Débruitage d'un signal par FFT

⟶ *Le but de ce TP est de mettre en œuvre une méthode de filtrage par transformée de Fourier, puis d'étudier son efficacité.*

1) **Transformée de Fourier et transformée inverse**

Générer un signal sinusoïdal de fréquence $f = 10$ Hz d'une durée de 1s.

Sur une figure à trois cadrans, représenter :

– le signal

– sa transformée de Fourier (*fft*)

– *la transformée de Fourier inverse (ifft) : on retrouve exactement le signal de départ.*

2) **Bruit blanc**

Générer un bruit blanc (signal aléatoire et de moyenne nulle, cf. *rand*) de la même taille que le signal sinusoïdal précédent.
Représenter sur une figure à 2 cadrans le bruit et son spectre.
Comparer l'amplitude du spectre du signal à l'amplitude du spectre du bruit blanc. Commenter.

3) **Bruitage du signal**

Ajouter au signal initial un bruit aléatoire d'amplitude 50% celle du signal de départ.
Représenter ce signal bruité.

4) **Filtrage par FFT**

Repérer l'amplitude maximale M de la transformée de Fourier. Définir un seuil S (10% par exemple).

Créer un filtre F selon les caractéristiques :

– F a la même taille que le signal (ou sa transformée de Fourier)

– lorsque l'amplitude de la FFT est plus petite que $S \times M$ le filtre vaut 0, et vaut 1 sinon.

Il ne reste plus alors qu'à appliquer le filtre à la FFT du signal (multiplication terme à terme du filtre par la FFT), et finalement à prendre la transformée de Fourier inverse.

Retrouve-t-on le signal de départ, sans le bruit additionnel ?

5) **Efficacité de la méthode**

a) Recommencer les simulations en variant les différents paramètres (amplitude du bruit, seuil du filtre).

b) Jusqu'à quel niveau de bruit le filtrage reste-t-il efficace ?

c) Recommencer les simulations avec des signaux initiaux plus complexes, par exemple avec le signal :

$$s(t) = \cos(2\pi f_1 t) + 3\cos(2\pi f_2 t) - 6\cos(2\pi f_3 t)$$

où, f_1, f_2 et f_3 sont trois fréquences différentes (par exemple, $f_1 = 10$ Hz, $f_2 = 55$ Hz et $f_3 = 122$ Hz).

6) **Spectre du signal**

Reprendre les calculs de FFT précédents en affichant sur une figure à trois cadrans :

– le signal dans le domaine temporel, avec l'échelle en temps correcte,

– le spectre du signal calculé par Matlab (*fft*), avec l'échelle fréquentielle correcte,

– le spectre réel du signal (*fftshift*), avec l'échelle fréquentielle correcte.

Retrouve-t-on les fréquences effectivement contenues dans le signal ? Avec quelle précision ?