



24 mai 2025

OUALDI-DJEBRIL Inès - WANG Mélina

SAR Traitements audio

Partie signal



IMT Atlantique

Bretagne-Pays de la Loire
École Mines-Télécom

Table des matières

1 Synthèse additive	4
1.1 Analyse d'un son harmonique	4
1.2 Synthèse	6
2 Synthèse soustractive	8
3 Effets audio-numériques	13
3.1 Effets de réverbération	13
3.1.1 Mesure de réponse impulsionnelle	13
3.1.2 Convolution classique	15
3.1.3 Convolution rapide	15

Introduction

L'objectif de ce projet est d'explorer et d'implémenter les principales techniques de traitement du signal appliquées à l'audio numérique. Il s'inscrit dans le cadre de l'enseignement SAR (~~Systemes Audio et Réseaux~~) et repose sur l'utilisation de l'environnement MATLAB, outil de référence dans le domaine du signal.

Non, Situation
Apprentissage
Riche.

À travers plusieurs manipulations expérimentales et simulations, ce projet propose d'aborder trois grandes familles de traitements audio :

- La **synthèse sonore**, avec la comparaison des approches *additive* et *soustractive*
- L'**analyse fréquentielle** via la transformée de Fourier discrète (TFD)
- L'implémentation d'**effets audio-numériques** tels que la réverbération et le délai

Les différents travaux s'appuient sur des sons réels (fichiers .wav) et visent à illustrer la manière dont les outils mathématiques permettent de *modéliser*, *manipuler*, et *modifier* des signaux audio.

Ce rapport présente successivement :

- Une analyse fréquentielle de signaux d'instruments
- Des techniques de synthèse basées sur les harmoniques et les transformées de Fourier
- La mise en œuvre de filtres pour le traitement en fréquence ou en temps (effets de réverbération et delay)

1 Synthèse additive

1.1 Analyse d'un son harmonique

Question 1.1

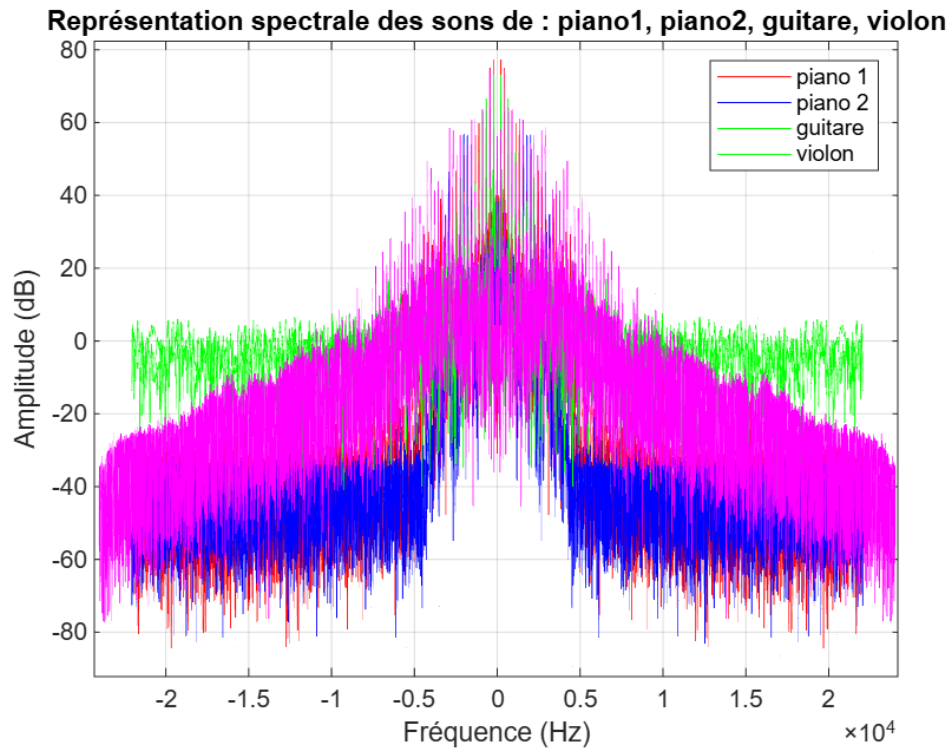


Figure 1 – Spectre en amplitude (dB)

On lit les fréquences fondamentales suivantes :

- Piano 1 : 220,456 Hz
- Piano 2 : 217,931 Hz
- Guitare : 219,889 Hz
- Violon : 444,425 Hz



Question 1.2

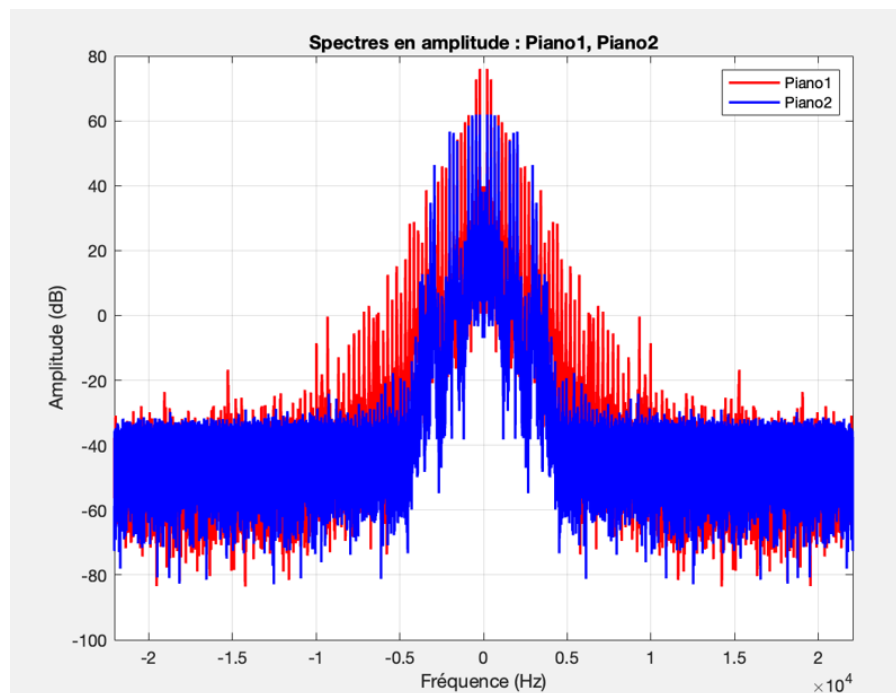


Figure 2 – Spectre en amplitude des pianos (dB)

