Traitement du signal - Filtrage numérique

Université de Montpellier
M1 IMAGINE

Adèle Imparato

Janvier 2023

1 Introduction

Ce projet vise à expérimenter différents types de filtrage sur des données audio à l'aide de la transformée de Fourier directe et inverse. Les filtrages testés dans ce rapport sont les suivants: filtre passe bas, filtre passe haut, filtre coupe bande, filtre passe bande et filtre de Butterworth. Ils sont utilisés sur des données audio variées, comme des accords, des gammes, des notes simples, mais aussi des sons du quotidien. Ce projet a été réalisé en C++, vous trouverez le code ainsi que les fichiers audio générés à cette adresse: `https://github.com/Adeleimpa/traitement_du_signal.git`

2 Synthétisation d'une note simple

Pour commencer, on tente de générer un fichier .wav contenant une seule note (donc une seule fréquence) à l'aide des classes fournies Wave.cpp et Wave.hpp. On va commencer par synthétiser un La en mono pendant 6 secondes. Pour cela, il nous faut connaître la fréquence du La. Selon le tableau ci-dessous, un La peut avoir différentes fréquences en fonction de l'octave sélectionnée.

Note/octave	-1	0	1	2	3	4	5	6	7	8	9
do ou *si* ♯	16,35	32,70	65,41	130,81	261,63	523,25	1046,50	2093,00	4186,01	8 372,02	16 744,04
do ♯ ou *ré* ♭	17,33	34,65	69,30	138,59	277,18	554,37	1108,73	2217,46	4434,92	8 869,84	17 739,68
ré	18,36	36,71	73,42	146,83	293,66	587,33	1174,66	2349,32	4698,64	9 397,28	18 794,56
ré ♯ ou *mi* ♭	19,45	38,89	77,78	155,56	311,13	622,25	1244,51	2489,02	4978,03	9 956,06	19 912,12
mi ou *fa* ♭	20,60	41,20	82,41	164,81	329,63	659,26	1318,51	2637,02	5274,04	10 548,08	21 096,16
fa ou *mi* ♯	21,83	43,65	87,31	174,61	349,23	698,46	1396,91	2793,83	5587,65	11 175,30	22 350,60
fa ♯ ou *sol* ♭	23,13	46,25	92,50	185,00	369,99	739,99	1479,98	2959,96	5919,91	11 839,82	23 679,64
sol	24,50	49,00	98,00	196,00	392,00	783,99	1567,98	3135,96	6271,93	12 543,86	25 087,72
sol ♯ ou *la* ♭	25,96	51,91	103,83	207,65	415,30	830,61	1661,22	3322,44	6644,88	13 289,76	26 579,52
la	27,50	55,00	110,00	220,00	440,00	880,00	1760,00	3520,00	7040,00	14 080,00	28 160,00
la ♯ ou *si* ♭	29,14	58,27	116,54	233,08	466,16	932,33	1864,66	3729,31	7458,62	14 917,24	29 834,48
si ou *do* ♭	30,87	61,74	123,47	246,94	493,88	987,77	1975,53	3951,07	7902,13	15 804,26	31 608,52

Figure 1: Fréquences de notes en Hertz dans la gamme tempérée.[4]

On commence par synthétiser un La ayant une fréquence de 440Hz. Afin de générer ce fichier .wav, il va nous falloir une fréquence d'échantillonage. En standard, cette valeur vaut 44100Hz (=44100 relevés par seconde), respectant ainsi le critère de Shannon puisque la fréquence max standard est de 20000Hz ($44100 \geq 2 \times 20000$).

Pour construire notre La, il va ensuite nous falloir un tableau de données (correspondant aux valeurs de l'onde sinusoïdale générée par la note). Ces données sont calculées suivant la formule générale du signal sinuisoïdal:

$$s(t) = A \sin(\omega t + \varphi)$$

(en remplaçant les valeurs comme vu en cours). *Note:* réduire la fréquence d'échantillonage revient à réduire le nombre de données de notre tableau (=nombre d'échantillons).

La figure ci-dessous montre clairement que le signal a une forme sinusoïdale régulière.

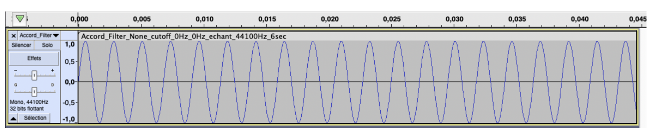


Figure 2: Zoom de l'onde sinusoïdale générée par un La de fréquence 440Hz et de fréquence d'échantillonage 44100Hz.

En changeant la fréquence à 523Hz, on obtient le même type de résultat mais cette fois pour la note Do. Sur la figure ci-dessous on voit que la courbe a une forme similaire à celle du La, à part que la période est plus petite (les ondes sont plus rapprochées par rapport au La) étant donné sa fréquence supérieure.

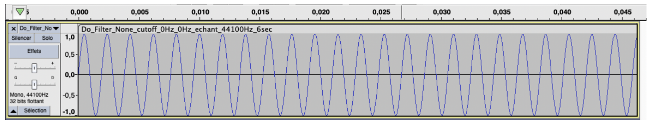
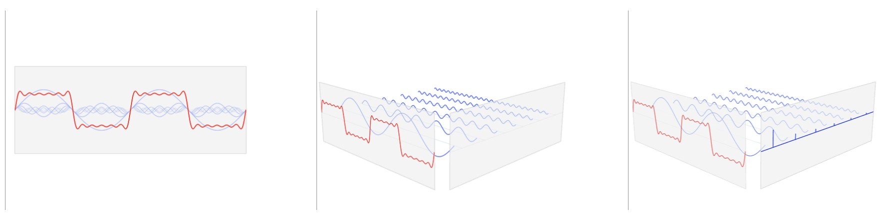


Figure 3: Zoom de l'onde sinusoïdale générée par un Do de fréquence 523Hz et de fréquence d'échantillonage 44100Hz.

3 Méthodes de transformations de Fourier discrètes directe et inverse

Tout d'abord qu'est-ce que la transformée de Fourier ? "La transformée de Fourier est une opération qui permet de représenter en fréquence des signaux qui ne sont pas périodiques."[2]. Afin d'y voir plus clair, voici un schéma réprésentant le concept de base de cette opération mathématique.



(a) Le signal en rouge est constitué de six signaux.

(b) La transformée de Fourier directe cherche à retrouver ces six signaux.

(c) Elle indique donc les six pics correspondants aux fréquences de ces signaux bleus.

Figure 4: Schéma de la transformée de Fourier directe [5]

On cherche donc à créer deux méthodes permettant de passer d'un signal à sa transformée et inversément (en temps discret).

3.1 Transformée de Fourier directe (TFD)

La transformée de Fourier directe d'un signal est définie par une suite de valeurs complexes qu'on peut noter

$$X_k = a_k + ib_k$$

avec

$$a_k = \Sigma_{n=0}^{N-1} x_n cos(2\pi \frac{kn}{N})$$

$$b_k = -\Sigma_{n=0}^{N-1} x_k sin(2\pi \frac{kn}{N})$$

et

$$k \in \{0, N-1\}$$

Note: N correspond au nombre d'échantillons.

3.2 Transformée de Fourier inverse (TFI)

La transformée de Fourier inverse d'un signal, quant à elle, permet de retomber sur le signal de base après lui avoir appliqué une transformée de Fourier directe. Elle prend donc en entrée la partie réelle et imaginaire calculée par la TFD (a_k et b_k). La transformée inverse, elle aussi, consiste en une suite de nombres complexes. Cependant, sa partie imaginaire est nulle dans notre cas, c'est pourquoi on peut se contenter de calculer sa partie réelle comme ceci:

$$x_n = \frac{1}{N}\Sigma_{k=0}^{N-1} a_k cos(2\pi\frac{kn}{N}) - b_k sin(2\pi\frac{kn}{N})$$

4 Lecture du fichier GammePiano.wav

A l'aide des fichiers Wave.cpp et Wave.hpp on peut lire un fichier de type .wav et obtenir son tableau de données pour ensuite lui appliquer des filtres. On considère le fichier GammePiano.wav et on visualise ses données:

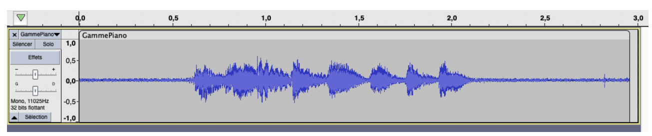


Figure 5: Visualisation de la courbe du fichier GammePiano.wav

5 Création d'une gamme

Afin de créer le fichier audio d'une gamme quelconque, il nous faut un tableau de données contenant les valeurs du signal pour chaque note qu'on combine à la suite afin d'obtenir les données pour la gamme.

Commençons par générer la gamme chromatique du La. On souhaite entendre ces notes dans cet ordre (mais pas en même temps): La-Si-Do-Ré-Mi-Fa-Sol. Pour ce faire, on va donc remplir un tableau de données pour un La sur un 7ème (car on a 7 notes dans notre gamme) du nombre d'échantillons total. On fait de même pour les six prochaines notes. A la fin, il suffit de prendre les tableaux de chaque note à la suite pour obtenir un tableau de données correspondant à la gamme (et qui aura donc une longuer de N échantillons). En suivant les valeurs données sur la Figure 1, on choisit une gamme de fréquences {440Hz, 494Hz, 523Hz, 587Hz, 659Hz, 698Hz, 784Hz}. Pour finir, on écrit ces données dans un fichier qu'on peut ensuite visualiser grâce à l'outil Audacity.

En observant l'onde sinusoïdale générée par la gamme chromatique du La, on voit clairement que toutes les 0.71 secondes (5s/7), la sinusoïde change de fréquence, indiquant qu'une nouvelle note est jouée.

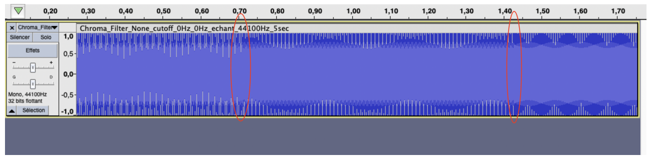


Figure 6: Signal généré par la gamme chromatique du La, sur 5 secondes avec une fréquence d'échantillonnage de 44100Hz. On peut voir en rouge le changement de sinusoïde lorqu'on passe d'une note à l'autre.

6 Création d'un accord

Pour créer le tableau de données d'un accord (et son fichier audio), on peut procéder de deux manières différentes. D'une part, on peut générer l'onde sinusoïdale pour chaque note de notre accord, et faire ensuite la moyenne des trois courbes pour obtenir les données de notre accord. C'est la manière 'intuitive'. Ou alors, on peut faire la transformée de Fourier de chaque note individuellement puis additionner les données pour obtenir les données de la TFD de notre accord. On a ensuite plus qu'à faire la transformée inverse pour retomber sur les données originales de l'accord (celles qu'on peut écouter et qui sonnent comme un accord).

Voici, à quoi ressemble le signal obtenu avec les deux méthodes pour un accord en Do majeur. On constate qu'ils sont identiques.

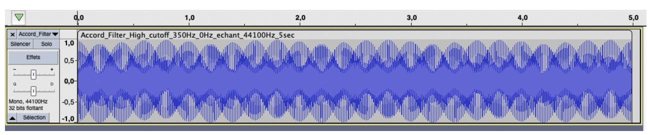


Figure 7: Signal généré par l'accord Do-Mi-Sol 'intuitivement' (durée: 5 secondes, fréquence d'échantillonnage: 44100Hz)

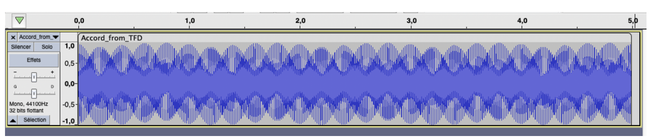


Figure 8: Signal généré par l'accord Do-Mi-Sol à l'aide des TFD (durée: 5 secondes, fréquence d'échantillonnage: 44100Hz)

Si on zoom sur le signal obtenu (Figure 10), on peut observer que la courbe correspond en effet à la somme des trois ondes sinusoïdales générées par les trois notes individuellement comme nous l'indique la Figure 9.

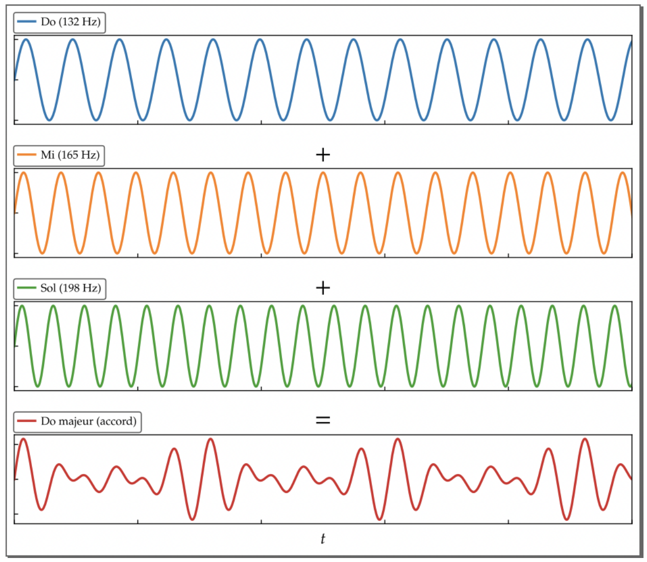


Figure 9: Ondes sonores composant l'accord do majeur [1].

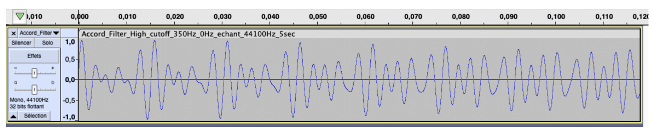


Figure 10: Zoom sur le signal généré par l'accord de Do majeur.

Les résultats sont similaire pour un accord en La mineur, si ce n'est que la Figure 12 n'a pas la même durée ni le même nombre d'échantillons.

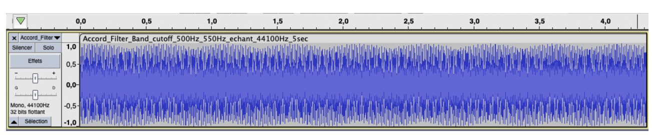


Figure 11: Signal généré par l'accord La-Do-Mi 'intuitivement' (durée: 5 secondes, fréquence d'échantillonnage: 44100Hz)

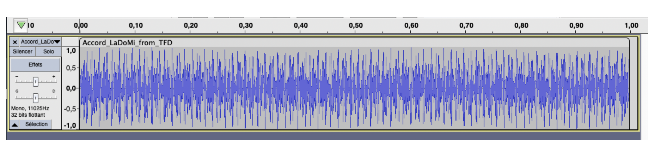


Figure 12: Signal généré par l'accord La-Do-Mi à l'aide des TFD (durée: 1 seconde, fréquence d'échantillonnage: 11025Hz)

7 Appliquer les transformées (TFD et TFI)

Maintenant que plusieurs sortes de données nous sont accessibles, on applique la transformée de Fourier discrète (directe et inverse) sur nos signaux.

7.1 Sur une note simple

Sur la TFD pour un La simple (Figure 13), on observe un premier pic à 0.05s et un deuxième à 4.95s. