Instrumentation

Joseph Moerschell, Marc Nicollerat

10 Analyse fréquencielle : DFT et FFT

- Dualité temps-fréquence
- Échantillonnage
- Transformée de Fourier discrète, FFT
- Fenêtrage

10.1 Rappel : Transformation de Fourier

La transformation de Fourier établit une relation entre un signal dans le temps et sa représentation spectrale :

$$X(\xi) = \int_{-\infty}^{\infty} x(t)e^{-j\xi t}dt$$

Plusieurs formes sont possibles, la forme suivante permet de mettre en évidence la relation avec la fréquence :

$$X(f) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} x(t)e^{-j2\pi f t}dt$$

Le facteur $\frac{1}{\sqrt{2\pi}}$ permet d'avoir une symétrie entre transformée directe et inverse. Il n'est pas toujours considéré.

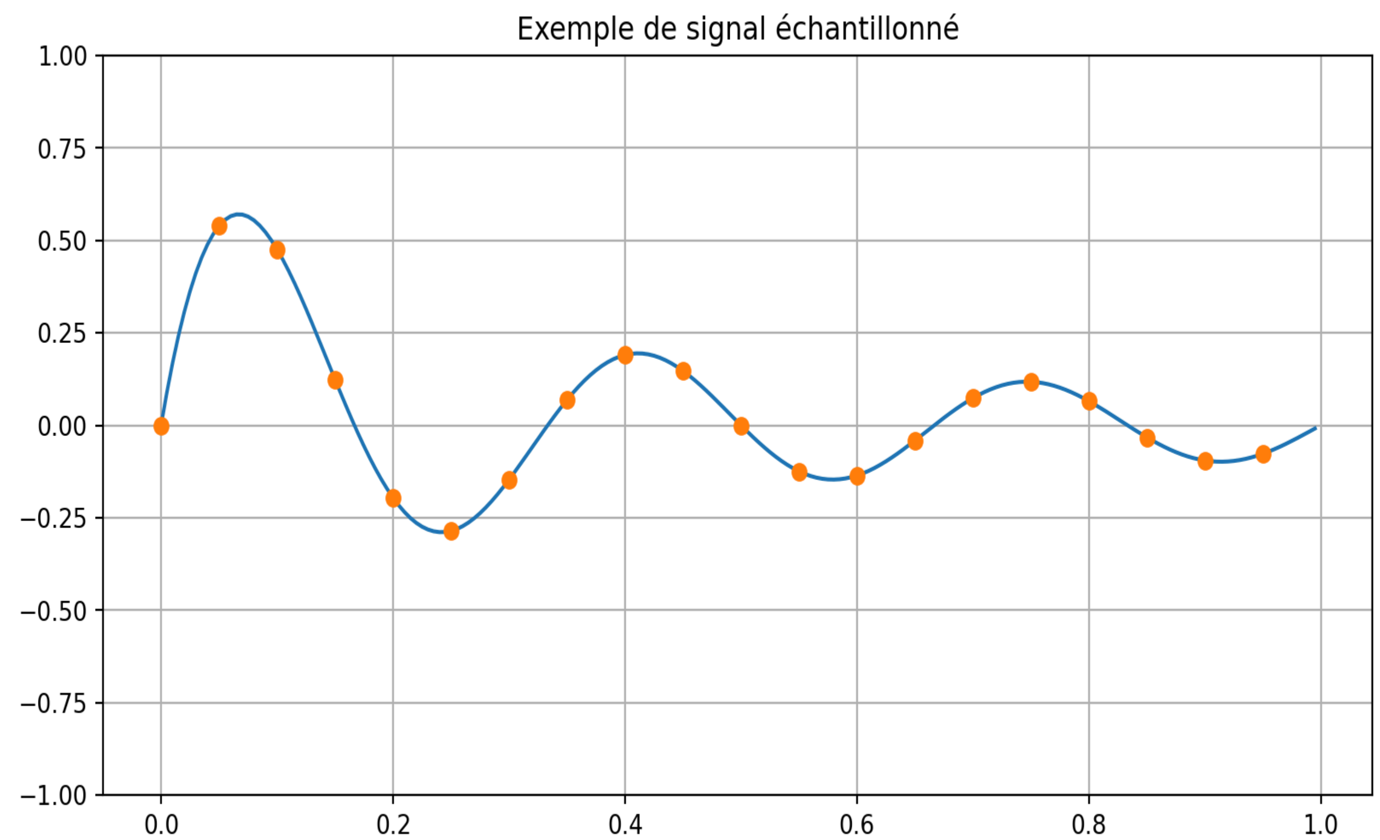
10.2 Échantillonnage d'un signal

On peut définir la version *discrète* de la transformée de Fourier qui agit sur un signal échantillonné. Un signal échantillonné à une fréquence d'échantillonnage T_e est définit comme suit :

$$x_e[n] = x(n \cdot Te)$$

soit la valeur $x(t)$ pour $t = n \cdot T_e$.

Le signal échantillonné est quelque chose de moins abstrait, il est limité dans le temps, avec un nombre fini (N) d'échantillons. De façon similaire à la version continue, on peut définir la **Transformée de Fourier Discrète** ou TFD




```
signal: [ 0.00000000e+00  5.39344663e-01  4.75528258e-01
1.23606798e-01
 -1.95928417e-01 -2.85714286e-01 -1.46946313e-01
6.86704432e-02
  1.90211303e-01  1.47093999e-01  6.12323400e-17
-1.24464153e-01
 -1.35865217e-01 -4.12022659e-02  7.34731565e-02
1.17647059e-01
  6.53094725e-02 -3.25281047e-02 -9.51056516e-02
-7.70492376e-02]
```

10.3 Transformation de Fourier discrète

En anglais **Discrete Fourier Transform** (DFT)

La définition de la transformée discrète est proche de sa version continue :

$$X[k] = \sum_{n=0}^{N-1} x[n] \cdot e^{-j\frac{2\pi k}{N}n}$$

L'opération est *inversible*, on a la relation suivante pour trouver x à partir de X:

$$x[n] = \sum_{k=0}^{N-1} X[k] \cdot e^{j\frac{2\pi n}{N}k}$$

La transformée est définie pour $0 < k \leq N$.

Tip

Ceci revient à multiplier tous les échantillons de $x[n]$ par les échantillons $e^{-j\frac{2\pi k}{N}n}$ pour une valeur de k. Pour avoir une valeur de X[k], il faut multiplier et additionner N éléments.

