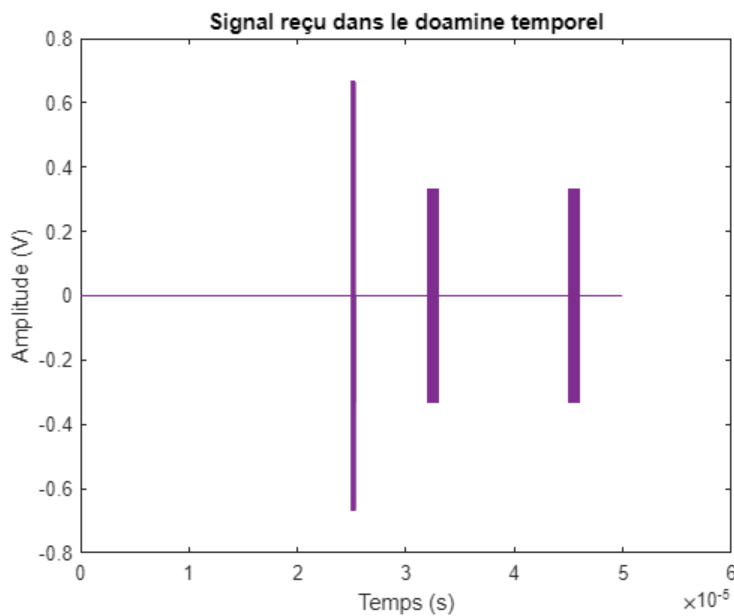
% Vecteur Signal radar reçu
Signal_recu = Donnees_reception(:,2);
```

### Affichage des signaux dans le domaine temporel

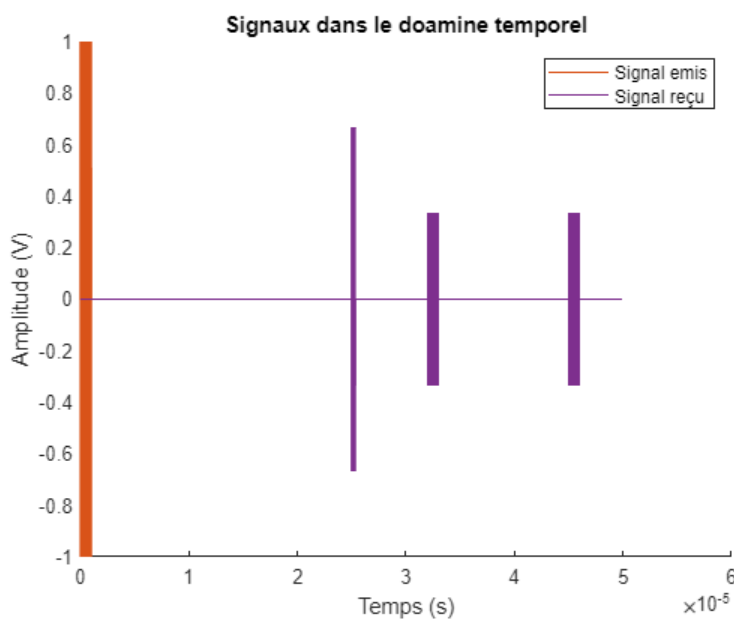
```
figure(11);
plot(Temps, Signal_emis, 'Color', '#D95319')
title('Signal émis dans le doamine temporel');
xlabel('Temps (s)');
ylabel('Amplitude (V)');
```



```
figure(12);
plot(Temps, Signal_recu,'Color','#7E2F8E')
title('Signal reçu dans le doamine temporel');
xlabel('Temps (s)');
ylabel('Amplitude (V)');
```



```
figure(13);
hold on
plot(Temps, Signal_emis,'Color','#D95319'); % Singal émis en temporelle
plot(Temps, Signal_recu,'Color','#7E2F8E'); % Singal reçu en temporelle
title('Signaux dans le doamine temporel');
legend('Signal emis', 'Signal reçu');
xlabel('Temps (s)');
ylabel('Amplitude (V)');
hold off
```



### 3 - Détermination des paramètres de la numérisation

```
% Nombre de point du vecteur Temps
NbPoint = length(Temps);

% Fréquence d'échantillonnage
Fe = 1/(Temps(NbPoint));

% Pas fréquentiel
PasFreq = Fe/NbPoint;

% Vecteur Fréquence Freq
Freq = PasFreq*((-(NbPoint-1)/2):((NbPoint-1)/2));

% Temps échantillonnage
Te = Temps(2) - Temps(1); % Temps entre deux échantillon
```

### 4 - Transformés de Fourier

#### Signal émis

```
% TF du signal émis
TF_Signal_emis = fft(Signal_emis)/NbPoint;
TF_Signal_emis = fftshift(TF_Signal_emis);

% Module et Phase de la TF du signal Radar Emis
Md_TF_Signal_emis = abs(TF_Signal_emis);
Ph_TF_Signal_emis = 360*angle(TF_Signal_emis)/(2*pi);
```

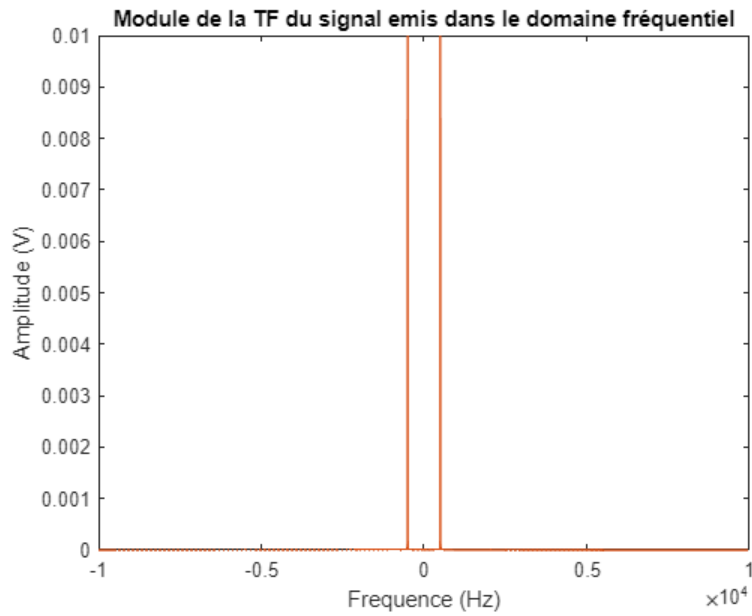
#### Signal reçu

```
% TF du signal reçu
TF_Signal_recu = fft(Signal_recu)/NbPoint;
TF_Signal_recu = fftshift(TF_Signal_recu);

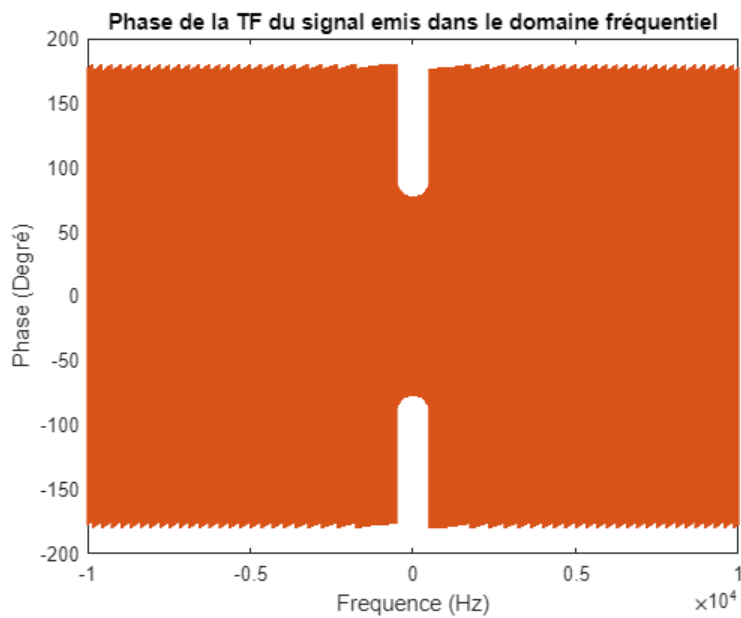
% Module et Phase de la TF du signal Signal_recu2
Md_TF_Signal_recu = abs(TF_Signal_recu);
Ph_TF_Signal_recu = 360*angle(TF_Signal_recu)/(2*pi);
```

#### Affichage des TF des signaux dans le domaine fréquentiel

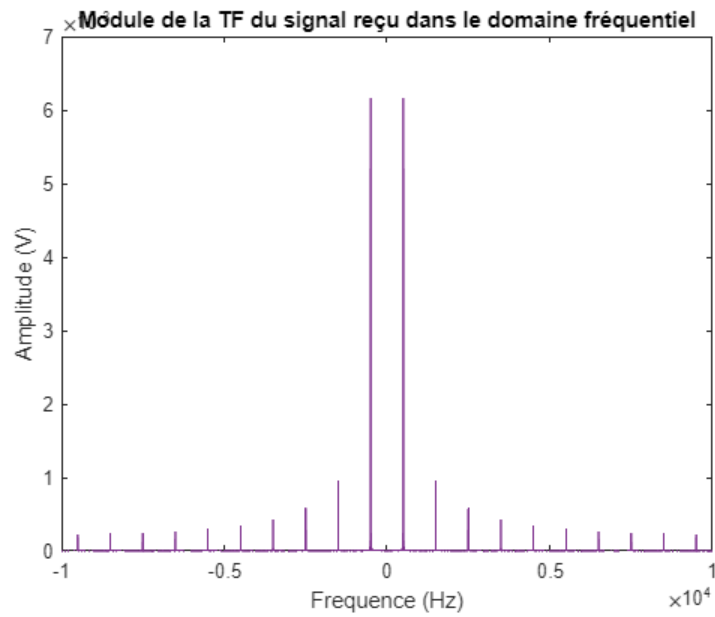
```
figure(21);
plot(Freq,Md_TF_Signal_emis,'Color','#D95319')
title('Module de la TF du signal émis dans le domaine fréquentiel');
xlabel('Frequence (Hz)');
ylabel('Amplitude (V)');
```



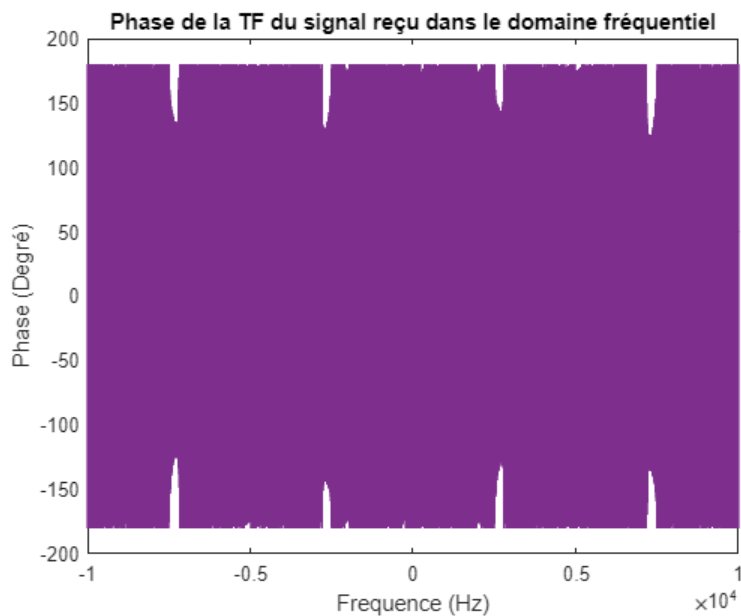
```
figure(22);
plot(Freq,Ph_TF_Signal_emis,'Color','#D95319')
title('Phase de la TF du signal émis dans le domaine fréquentiel');
xlabel('Frequence (Hz)');
ylabel('Phase (Degré)');
```