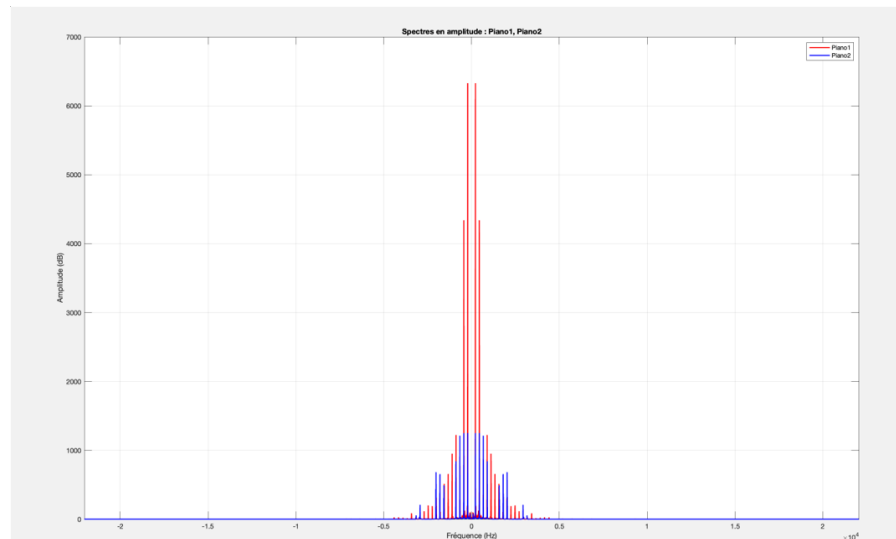


Figure 3 – Spectre en amplitude des pianos en valeur absolue



1. Synthèse additive

A l'aide de la formule du degré d'inharmonicité donnée dans l'énoncé, nous obtenons les tableaux suivants pour les pianos 1 et 2.

Fréquence (Hz)	f_2 (Hz)	f_3 (Hz)	f_4 (Hz)	f_5 (Hz)	f_6 (Hz)	f_7 (Hz)
Théorique	440.912	661.368	881.824	1102.28	1322.736	1543.192
Mesurées	442.124	663.307	885.622	1108.42	1331.54	1556.45
Inharmonicité	4.7524	5.0682	7.4404	9.6167	11.4847	14.8100

Figure 4 – Tableau de fréquence pour le piano 1

Fréquence (Hz)	f_2 (Hz)	f_3 (Hz)	f_4 (Hz)	f_5 (Hz)	f_6 (Hz)	f_7 (Hz)
Théorique	435.862	653.793	871.724	1089.655	1307.586	1525.517
Mesurées	443.004	668.077	892.717	11572.26	1797.34	2022.41
Inharmonicité	28.1380	37.4166	41.1978	634.7621	550.7490	488.1327

*Vous avez
à rater
des pics ici.*

Figure 5 – Tableau de fréquence pour le piano 2

En comparant les fréquences théoriques et les fréquences mesurées, on remarque que pour le piano 1, les composantes fréquentielles sont plus ou moins placées sur des multiples entiers de la fréquence fondamentale contrairement au piano 2 où les composantes fréquentielles s'éloignent de plus en plus des multiples entiers de la fréquence fondamentale. On en déduit que le piano 1 est plus harmonieux que le piano 2 et cela est bien en accord avec les valeurs d'inharmonicité.



1.2 Synthèse

Question 1.3

D'après la question précédente, le son le plus harmonique est celui du piano 1. On extrait ainsi les amplitudes des 8 premières harmoniques :

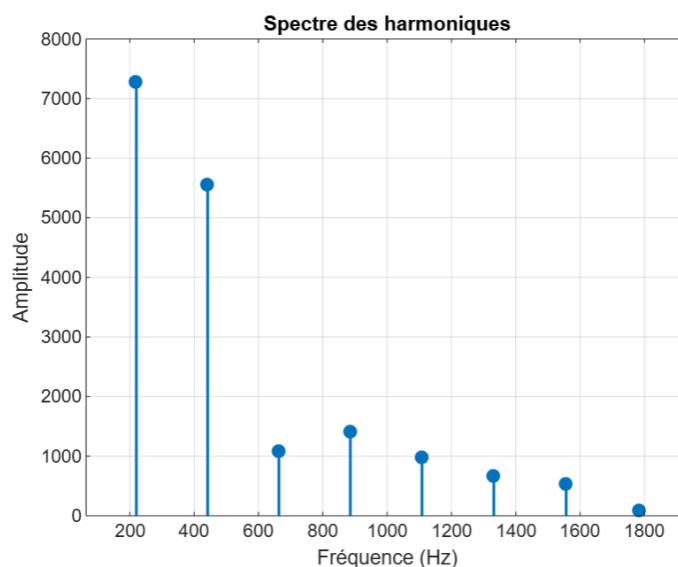


Figure 6 – Spectre en amplitude du piano 1



En superposant les ondes sinusoïdales aux fréquences et amplitudes correspondantes, on obtient le graphique suivant :

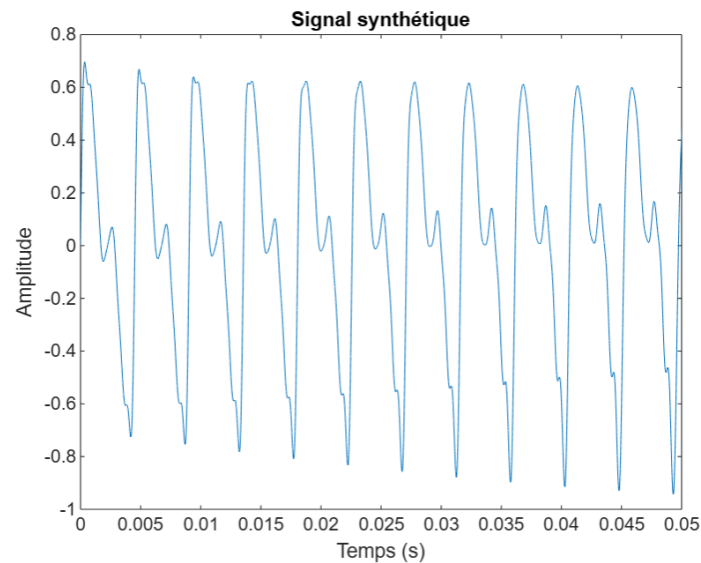


Figure 7 – Représentation graphique du son de piano 1 synthétisé

On peut ensuite écouter le son généré dans le fichier *synth_piano1.wav*.

Question 1.4

Pour ajouter une enveloppe de type ADSR, il faut choisir des temps d'attaque (*attackTime*), de chute (*decayTime*), de maintien (*sustainTime*) et de relâchement (*releaseTime*). Ces temps dépendent de l'instrument à modéliser, mais dans notre cas, nous avons choisi les valeurs de sorte à se rapprocher le plus possible du graphique représentant l'amplitude du son du piano en fonction du temps. On obtient alors les graphiques suivants :

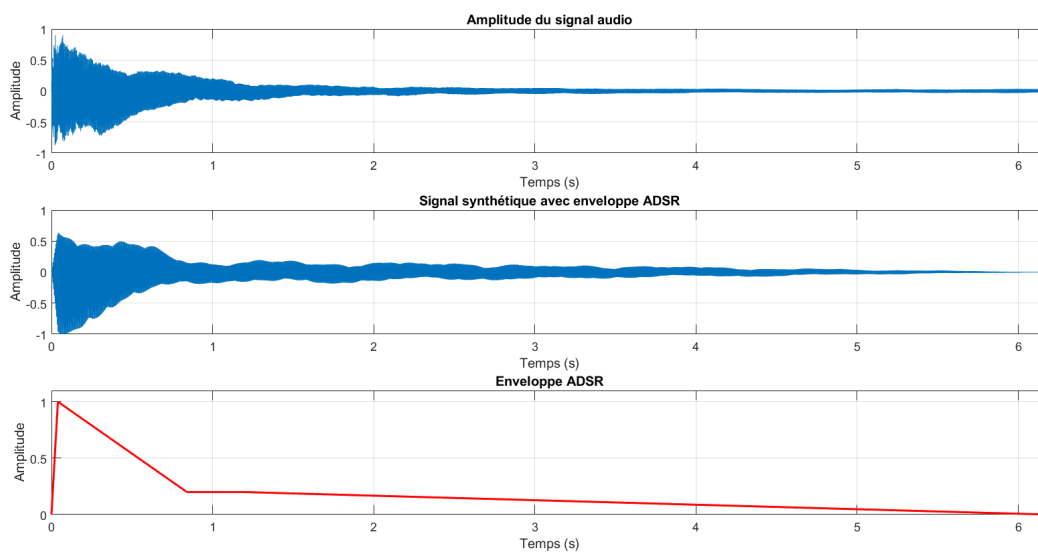


Figure 8 – Représentation graphique du son de piano 1 synthétisé avec enveloppe ADSR

Le son peut ensuite être écouté dans le fichier *synth_piano1_adsr.wav*.

Question 1.5

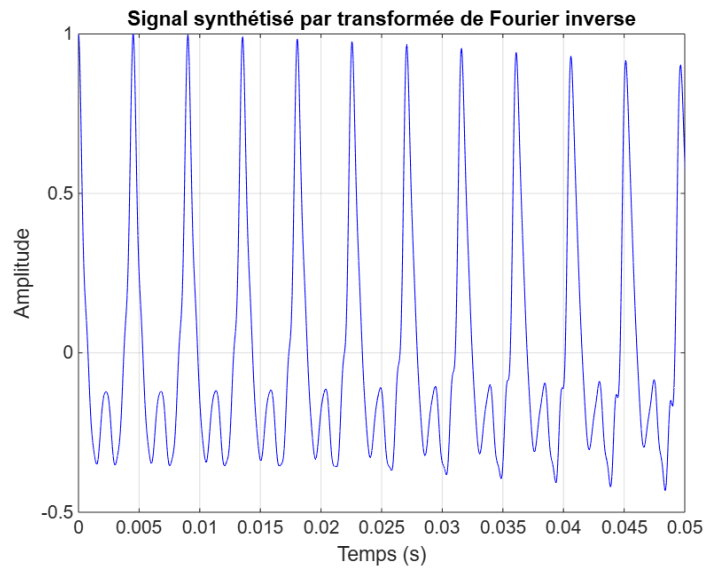


Figure 9 – Représentation graphique du son de piano 1 synthétisé par transformée de Fourier discrète inverse

On constate qu'en faisant une synthèse, en passant cette fois-ci par la transformée de Fourier discrète inverse, nous n'obtenons pas exactement le même résultat graphiquement. Cependant, à l'écoute du fichier `synth_piano1_idft.wav`, les deux sons semblent similaires.

↳ souvent dû à des différences de phase, ou d'échantillonnage.

2 Synthèse soustractive

Question 2.1

Un signal carré centré et périodique de période T est défini tel que :

$$s(t) = \begin{cases} -1, & -\frac{T}{2} < t < 0 \\ 1, & 0 < t < \frac{T}{2} \end{cases} \quad \text{avec} \quad s(t+T) = s(t)$$

Puisque la fonction est impaire, son développement en série de Fourier ne contient que les harmoniques impaires. La série de Fourier d'un signal impair et périodique s'écrit donc :

$$s(t) = \sum_{n=1}^{\infty} b_n \sin\left(\frac{2\pi n t}{T}\right)$$

Avec

$$\begin{aligned}
 b_n &= \frac{2}{T} \int_{-\frac{T}{2}}^{\frac{T}{2}} s(t) \sin\left(\frac{2\pi nt}{T}\right) dt \\
 &= \frac{2}{T} \left[\int_0^{\frac{T}{2}} \sin\left(\frac{2\pi nt}{T}\right) dt - \int_{-\frac{T}{2}}^0 \sin\left(\frac{2\pi nt}{T}\right) dt \right] \\
 &= \frac{4}{T} \int_0^{\frac{T}{2}} \sin\left(\frac{2\pi nt}{T}\right) dt \\
 &= \frac{4}{T} \times \left[-\frac{T}{2\pi n} \cos\left(\frac{2\pi nt}{T}\right) \right]_0^{\frac{T}{2}} \\
 &= \frac{4}{T} \times \frac{T}{2\pi n} \times (1 - (-1)^n) \\
 &= \frac{2}{\pi n} \times (1 - (-1)^n)
 \end{aligned}$$

On obtient donc les coefficients :

$$b_n = \begin{cases} \frac{4}{\pi n}, & \text{si } n \text{ impair} \\ 0, & \text{si } n \text{ pair} \end{cases}$$



Un signal dent de scie centré et périodique de période T est défini tel que :

$$s(t) = \frac{2t}{T} \quad \text{pour } t \in \left[-\frac{T}{2}, \frac{T}{2}\right]$$

Puisque la fonction est impaire, son développement en série de Fourier ne contient que les harmoniques impaires. La série de Fourier d'un signal impair et périodique s'écrit donc :

$$s(t) = \sum_{n=1}^{\infty} b_n \sin\left(\frac{2\pi nt}{T}\right)$$

Avec

$$\begin{aligned}
 b_n &= \frac{2}{T} \int_{-\frac{T}{2}}^{\frac{T}{2}} s(t) \sin\left(\frac{2\pi nt}{T}\right) dt \\
 &= \frac{2}{T} \int_{-\frac{T}{2}}^{\frac{T}{2}} \frac{2t}{T} \sin\left(\frac{2\pi nt}{T}\right) dt \\
 &= \frac{4}{T^2} \int_{-\frac{T}{2}}^{\frac{T}{2}} t \sin\left(\frac{2\pi nt}{T}\right) dt \\
 &= \frac{4}{T^2} \left[-t \frac{T}{2\pi n} \cos\left(\frac{2\pi nt}{T}\right) + \frac{T^2}{4\pi^2 n^2} \sin\left(\frac{2\pi nt}{T}\right) \right]_{-\frac{T}{2}}^{\frac{T}{2}} \\
 &= -\frac{2}{n\pi} (-1)^n
 \end{aligned}$$



On obtient donc les coefficients :

$$b_n = \frac{2}{n\pi} (-1)^{n+1}$$

Pour vérifier numériquement le spectre théorique d'un signal carré périodique, ou un signal dent de scie périodique, on peut adopter une approche fondée sur l'analyse de Fourier. Le principe consiste à comparer les coefficients théoriques issus du développement en série de Fourier aux composantes fréquentielles obtenues par une transformée de Fourier discrète (FFT) appliquée à une version échantillonnée du signal. Pour cela, on génère le signal souhaité, en s'assurant d'avoir plusieurs périodes et une fréquence d'échantillonnage suffisamment élevée pour capturer correctement les discontinuités. On applique ensuite la FFT au signal échantillonné afin d'en extraire les composantes fréquentielles, que l'on compare aux coefficients théoriques de la série de Fourier.

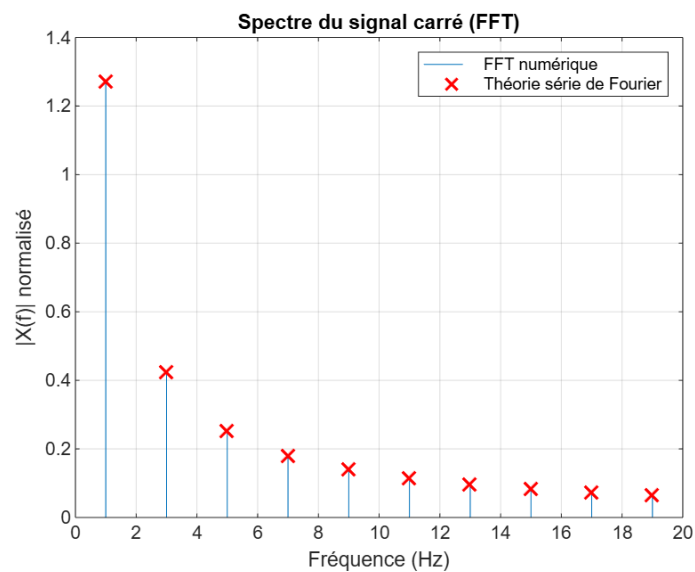


Figure 10 – Vérification numérique du signal carré

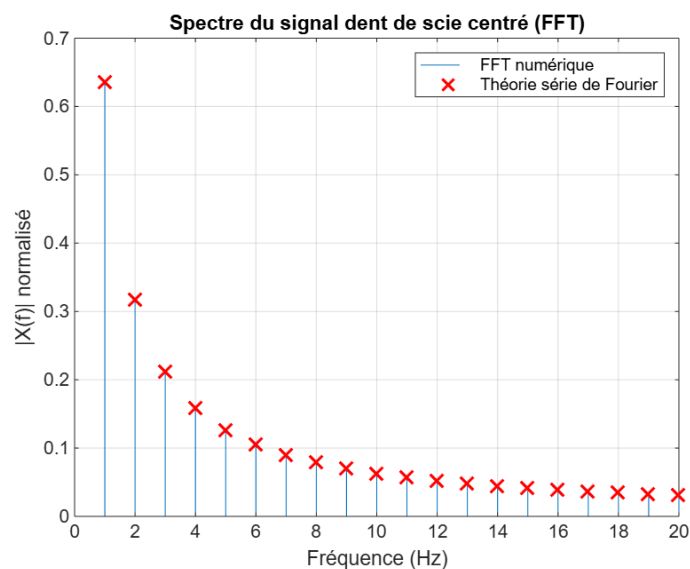


Figure 11 – Vérification numérique du signal dent de scie

Cependant, cette méthode a plusieurs limites :

- La résolution en fréquence dépend de la durée du signal : si le signal est trop court, les pics dans le spectre seront plus larges, ce qui rendra les fréquences plus difficiles à identifier précisément
- Le repliement spectral peut survenir si la fréquence d'échantillonnage est insuffisante (phénomène d'aliasing) ✓

Question 2.2

Considérons le filtre passe-bas d'ordre 1 :

$$y(k) = \frac{1}{2}(x(k) + x(k-1))$$

Le filtre a pour fonction de transfert :

$$H(f) = \frac{1}{2}(1 + \exp(-j2\pi \frac{f}{f_e}))$$

Calculons son module :

$$\begin{aligned} |H(f)| &= \left| \frac{1}{2}(1 + \exp(-j2\pi \frac{f}{f_e})) \right| \\ &= \frac{1}{2} |1 + \cos(2\pi \frac{f}{f_e}) - j \sin(2\pi \frac{f}{f_e})| \\ &= \sqrt{\frac{1 + \cos(2\pi \frac{f}{f_e})}{2}} \end{aligned}$$

D'où :

$$|H(f)| = |\cos(\pi \frac{f}{f_e})|$$

Après simulation, nous obtenons les résultats suivants :

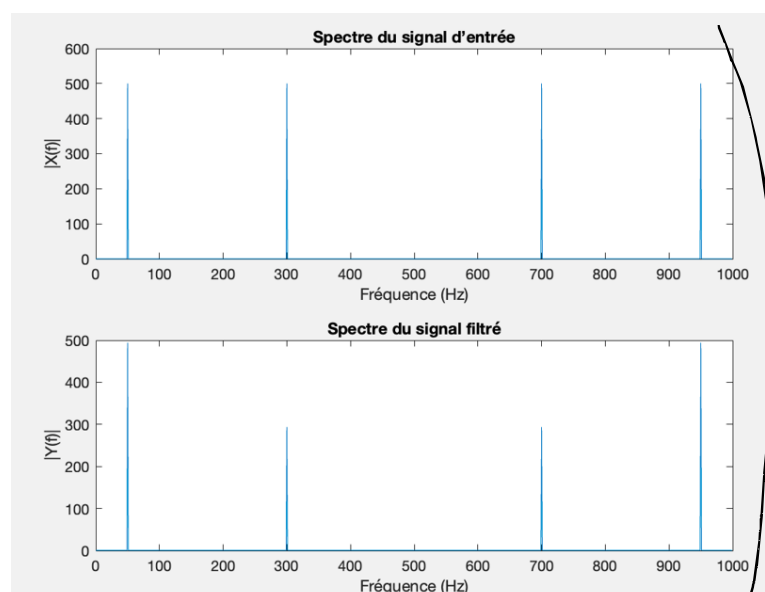


Figure 12 – Simulation du filtre passe-bas

résultat correct, mais trompeur. Aurait été mieux en utilisant "fftshift", pour ne pas confondre les fréquences $> f_s$ avec des hautes fréquences.

En calculant le gain du filtre aux fréquences représentées, puis en multipliant ce gain par les amplitudes du signal d'entrée, on retrouve bien les amplitudes du signal filtré. On constate aussi que le signal de basse fréquence n'est pas filtré contrairement à celui de haute fréquence, ce qui valide l'hypothèse d'un passe-bas. ✓

Question 2.3

Après simulation, on constate que les sons produits par synthèse soustractive et additive sont identiques (au nombre d'harmoniques près pour la synthèse additive). Ce résultat peut cependant différer dans le cas où le filtre utilisé lors de la synthèse soustractive n'est pas adapté aux fréquences du signal à filtrer. On remarque aussi qu'entre le signal dent de scie, qui est un signal riche en harmoniques, et le signal carré, le son perçu est différent. C'est ce qu'on peut écouter dans les fichiers audio : `sawtooth_add.wav`, `sawtooth_sous.wav`, `square_add.wav` et `square_sous.wav`. *→ c'est-à-dire ? Nécessite beaucoup d'harmoniques ? Aurait été intéressant de développer.*

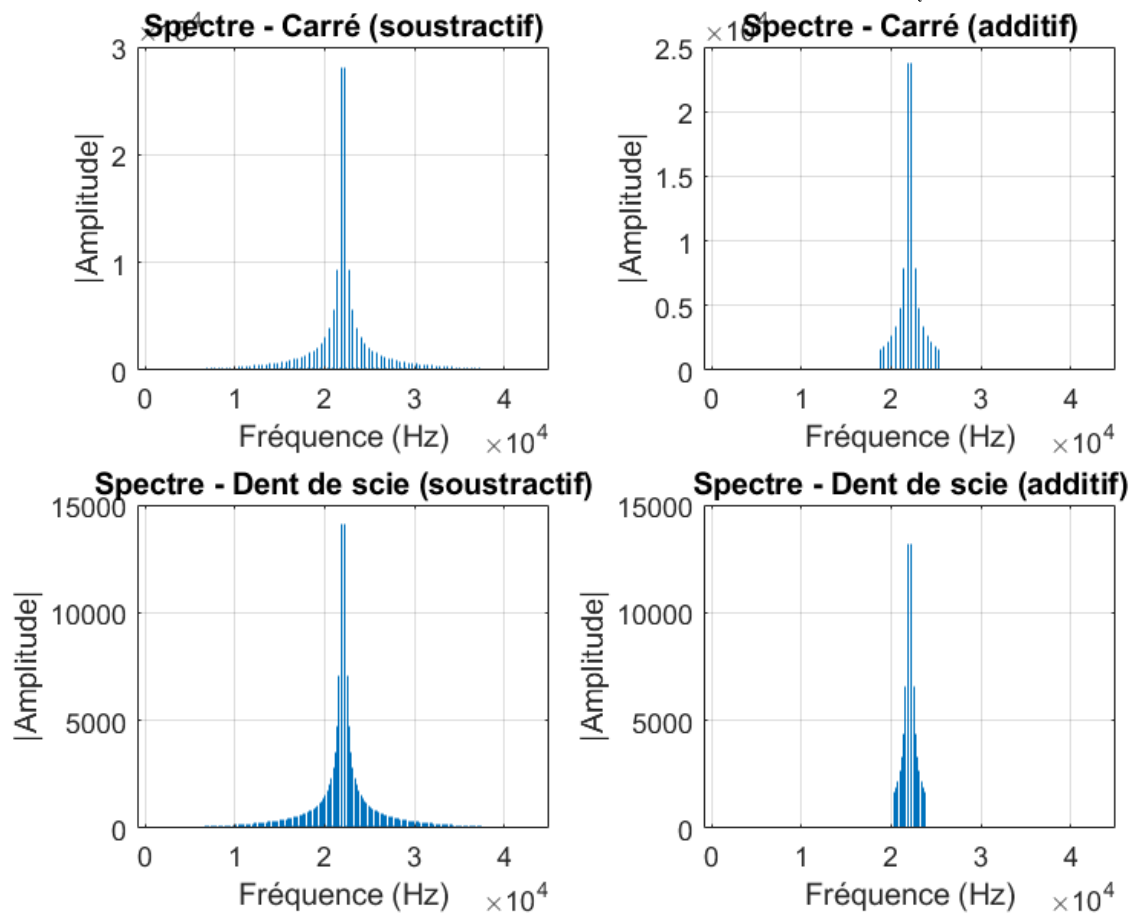


Figure 13 – Comparaison synthèse soustractive et additive

OK, mais on s'attendait plutôt à ce que vous compariez aux résultats des questions de la partie I, où la différence de rendu apparaît de façon flagrante.

Question 2.4

En testant différents types de filtres, nous constatons que les filtres FIR produisent un son plus clair et plus stable que les filtres IIR : ils donnent une meilleure qualité sonore. Concernant la fréquence de coupure, une valeur trop basse rend le son trop étouffé, tandis qu'une fréquence trop haute laisse passer trop d'aigus. Ainsi, il faudrait plutôt choisir une fréquence intermédiaire pour garder un bon équilibre. Pour l'ordre du filtre, un ordre élevé améliore la précision du filtrage et la netteté du son, alors qu'un ordre trop faible donne un filtrage moins efficace et un son moins défini.

Par conséquent, pour améliorer le son, il est préférable d'utiliser un filtre FIR avec un ordre suffisamment élevé et une fréquence de coupure bien choisie (ni trop haute, ni trop basse). On peut écouter le son final dans le fichier audio *son_ameliore.wav*.

3 Effets audio-numériques

3.1 Effets de réverbération

3.1.1 Mesure de réponse impulsionnelle

Question 3.1

Par définition de l'intercorrélation de deux signaux d'énergie finie :

$$R_{yx}(\tau) = y(\tau) * x^*(-\tau) = \int y(t)x^*(t-\tau)dt$$

Comme $y(t) = \int h(u)x(t-u)du$ on a finalement :

$$\begin{aligned} R_{yx}(\tau) &= \int \left(\int h(u)x(t-u)du \right) x^*(t-\tau)dt \\ &= \int h(u) \left(\int x(t-u)x^*(t-\tau)dt \right) du \\ &= \int h(u)R_{xx}(\tau-u)du \\ &= (h * R_{xx})(\tau) \end{aligned}$$

On obtient donc la relation :

$$R_{yx} = h * R_{xx}$$

Question 3.2

En supposant que $R_{xx}(u) \approx d(u)$, on a :

$$R_{yx}(u) \approx (h * d)(u)$$

D'où :

$$R_{yx} \approx h$$

Question 3.3

Pour déterminer quel signal est le plus adapté pour mesurer la réponse impulsionnelle d'une pièce, on cherche un signal d'excitation dont l'autocorrélation est la plus proche possible d'un Dirac (impulsion unité) : un pic fort en zéro et quasi nul ailleurs.

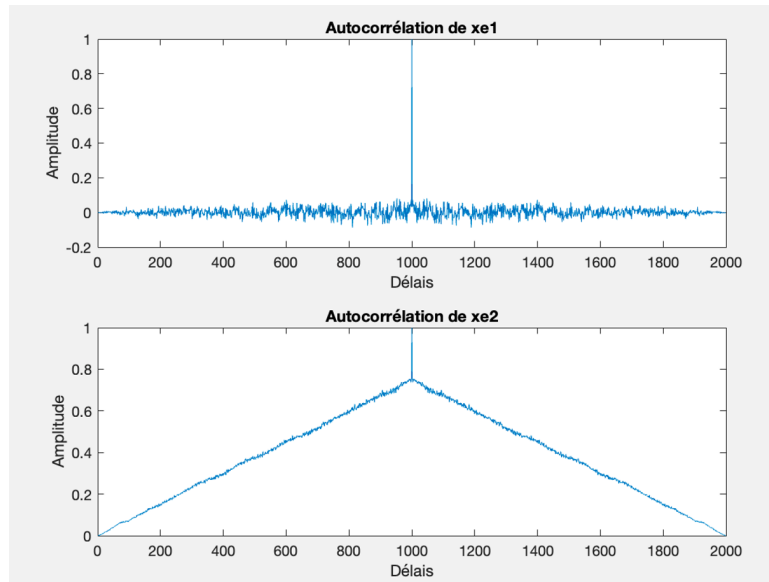


Figure 14 – Représentation graphique de l'autocorrélation de xe1 et xe2

On constate que l'autocorrélation du signal xe1 est plus proche d'une impulsion de Dirac que celle du signal xe2 : xe1 est donc mieux adapté comme signal d'excitation pour mesurer la réponse impulsionnelle de la pièce.

Question 3.4

On obtient la réponse impulsionnelle suivante :

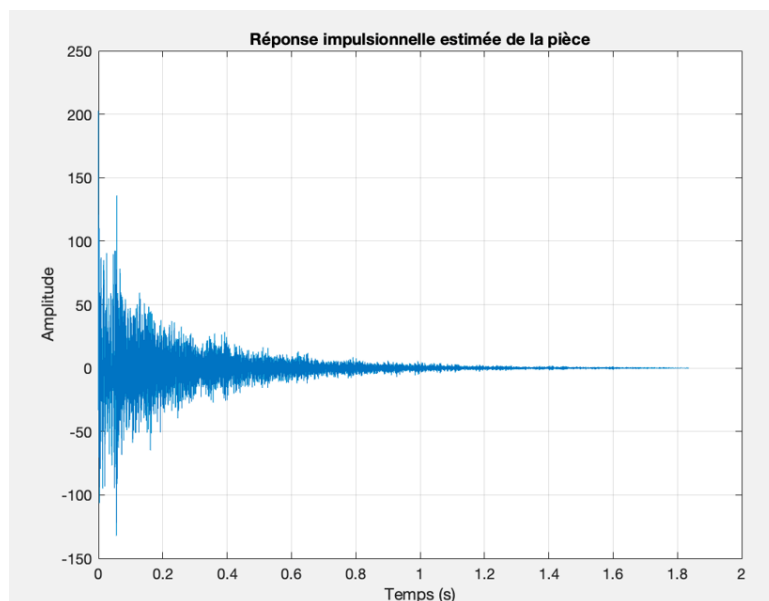


Figure 15 – Réponse impulsionnelle estimée en fonction du temps

3.1.2 Convolution classique

Question 3.5

Voir le code MATLAB

Question 3.6

Après avoir lancé le script `test_effet_reverb`, nous obtenons un temps de calcul de 286,83 secondes.

↳ c'est étrangement long!

3.1.3 Convolution rapide

Question 3.7

En réalisant le filtrage dans le domaine fréquentiel, on obtient un temps de calcul de 0,12 secondes, ce qui est bien plus rapide que le temps de calcul obtenu par application de convolution directe. Cela peut être expliqué par le fait que, dans le domaine fréquentiel, on utilise des fonctions qui ont déjà été optimisées, tandis que dans le cas de la convolution directe, on utilise des boucles imbriquées, augmentant le temps de calcul.

Question 3.8

↳ Ok, mais provient aussi du fait que : conv en temporel \Leftrightarrow produit en fréquentiel.

Mathématiquement, la convolution dans le domaine temporel est équivalente à une multiplication dans le domaine fréquentiel :

$$x * h = \mathcal{F}^{-1}(\mathcal{F}(x) \cdot \mathcal{F}(h))$$

ah, voilà!

Cependant, numériquement, afin que les deux méthodes puissent donner les mêmes résultats, il y a certaines conditions à respecter notamment le fait qu'il ne faut pas qu'il y ait de repliement temporel (la taille de la transformée de Fourier doit être supérieure à la taille de x + la taille de $h - 1$: $L_{x*h} \geq L_x + L_h - 1$). De plus, puisque la transformée de Fourier applique naturellement une convolution circulaire, il ne faut pas oublier de compléter les signaux par des zéros (zero-padding). Enfin, il faut prendre la partie réelle de l'ifft pour ne pas prendre en compte les imprécisions numériques.

Question 3.9

On considère la relation de récurrence suivante ($\tau \in N$) :

$$y(k) = x(k) - gy(k - \tau)$$

En posant $x(k) = d(k)$, la réponse impulsionnelle $h(k)$ du système vérifie $h(k) = u(k)(d(k) - gh(k - \tau))$. On peut alors trouver une expression explicite de $h(k)$ en développant la relation de récurrence, à savoir :

$$h(0) = d(0) - gh(-\tau) = 1$$

$$h(\tau) = d(1) - gh(0) = -g$$

$$h(2\tau) = d(2) - gh(\tau) = (-g)^2$$

⋮

On obtient alors :

$$h(k) = \begin{cases} (-g)^n, & \text{si } k = n\tau \\ 0, & \text{sinon} \end{cases}$$

→ ok
→ comment c'est justifié?

Question 3.10

Pour que le filtre de la figure soit stable, il faut que sa réponse impulsionnelle soit sommable. Ainsi, g doit être dans $] -1; 1[$.
ok.

Question 3.11

Pour rappel, le vecteur a est un vecteur comportant les coefficients de la partie réursive tandis que le vecteur b est un vecteur comportant les coefficients de la partie non-réursive.

D'après l'équation de Delay :

$$y(k) + gy(k - \tau) = x(k)$$

Ainsi, nous avons les vecteurs suivants :

$$a = [1, 0, \dots, 0, g] \text{ et } b = [1]$$

Avec a de taille $\tau + 1$

Question 3.12

Après exécution du script `analyse_delay`, nous obtenons le graphique suivant :

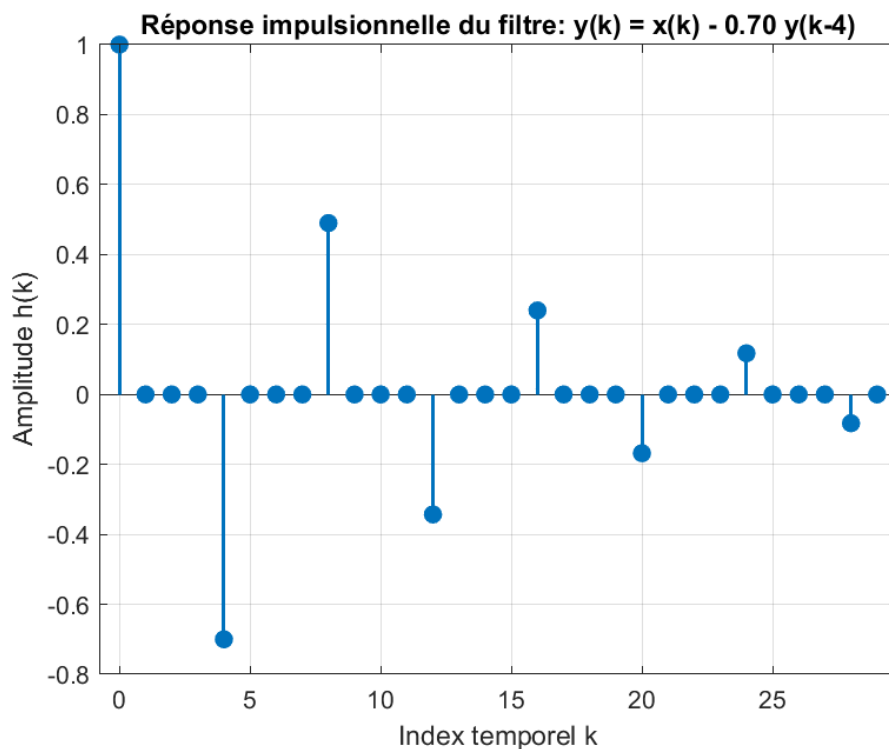


Figure 16 – Représentation temporelle de $h(k)$

Question 3.13

On considère l'équation de récurrence suivante :

$$y(k) = x(k) - gy(k - \tau)$$

On applique la transformée de Fourier discrète aux deux membres et on factorise par $Y(\nu)$:

$$Y(\nu) = X(\nu) - ge^{-j2\pi\nu\tau}Y(\nu)$$

$$Y(\nu)(1 + ge^{-j2\pi\nu\tau}) = X(\nu)$$

On en déduit la réponse en fréquence du filtre :

$$\hat{h}(\nu) = \frac{Y(\nu)}{X(\nu)} = \frac{1}{1 + ge^{-j2\pi\nu\tau}}$$

On calcule ensuite le module :

$$\begin{aligned} |\hat{h}(\nu)| &= \frac{1}{|1 + ge^{-j2\pi\nu\tau}|} \\ &= \frac{1}{\sqrt{1 + 2g \cos(2\pi\nu\tau) + g^2}} \end{aligned}$$

Ainsi, le module est :

$$|\hat{h}(\nu)| = \frac{1}{\sqrt{1 + 2g \cos(2\pi\nu\tau) + g^2}}$$

Pour la phase, cette dernière est donnée par l'argument du dénominateur :

$$\arg(\hat{h}(\nu)) = -\arg(1 + ge^{-j2\pi\nu\tau})$$

Soit en notant $z = ge^{-j2\pi\nu\tau}$:

$$\arg(\hat{h}(\nu)) = -\arctan\left(\frac{g \sin(2\pi\nu\tau)}{1 + g \cos(2\pi\nu\tau)}\right)$$

Question 3.14

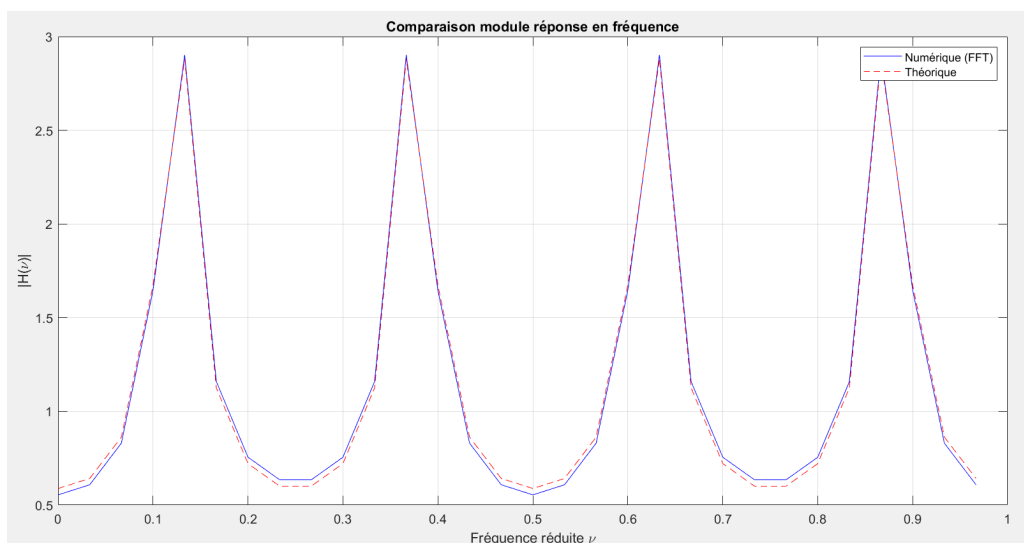


Figure 17 – Comparaison entre le module de la réponse en fréquence obtenu théoriquement et numériquement pour un échantillon de 30 points

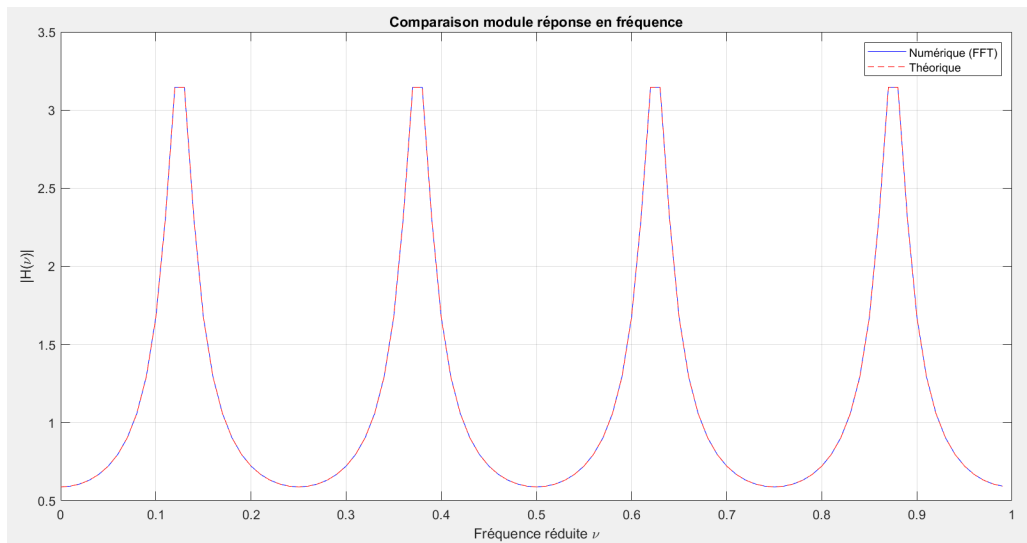


Figure 18 – Comparaison entre le module de la réponse en fréquence obtenu théoriquement et numériquement pour un échantillon de 100 points

On remarque que les courbes théorique et numérique de la réponse en fréquence du filtre de delay sont globalement très proches, confirmant la validité du modèle théorique. Cependant, on observe de légères différences entre les amplitudes des deux courbes, surtout entre les pics de résonance. Aussi, on constate que sur un échantillon de 30 points, les modules théorique et numérique ne se superposent pas tandis que dans le cas d'un échantillon de 100 points, ces derniers se superposent. On en conclut que les différences observées sont dues au fait qu'on a pris un petit échantillon, ce qui rend les graphiques moins précis.

Question 3.15

Voir le code MATLAB

Question 3.16

Voir le fichier *signal_delay.wav*.

Question 3.17

Voir le code MATLAB

Question 3.18

Voir le fichier *piano_delay_filtre.wav*

Question 3.19

Le filtre considéré a pour réponse impulsionnelle :

$$h_r(k) = \begin{cases} \frac{1}{K} & \text{pour } 0 \leq k < K \\ 0 & \text{sinon} \end{cases}$$

Pour étudier son effet fréquentiel, on calcule sa transformée de Fourier discrète :

$$H(f) = \sum_{k=-\infty}^{\infty} h_r(k) \cdot e^{-j2\pi f k}$$

Or, comme $h_r(k) = 0$ pour $k < 0$ ou $k \geq K$, la somme se réduit à :

$$\begin{aligned} H(f) &= \sum_{k=0}^{K-1} \frac{1}{K} \cdot e^{-j2\pi f k} \\ &= \frac{1}{K} \sum_{k=0}^{K-1} e^{-j2\pi f k} \\ &= \frac{1}{K} \cdot \frac{1 - e^{-j2\pi f K}}{1 - e^{-j2\pi f}} \\ &= \frac{1}{K} \cdot e^{-j\pi f (K-1)} \cdot \frac{\sin(\pi f K)}{\sin(\pi f)} \end{aligned}$$

Le module de la réponse fréquentielle est donc :

$$|H(f)| = \left| \frac{1}{K} \cdot \frac{\sin(\pi f K)}{\sin(\pi f)} \right|$$

On constate que le module $|H(f)|$ atteint un maximum en $f = 0$ avec $|H(0)| = 1$, et qu'il s'annule pour $f = \pm \frac{1}{K}, \pm \frac{2}{K}, \dots$. De plus, l'enveloppe de $|H(f)|$ décroît globalement lorsque la fréquence augmente, ce qui caractérise un filtre passe-bas et explique que le filtre $h_r(k)$ semble "couper" progressivement les hautes fréquences du signal. Cette propriété est d'ailleurs confirmée par la simulation numérique.

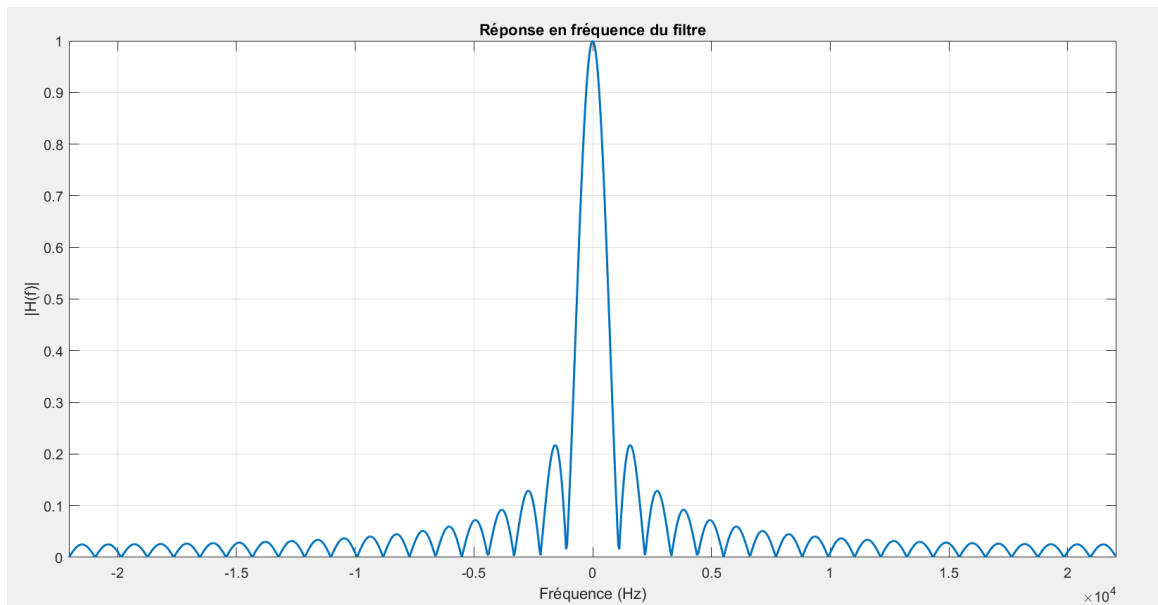


Figure 19 – Réponse en fréquence de l'effet de retard filtré

Conclusion

En conclusion, ce projet nous a permis de mieux comprendre comment des outils mathématiques et numériques peuvent être utilisés pour explorer et transformer le son. À travers des manipulations concrètes sur Matlab, nous avons non seulement mis en œuvre des techniques classiques du traitement du signal, mais aussi pris conscience de l'impact que peuvent avoir ces traitements sur notre perception de l'audio.