Note

On note par une *majuscule* la transformée, et par une *minuscule* le signal

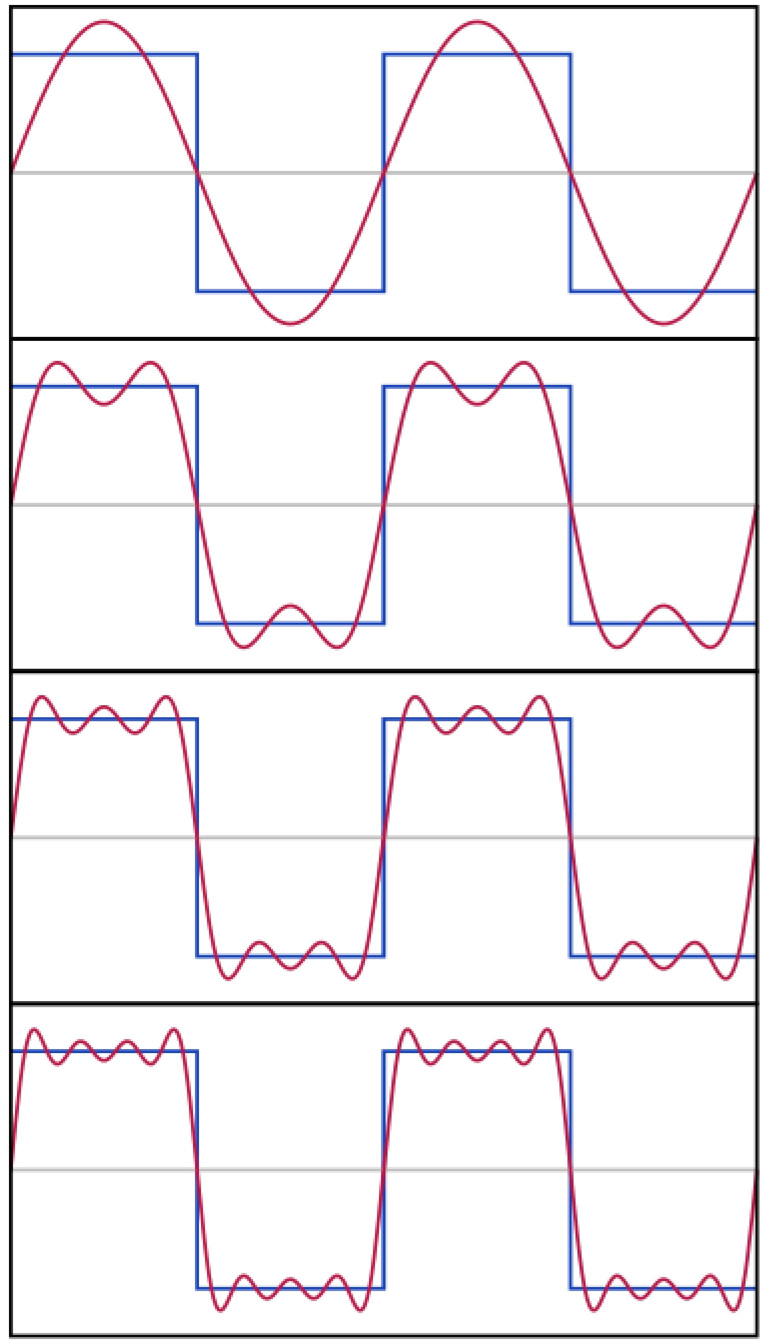
10.4 Illustration du principe de la décomposition

La Série de Fourier est la décomposition d'un signal périodique quelconque en une somme de composantes fondamentales (des sinus et des cosinus…).

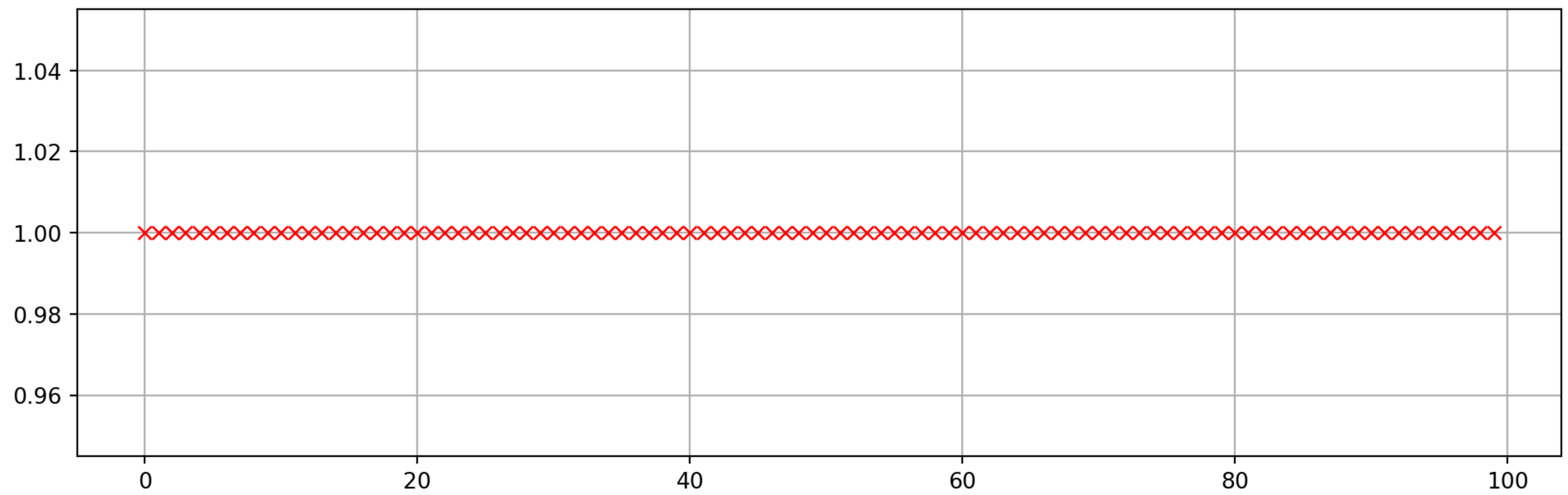
Expérimentation

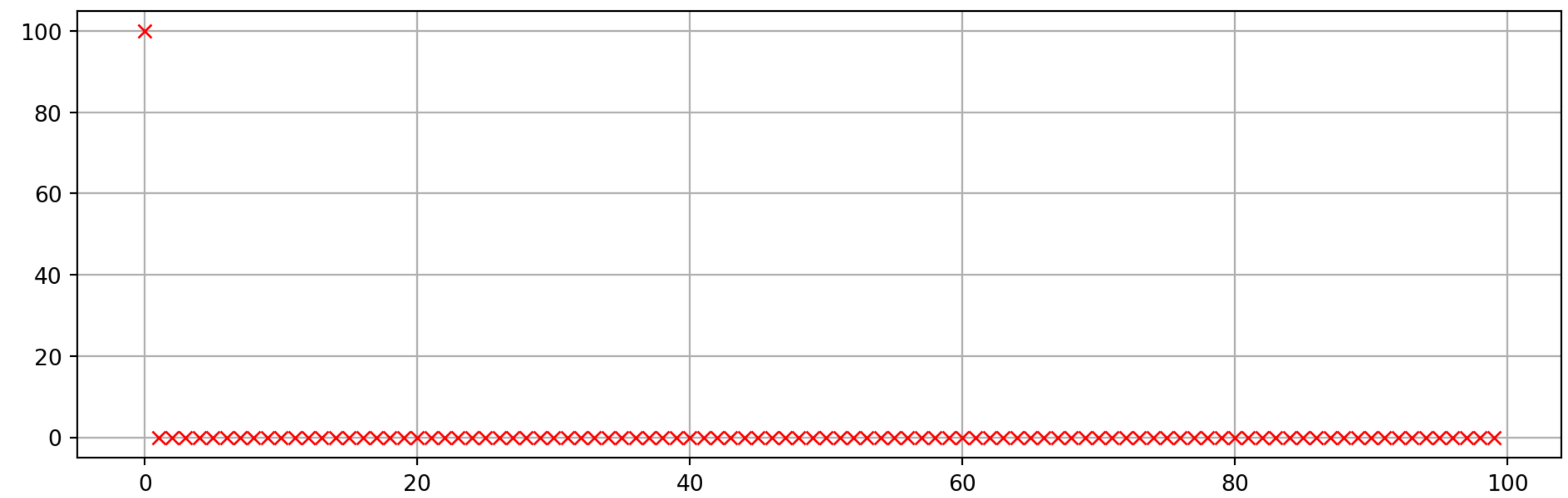
Le jupyter notebook **dev_10.1_inside_dft** permet de jouer avec le principe de la décomposition.

- Signal carré
- Convolution avec un sin/cos de même période
- Soustraction du résultat
- Itération avec les fréquences, jusqu'à 0



10.5 Exemples : DFT d'une constante

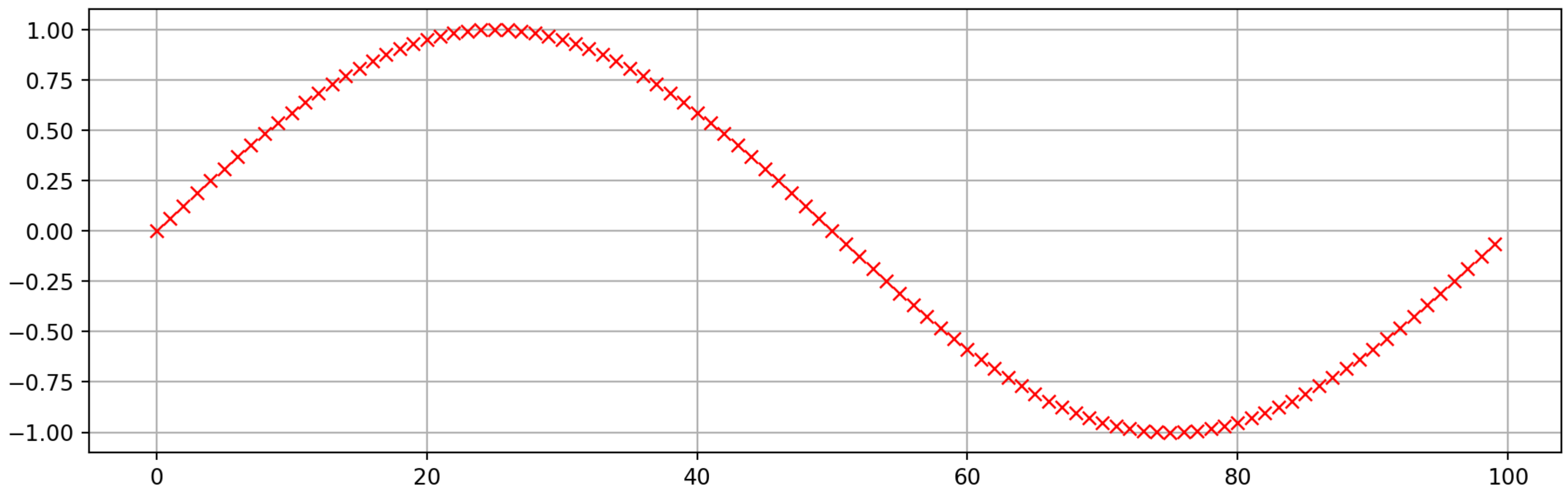


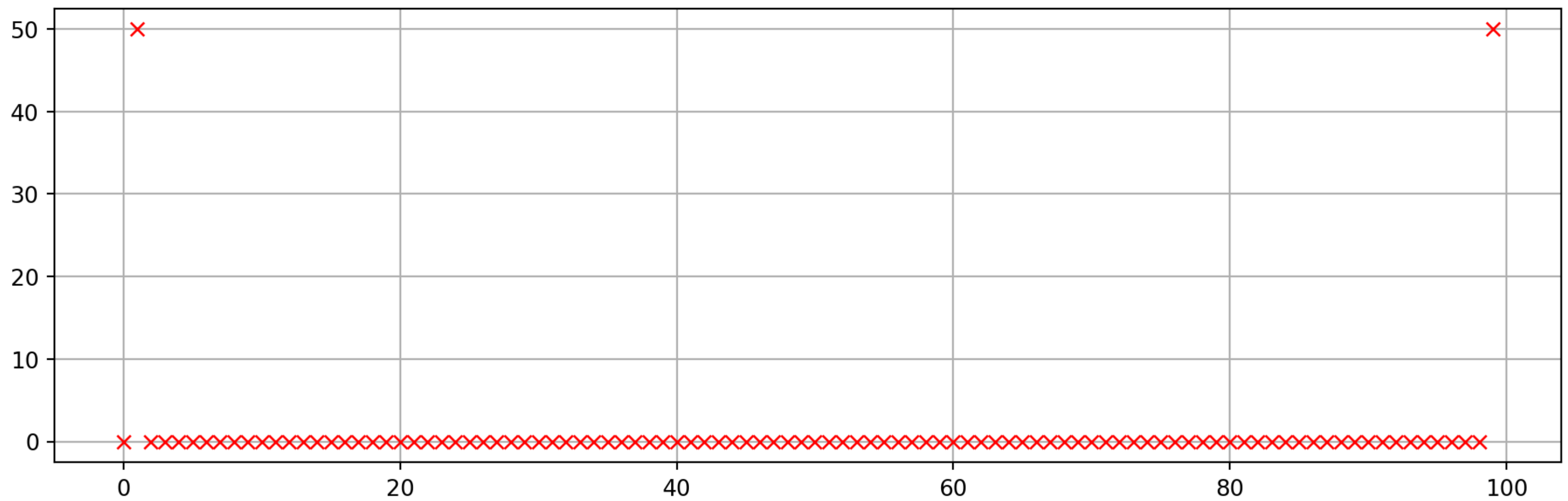


Seule la premième valeur est non nulle.

> **Important**
>
> On trace sur le graphique les **valeurs absolues** de la FFT. La FFT retourne des nombres *imaginaires*. Ils représentent une fréquence et une *phase*. L'amplitude est multipliée par N.

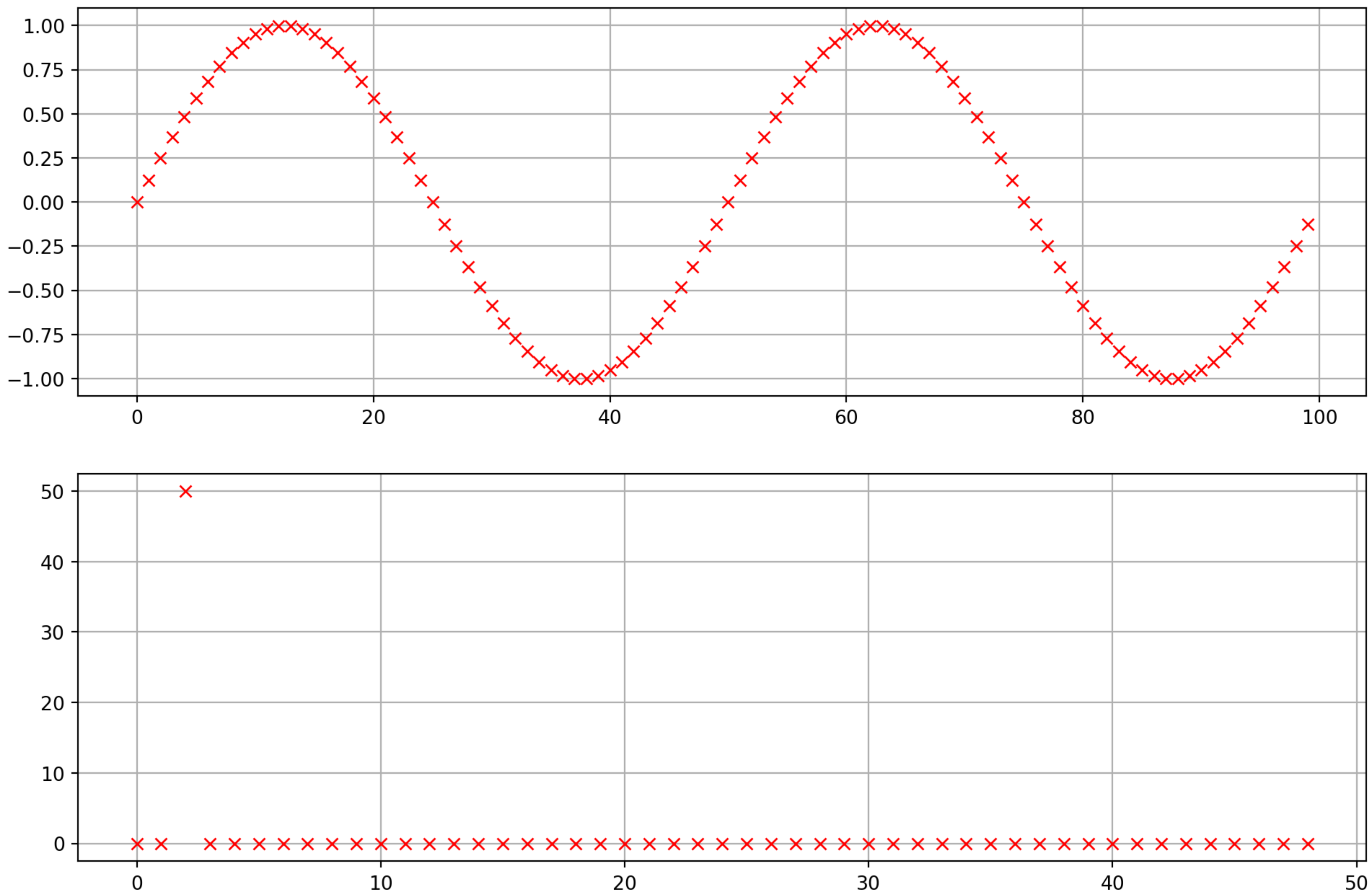
10.6 Exemples : DFT d'une fondamentale





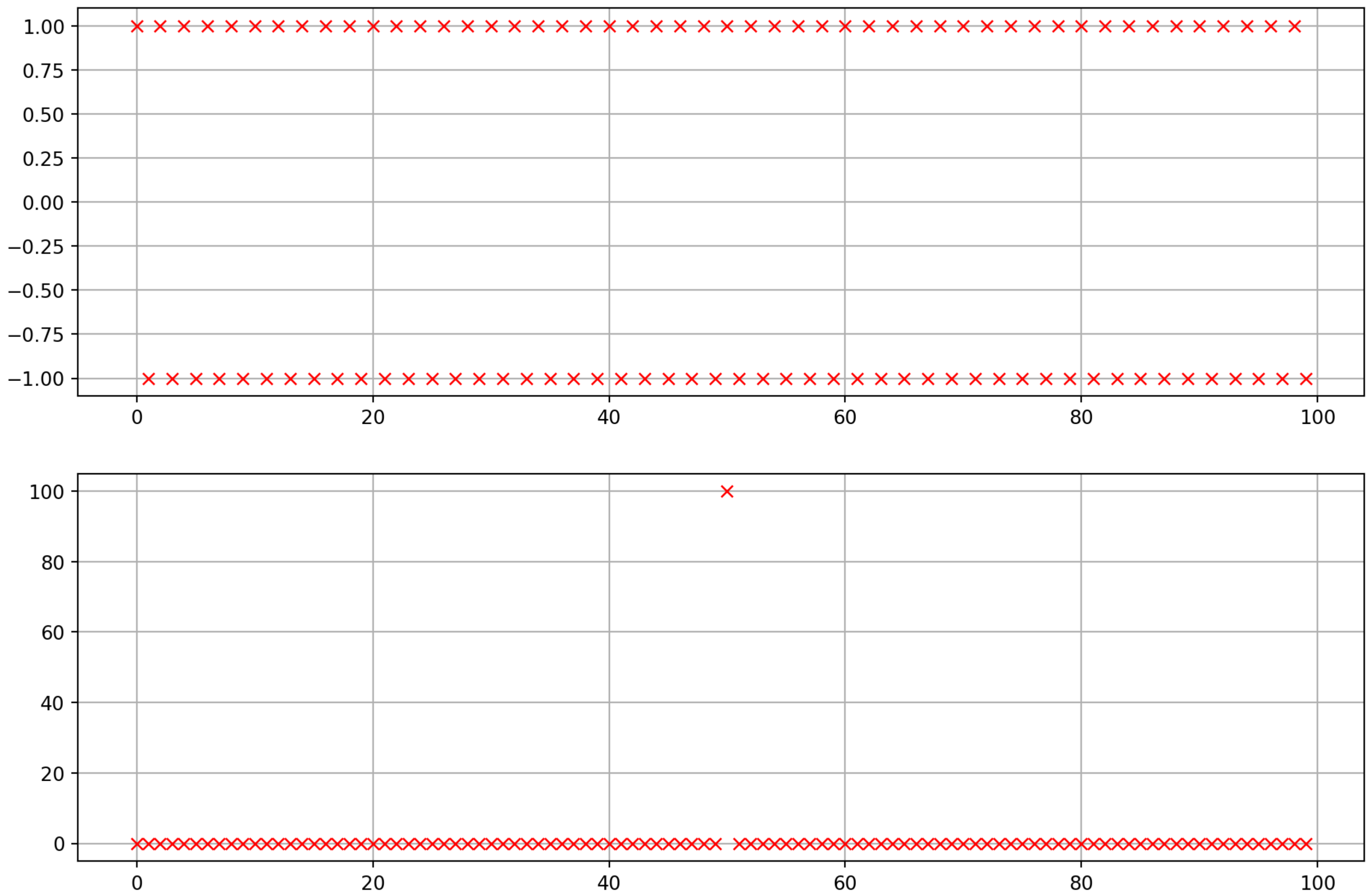
La seconde et la dernère valeurs sont non nulles. *La FFT retourne un résultat symétrique.*

10.7 Exemples : Double de fréquence



Chaque fois que la fréquence est multipliée, un point suivant la représente. La fréquence maximum est atteinte au milieu de la FFT.

10.8 Exemples : Fréquence maximum



La fréquence maximum est atteinte au milieu de la FFT.

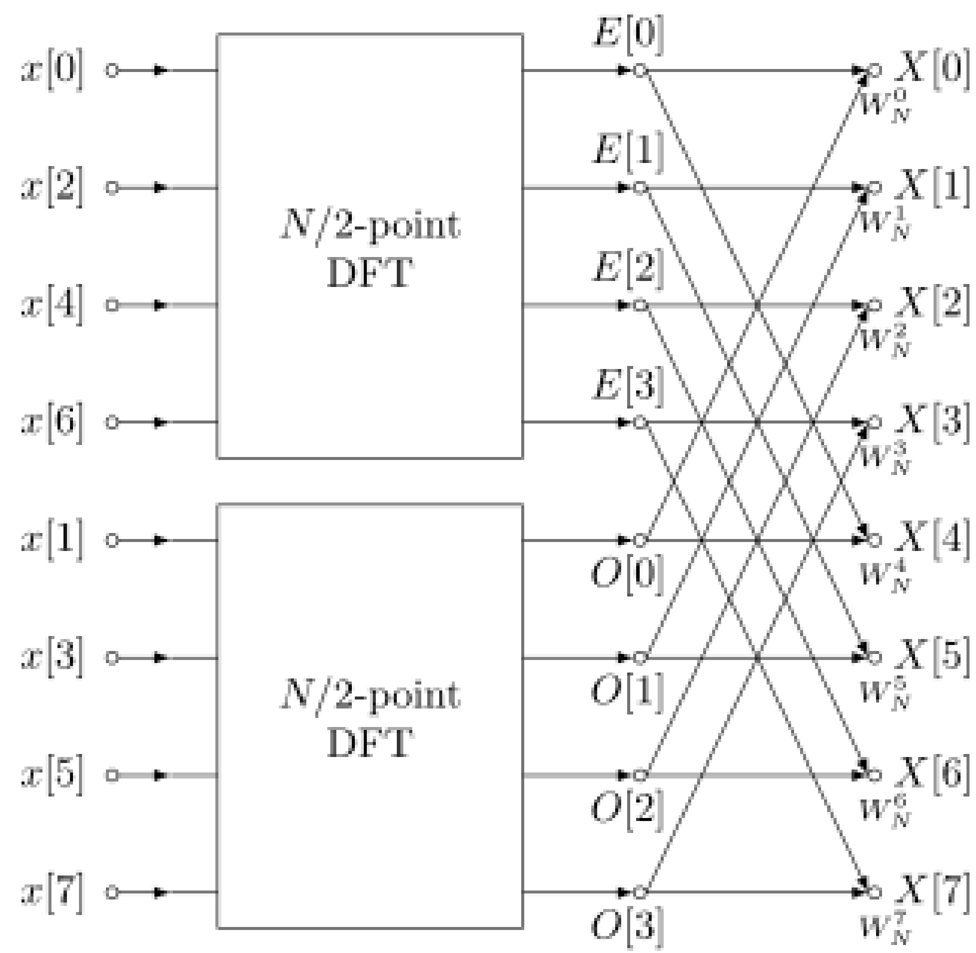
> **Important**
>
> On a représenté TOUTE la FFT. Il n'y a qu'un point et son amplitude est *double*.

10.9 DFT et FFT

Le calcul de la DFT demande passablement de calculs. Si on a déjà calculé les valeurs $e^{-j\frac{2\pi k}{N}n}$, il reste N*N multiplication-additions à faire. Il est possible d'optimiser le calcul.

FFT tient pour *Fast Fourier Transform*. C'est un algorithme qui permet d'accélérer le calcul.

- L'algorithme est défini pour des échantillons longs d'une puissances de 2 (N=512,1024,...)
- Des variations de l'algorithme s'accomodent de n'importe quelle longueur



La FFT réduit le nombre d'opérations à $N * \log_2(N)$.

10.10 Propriétés de la FFT

- Les valeurs retournées par la FFT sont des nombres *complexes*. On utilise en général la *norme* pour l'afficher.
- Toutes les valeurs sont multipliées par N
- La FFT est *symétrique*, les fréquences sont représentées de f_s/N à $f_s/2$, puis on retrouve les mêmes valeurs dans le sens inverse. Chaque paire de valeur représente l'amplitude, il apparaît un facteur 2.
- Le point 0 donne la moyenne
- Le point milieu donne la fréquence maximum ($f_e/2$).
- La résolution en fréquence est liée à la fréquence d'échantillonnage f_e et au nombre de points. Elle vaut $\Delta f = f_e/N$.

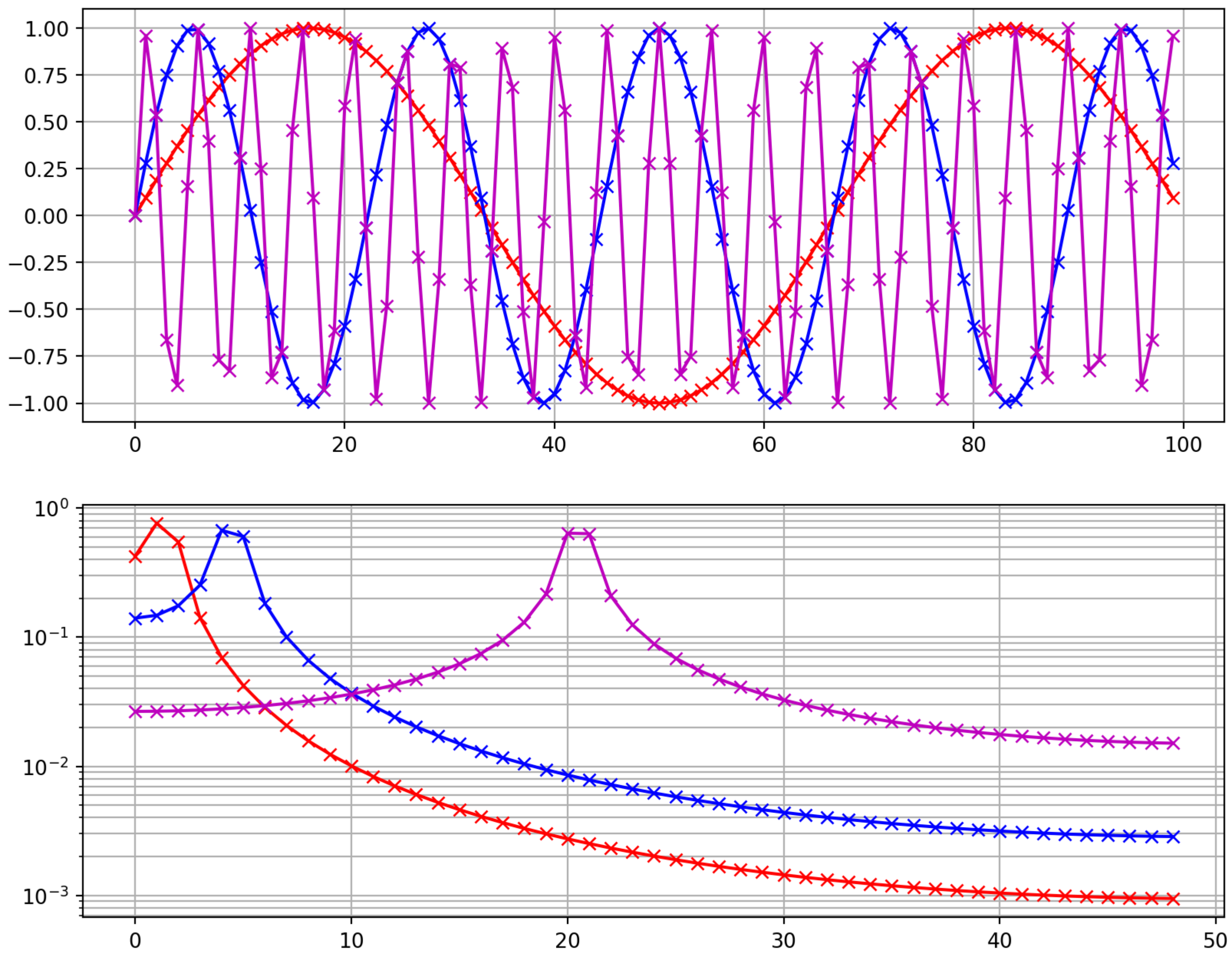
Note

On peut représenter la FFT de différentes façons :

- Telle quelle
- N'afficher que la moitié des points
- De façon symétrique en posant la moitié de droite à gauche.

10.11 Limitations de la FFT

- On peut exprimer n'importe quel signal périodique avec une FFT
- Tout fonctionne bien tant que le signal a un nombre de périodes entier sur la *fenêtre*



Attention

Si la période n'est pas entière sur les échantillons, le résultat est moins bien défini.

10.12 L'effet des fenêtres

Intuitivement, on voit que si la période ne correspond pas, lors de la *convolution*, on perd la symétrie entre le signal et le terme exponentiel.

