ELE2700: Analyse des signaux
Département de génie électrique
Polytechnique Montréal

Introduction à la transformée de Fourier rapide (FFT)

Dernière mise à jour : 27 août 2024.
Note : Ce document introduit les bases de la transformée de Fourier rapide utilisée pendant les travaux pratiques. Son traitement complet sera fait au Chapitre 6.

---

1 Introduction

La transformée de Fourier rapide, connue en anglais sous le nom *Fast Fourier Transform* ou FFT, est un algorithme essentiel à l'analyse des signaux. La première version de cet algorithme a été publiée en 1965, par les mathématiciens J. W. Cooley et J. W. Tukey [1]. Cet algorithme a été développé dans le but de réduire les ressources informatiques nécessaires pour le calcul de la transformée discrète de Fourier, dont la complexité est $\mathcal{O}(n^2)$, où n représente le nombre d'échantillons du signal.

Ce document servira comme introduction à la méthode numérique, détaillant le fonctionnement de l'algorithme et l'impact des différents paramètres utilisés. Le traitement détaillé de la FFT sera fait au Chapitre 6 du cours magistral.

2 Bases de la transformée de Fourier

Le transformée de Fourier est une transformation permettant de convertir un signal d'une représentation temporelle à une représentation fréquentielle. Cette transformée est décrite par l'équation suivante :

$$X(f) = \mathcal{F}\{x(t)\} = \int_{-\infty}^{+\infty} x(t)\mathrm{e}^{-j2\pi ft}\mathrm{d}t.$$

Cette représentation est excellente pour une approche analytique, où l'on considère des signaux théoriques, continus et connus sur l'entièreté du domaine temporel.

2.1 Transformée discrète de Fourier

Cependant, les signaux observés en pratique sont toujours représentés avec une certaine résolution, soit le nombre d'échantillons qui composent ce signal. Pour cela, une version discrète de la transformée de Fourier a été développée, la transformée discrète de Fourier (TDF). Celle-ci peut être écrite sous la forme suivante :

$$X[k] = \mathcal{F}_{\mathrm{D}}\{x[n]\} = \sum_{n=0}^{N-1} x[n]\mathrm{e}^{-j2\pi\frac{k}{N}n}.$$

Dans cette équation, N représente le nombre d'échantillons du signal et k, variant de 0 à $N-1$, représente l'intervalle de fréquence pour lequel nous calculons l'amplitude. Nous observons alors que pour un signal de taille N, il faut effectuer le calcul $x[n]\mathrm{e}^{-j2\pi\frac{k}{N}n}$ un total de N fois pour chaque intervalle. Puisque le nombre d'intervalles est égal au nombre d'échantillons, il est nécessaire de faire N^2 multiplications, produisant alors un algorithme dont la complexité est $\mathcal{O}(n^2)$.

3 Transformée de Fourier rapide (FFT)

À partir de la transformée discrète de Fourier, une approche efficace peut être développé, la FFT.

3.1 Bases de l'algorithme

Malgré le bénéfice qu'apporte la transformée à temps discret de Fourier, cette approche était beaucoup trop exigeante en termes de ressources informatiques pour être utilisée dans des applications réelles. Ce désavantage était la motivation principale pour le développement d'un algorithme capable d'obtenir les mêmes résultats de façon plus efficace. Aujourd'hui, cet algorithme est connu comme la transformée rapide de Fourier (*fast Fourier transform*, FFT).

Pour illustrer le fonctionnement de cet algorithme, reprenons l'équation qui représente la TDF. Considérons une situation simple, soit un signal avec $N = 8$ échantillons.

$$X[k] = \sum_{n=0}^{7} x[n]\mathrm{e}^{-j2\pi\frac{k}{8}n}.$$

Nous pouvons voir que chaque intervalle nécessite 8 multiplications, soit un total de 64 multiplications pour 8 bacs. Nous remarquons cependant que chaque multiplication comprend une exponentielle complexe, une fonction qui est périodique.