```
figure(23);
plot(Freq,Md_TF_Signal_recu,'Color','#7E2F8E')
title('Module de la TF du signal reçu dans le domaine fréquentiel');
xlabel('Frequence (Hz)');
ylabel('Amplitude (V)');
```



```
figure(24);
plot(Freq,Ph_TF_Signal_recu,'Color','#7E2F8E')
title('Phase de la TF du signal reçu dans le domaine fréquentiel');
xlabel('Frequence (Hz)');
ylabel('Phase (Degré)');
```



## 5 - Corrélation des signaux

### Autocorrélation du signal EMIS RADAR

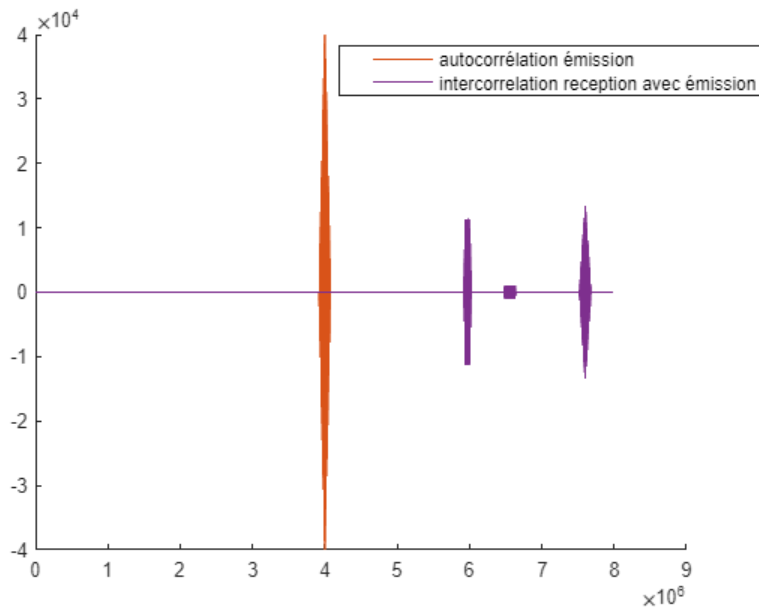
```
AutoCorr_Signal_emis = xcorr(Signal_emis);
```

### Interrelation du signal signal reçu et du signal émis

```
InterCorr_Signal_emis_Signal_recu = xcorr(Signal_recu,Signal_emis);
```

### Affichage des corrélations

```
figure(31);  
hold on  
plot(AutoCorr_Signal_emis,'Color','#D95319');  
plot(InterCorr_Signal_emis_Signal_recu,'Color','#7E2F8E');  
legend('autocorrélation émission', 'intercorrelation reception avec émission ');  
legend("Position",[0.49167,0.82817,0.53929,0.082143]);  
hold off
```



## 6 - Calcul final

```
[Decalage_temporel] =  
calculRetard_correlation(AutoCorr_Signal_emis,InterCorr_Signal_emis_Signal_recu,Te);  
  
vitesse_onda = 3e8;  
distance = (Decalage_temporel/2) * vitesse_onda;  
fprintf("Distance avion = %4.2f mètres", distance) %disp(distance)
```

Distance avion = 6750.00 mètres