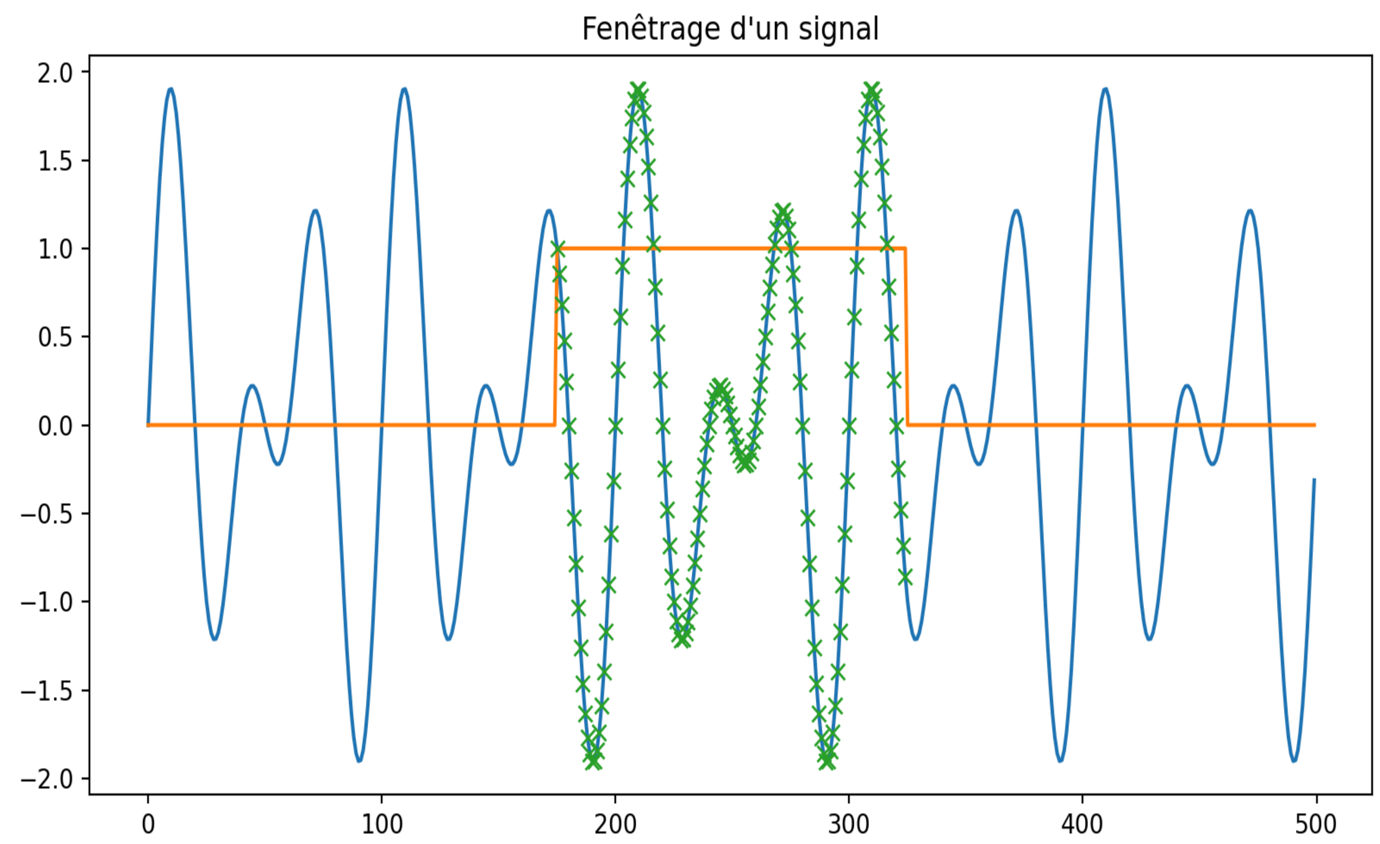
$$X[1] = \sum_{n=0}^{N-1} x[n] \cdot e^{-j2\pi 1 \frac{n}{N}}$$

Le fait de *tronquer* un signal sensé être infini revient à le multiplier par une fenêtre $w(t)$:

$$x_w(t) = x(t) \cdot w(t)$$

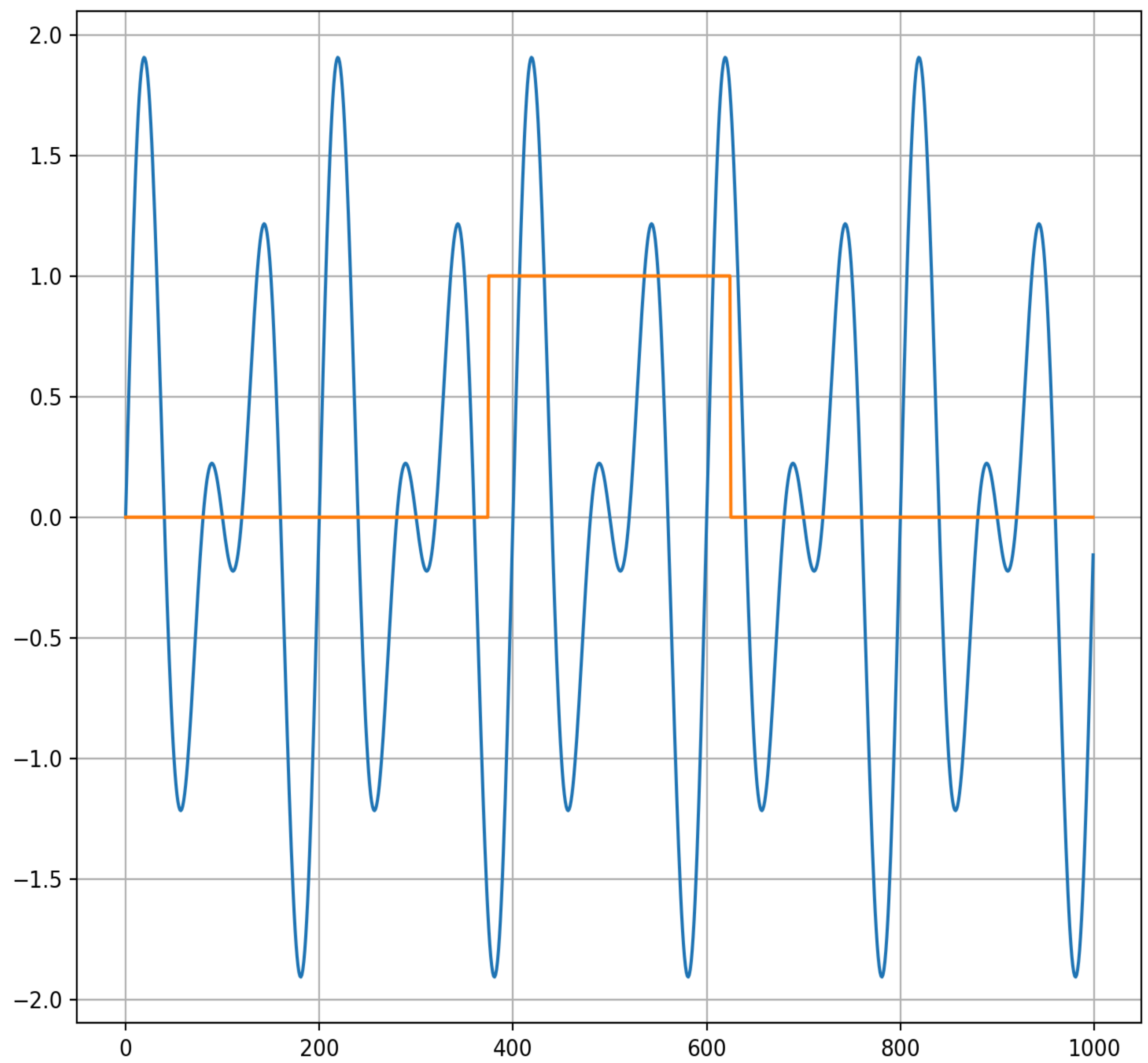
Le fait de multiplier 2 signaux dans le temps revient à les *convoluer* en fréquence.