Observons alors les multiplications qui considèrent l'échantillon $n = 4$. Il faut multiplier la valeur $x[4]$ par l'exponentielle $\mathrm{e}^{-j2\pi\frac{k}{8}4}$, soit $\mathrm{e}^{-j\pi k}$. On remarque alors que pour les valeurs de k impaires, soit $k \in \{1, 3, 5, 7\}$, l'exponentielle complexe a la même valeur, soit -1. Le même peut être dit pour les valeurs paires, soit $k \in \{0, 2, 4, 6\}$ où la valeur de l'exponentielle est égale à 1. Nous pouvons alors éviter de répéter cette multiplication en réutilisant les valeurs calculées précédemment. Pour l'échantillon 4, cette technique réduit le nombre de multiplications à effectuer de 8 à 2.

En adoptant une approche similaire pour le reste des échantillons, il est possible de réduire le nombre de multiplications à effectuer de 64 à 24. Cette nouvelle technique produit alors un algorithme avec complexité $\mathcal{O}(n \log_2 n)$. Cela peut sembler comme une faible amélioration pour un signal de 8 échantillons. Cependant, si l'on considère un signal avec un million d'échantillons, un nombre qui est assez réaliste en considérant les systèmes modernes, cette nouvelle complexité fait la différence entre $(10^6)^2 = 10^{12}$ multiplications et $10^6 \log_2(10^6) \approx 2 \times 10^7$ multiplications ! À la vitesse des ordinateurs des années 1960, tel que le IBM 7030 "Stretch", nous comparons un temps de calcul de 9 jours et 16 heures à environ 17 secondes. Dans un contexte moderne, ceci peut se traduire par la possibilité d'analyser des signaux via la FFT sur des dispositifs de plus en plus petits.

3.2 Utilisation de la FFT

Maintenant que le fonctionnement de l'algorithme est établi, il est temps d'étudier l'impact des différents paramètres sur le spectre produit par la FFT.

Le premier paramètre à considérer est le nombre d'échantillons présents dans le signal analysé. Il est important de noter que la FFT ne fonctionne qu'avec des signaux dont la taille est un exposant de 2. Cela ne veut pas dire que nous ne pouvons pas utiliser la FFT pour trouver le spectre d'un signal avec une taille différente, il suffit d'ajouter des zéros à la fin du signal jusqu'à ce que la taille soit adéquate. Ce paramètre sera noté n_{FFT} dans ce document.

Le deuxième paramètre à considérer est la fréquence d'échantillonnage. Lorsqu'une séquence de données est présentée à l'algorithme, il n'y a aucune information dans la séquence qui détermine l'espacement entre chaque échantillon. Cette information est cependant disponible à l'utilisateur, sous forme de fréquence d'échantillonnage ou de période d'échantillonnage. Ce paramètre sera noté f_{s} dans ce document.

3.2.1 Résolution du spectre

Considérons un signal avec $n_{\mathrm{FFT}} = 50000$ points, échantillonné à un fréquence $f_{\mathrm{s}} = 10000$ Hz et tel que présenté à la Figure 1.

Comme mentionné précédemment, la FFT nécessite un nombre d'échantillons égal à un exposant de 2. Nous ajoutons alors des zéros à la fin du signal pour obtenir un total de $2^{16} = 65536$ points, le premier exposant de 2 supérieur à 50000. D'après le théorème de Nyquist-Shannon, la plus grande fréquence que nous pouvons détecter est égale à $f_{\mathrm{s}}/2$, ou dans notre cas 5000 Hz. Le spectre de fréquence comprend aussi les valeurs négatives, la bande de fréquence variera alors entre -5000 Hz et 5000 Hz, soit une bande de 10000 Hz. Puisque la FFT retourne un nombre

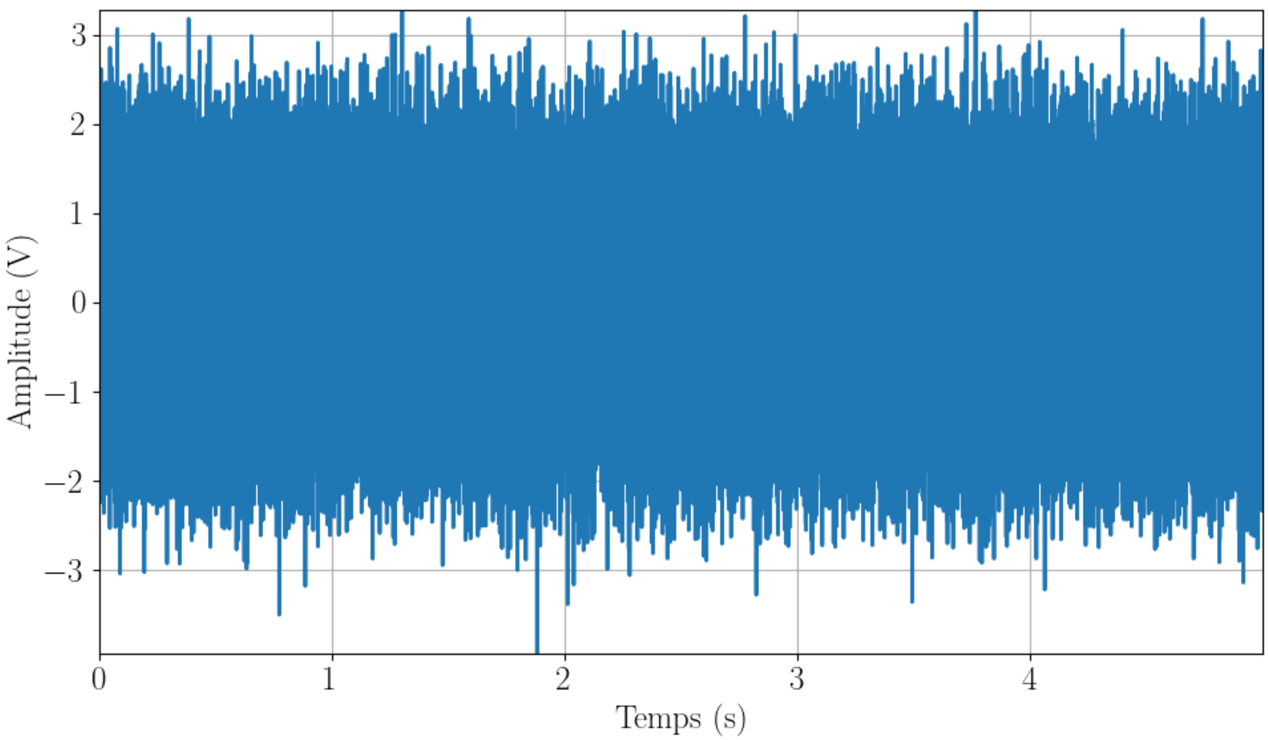


FIGURE 1 – Signal $s(t)$

d'intervalles égal au nombre d'échantillons donnés, nous pouvons déduire que 65536 intervalles de fréquences seront présents en sortie. En considérant que les intervalles sont tous de taille égale, nous pouvons déduire que chaque intervalle regroupe $\frac{10000}{65536} = 0.15259$ Hz. Plus concrètement la taille de chaque intervalle peut être exprimée par l'équation $\frac{f_s}{n_{\text{FFT}}}$. Le spectre du signal considéré est présenté à la Figure 2.

La résolution de ce spectre est très haute. Cela provient du grand nombre d'échantillons qui composent le signal analysé. Considérons maintenant un signal considérablement plus court. Supposons que nous étudions un nouveau signal d'une durée de 1.0 ms, encore échantillonné à $f_s = 10000$ Hz, dont les propriétés sont inconnues. Le signal est présenté à la Figure 3

Nous obtenons alors un signal avec 10 échantillons, ou 16 échantillons une fois que les zéros sont ajoutés. Effectuant le même calcul, nous trouvons une résolution de $10000/16 = 625$ Hz. Le spectre du signal d'une durée de 1 ms est donné à la Figure 4.

Cette résolution est comparativement beaucoup plus faible, mais puisque le nombre d'échantillons est inférieur, le calcul sera effectué beaucoup plus rapidement.

La résolution du spectre recommandée dépendra des applications. Certains contextes nécessiteront une faible résolution alors que d'autres nécessiteront une haute résolution. Il est alors important de bien connaître la résolution voulue pour éviter de surdimensionner nos systèmes.

Reprenons l'exemple à 10 échantillons, et supposant que nous avons besoin d'une résolution de 100 Hz au maximum. Pour obtenir cela sans devoir échantillonner des périodes supplémentaires, il suffit d'augmenter le nombre de zéros ajoutés à la fin du signal. En d'autres mots, il n'est pas nécessaire de se limiter à une taille de 16 échantillons, des tailles de 32, 64, 128, et plus d'échantillons serait aussi adéquats. En reprenant le calcul, une taille de 32 échantillons donnerait une résolution de $10000/32 = 312.5$ Hz, une taille de 64 échantillons donnerait une résolution de $10000/64 = 156.25$ Hz, et une taille de 128 échantillons donnerait une résolution de $10000/128 = 156.25$ Hz. Nous remarquons alors qu'il faut étendre les données à une taille de 128 échantillons pour répondre aux exigences. Le spectre pour les différentes tailles d'échantillon n_{FFT} est présenté à la Figure 5

Cette technique d'ajout de zéros est connue en anglais sous le nom *zero padding*, et est faite automatiquement par les logiciels permettant le calcul d'une FFT (Python, MATLAB, etc.). Le concept est vu en plus de détails dans le cadre du cours au Chapitre 6.

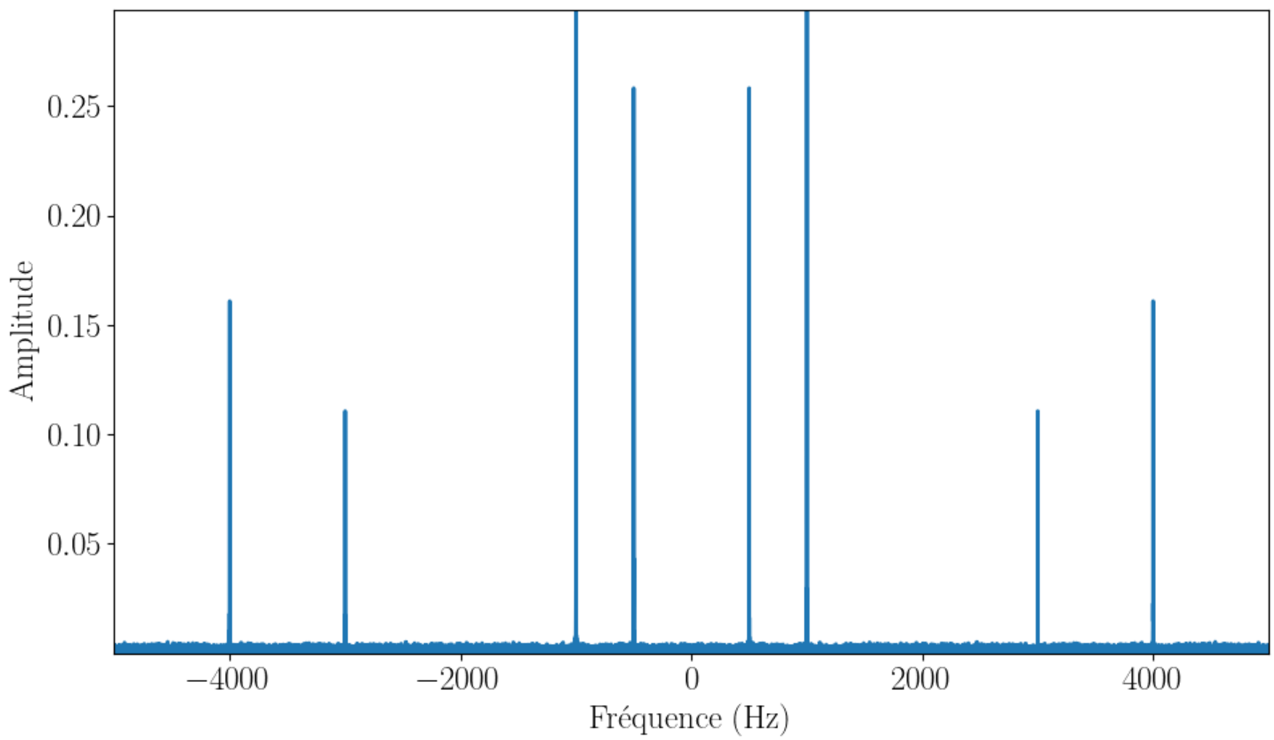


FIGURE 2 – Spectre $S[n] = \mathrm{FFT}(s(t))$

3.2.2 Normalisation du spectre

Supposons maintenant que nous voulons comparer les spectres de deux signaux différents. Les deux signaux sont échantillonnés à la même fréquence, indiquant que la bande de fréquence observée est la même pour les deux spectres. Cependant, le Signal 2 est composé de plus d'échantillons que le Signal 1. Puisque la FFT est essentiellement une somme, les valeurs obtenues à la sortie auront tendance à augmenter avec le nombre d'échantillons du signal. Il serait alors difficile de comparer ces signaux si les amplitudes sont influencées par la taille de chaque signal. Pour remédier à ce problème, il suffit de normaliser chaque spectre en divisant les amplitudes par la taille du signal.

4 Conclusion

La transformée de Fourier rapide est un algorithme essentiel au fonctionnement des technologies modernes. Dans ce document nous avons étudié les avantages de la FFT et son fonctionnement. Nous avons illustré l'impact des paramètres, soit la fréquence d'échantillonnage et le nombre d'échantillons, sur le spectre produit par l'algorithme.

Références

[1] M. Heideman, D. Johnson, and C. Burrus, "Gauss and the history of the fast fourier transform," *Archive for History of Exact Sciences*, vol. 34, pp. 265–277, 01 1985.