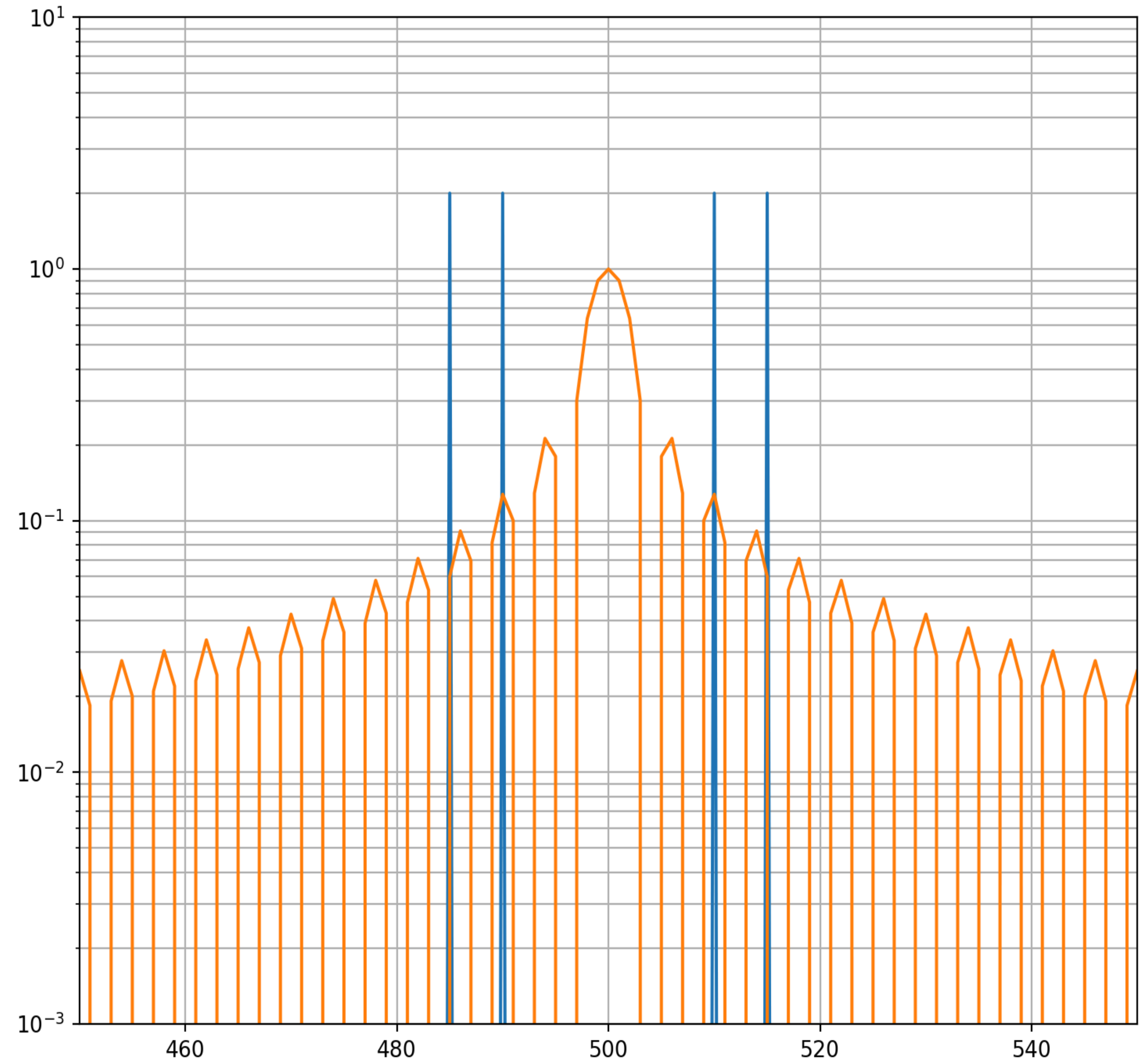
$$X_w(f) = X(f) * W(f)$$



10.13 Effet des fenêtres, la suite

La fenêtre dans le temps agit comme un filtre en fréquences.





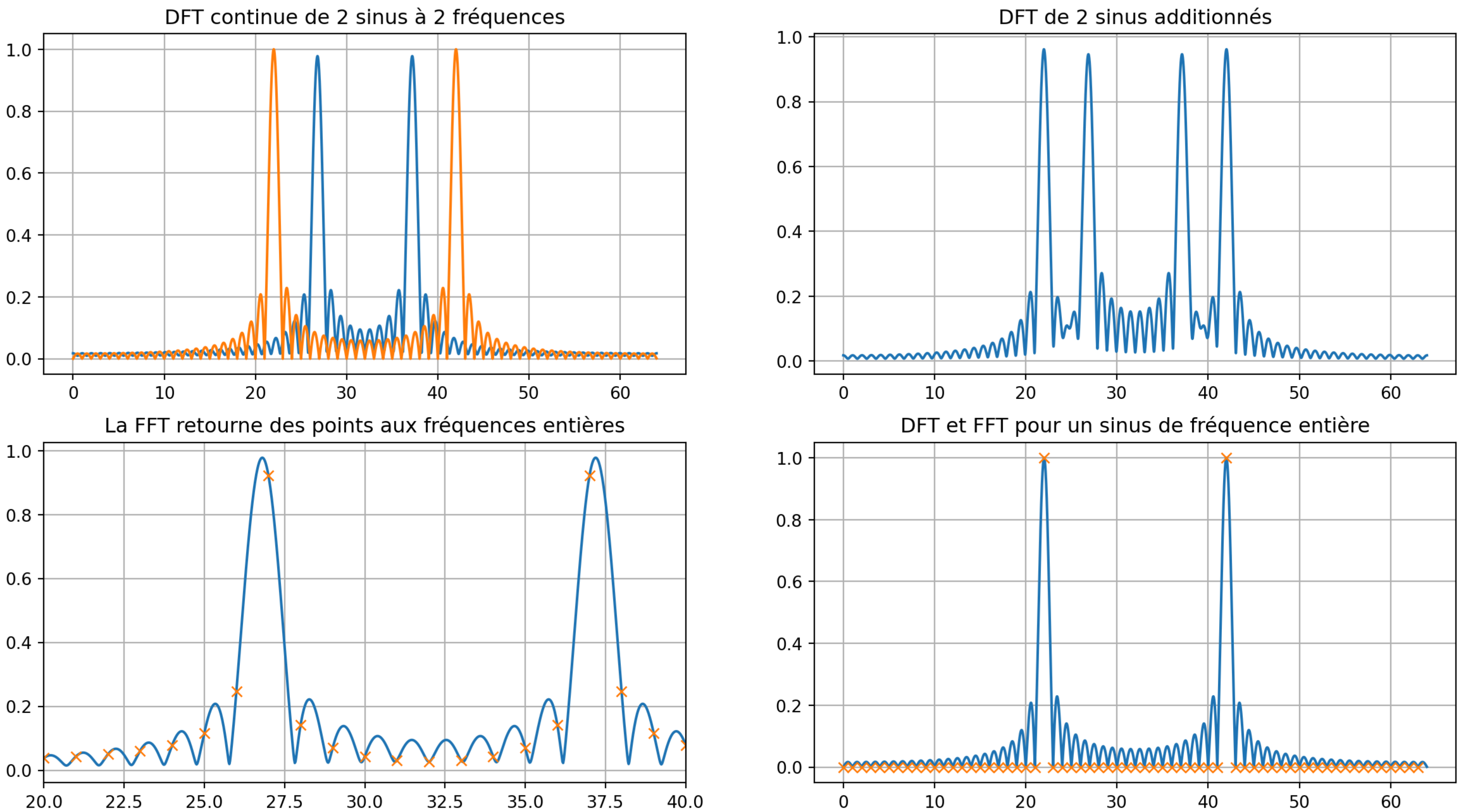
Dans la figure, le signal périodique en bleu se transforme en 4 raies, la fenêtre rouge en quelque chose de plus compliqué…

La *convolution*, dans le cas des *raies*, revient à superposer 4x le signal en orange sur les raies en bleu.

10.14 Resultat de la *convolution*

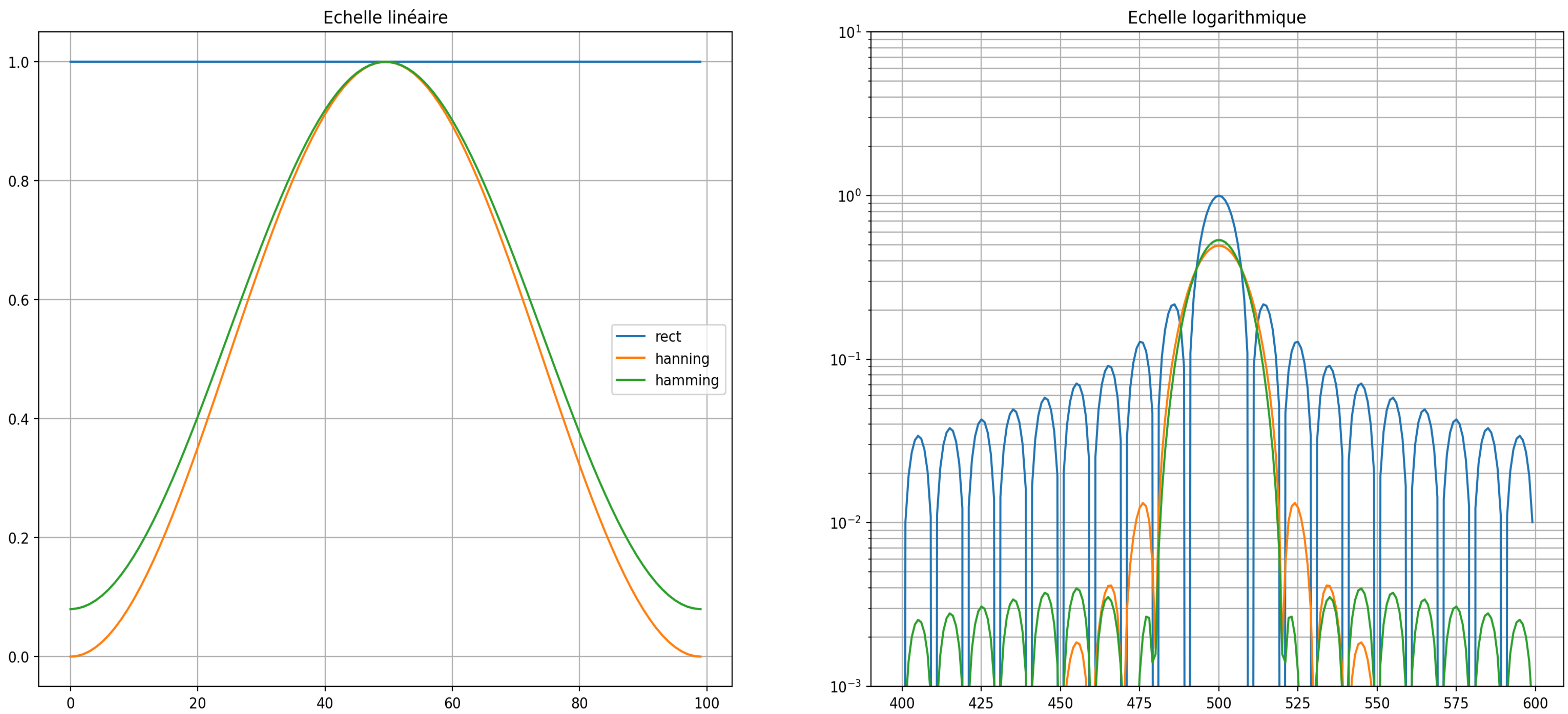
Un signal $x(t) = sin(2\pi f_0 t) \cdot rect(t)$ a une DFT de la forme

$$X(f) = \frac{\delta(f - f_0) + \delta(f + f_0)}{2} * sinc(f)$$



10.15 Type de fenêtres

Il y a plusieurs types de fenêtres, idéalement, $W(f)$ devrait ressembler à un $\delta(f)$ pour que X_w représente exactement $X(f)$.

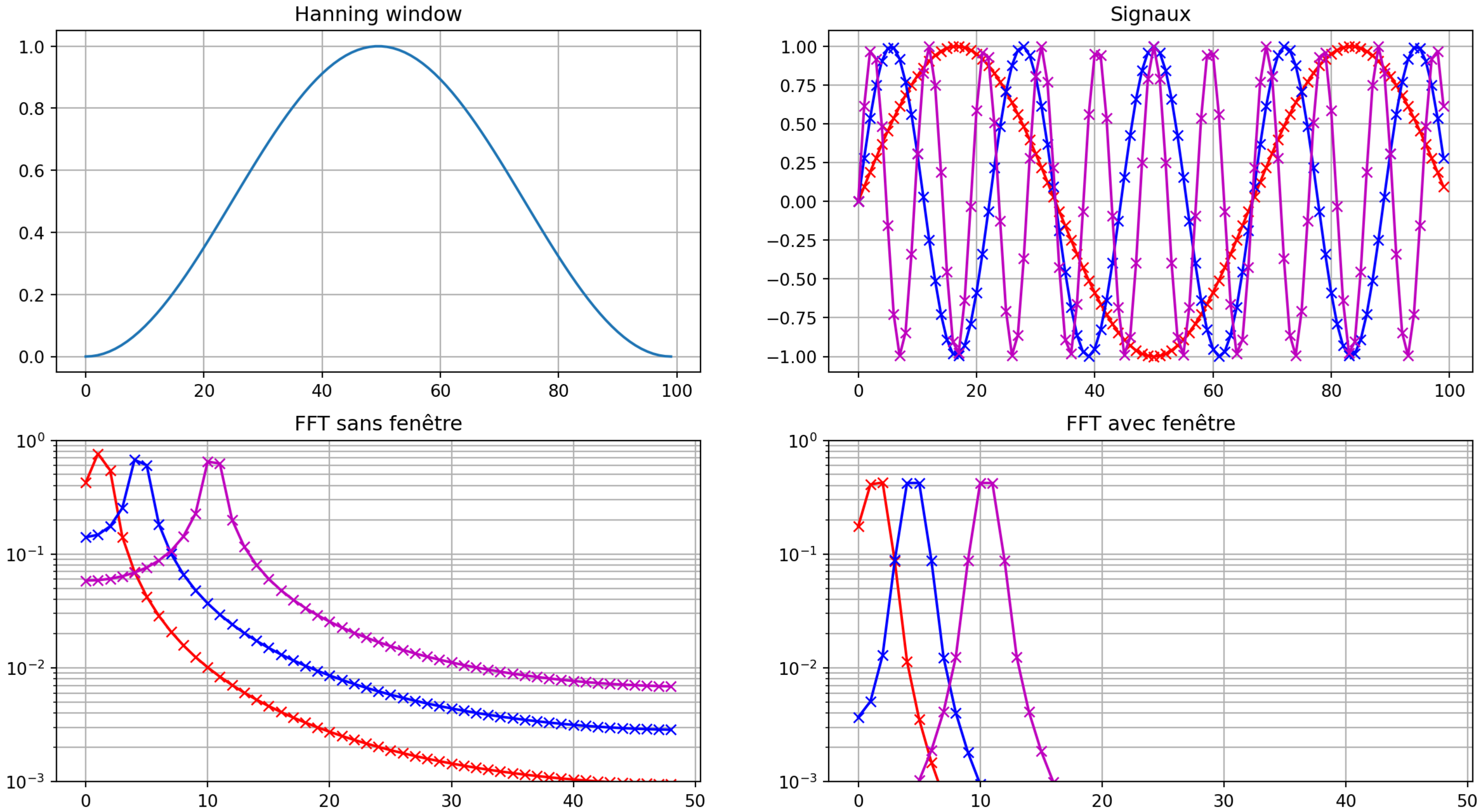


La fenêtre la plus pointue correspond au rectangle, mais les lobes vont sélectionner les fréquences voisines.

La fenêtre de hamming a des lobes secondaires très faibles mais elle est plus large au centre.

10.16 Utilisation d’une fenêtre

L’utilisation d’un *fenêtre bien choisie* permet de réduire la largeur des raies.



10.17 Différentes fenêtres et leur FFT

Pratiquez !

exercice

Essayer de calculer la FFT de différents signaux :

- Signal composé de plusieurs fréquences.
- Signal aléatoire.

Une base est proposée dans ex_10.1_TestFFT.ipynb

Exercice

Utilisez le jupyter notebook ex_10.2_CalcFFTwindow.ipynb pour tester différentes fenêtres.

Pratique

Suivez les suggestions du notebook dev_10.4_windowing_effect.ipynb

10.18 Les fenêtres en bref..

La fenêtre rectangulaire est tacitement présente lors de l’échantillonnage. Si le signal est périodique, c’est le mieux.

Si le signal n’est pas périodique, le choix de la fenêtre permet de mieux ajuster la mesure, selon les besoins :

- Détecter l’amplitude des signaux avec précision (par exemple flattop)
- Discerner un signal de faible amplitude (par exemple Hanning)

La fenêtre introduit un gain équivalent à

$$g_w = \frac{\sum_{n=0}^{N-1} w[n]}{N}$$