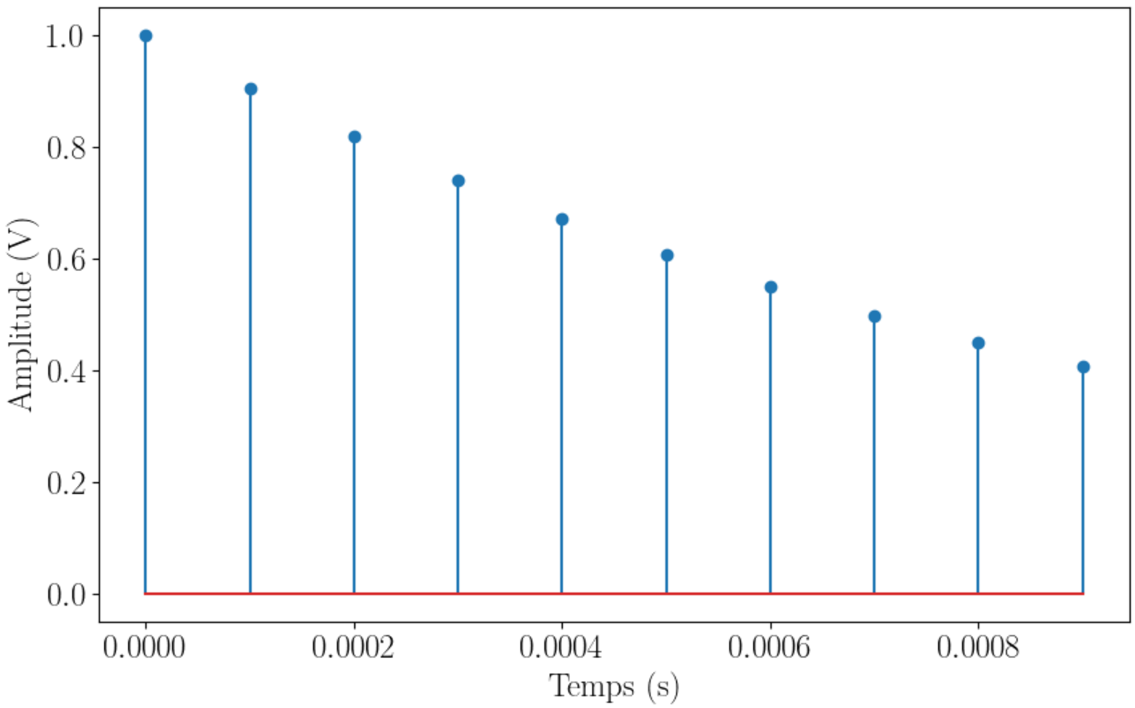


Figure 3 – Signal $s(t)$ d'une durée de 1 ms échantillonné à 10 kHz

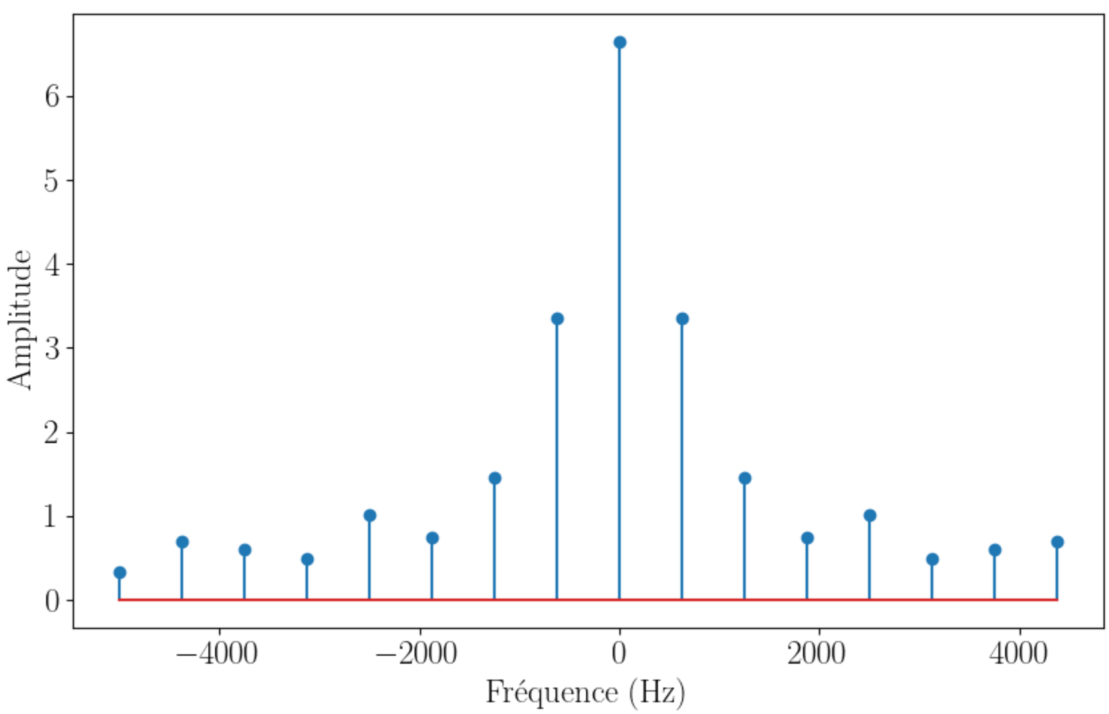
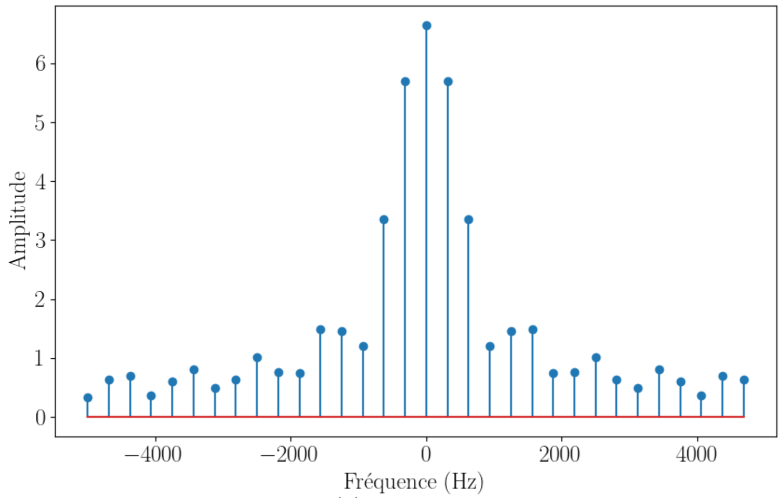
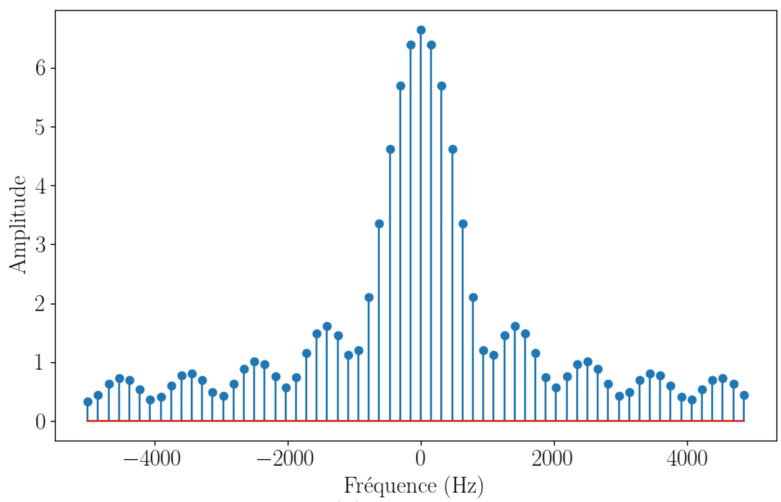


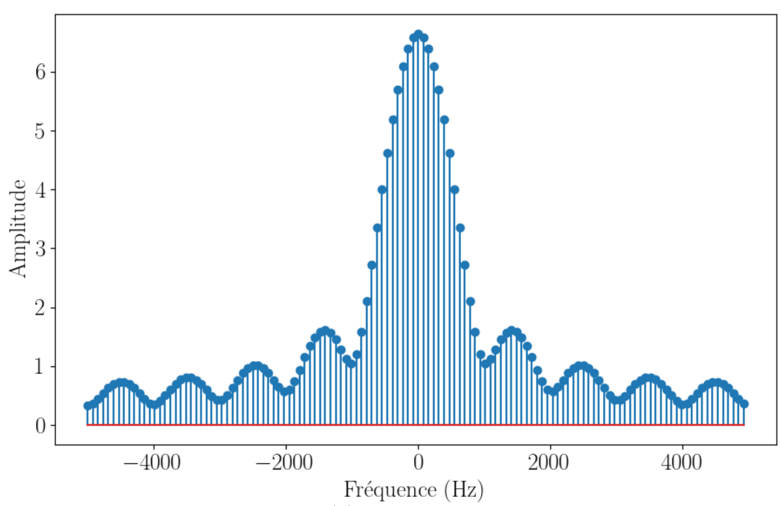
Figure 4 – Spectre du signal $s(t)$ d'une durée de 1 ms échantillonné à 10 kHz



(a) $n_{\mathrm{FFT}} = 32$



(b) $n_{\mathrm{FFT}} = 64$



(c) $n_{\mathrm{FFT}} = 128$

FIGURE 5 – Spectre de $s(t)$ d'une durée de 1 ms pour différentes valeurs de n_{FFT}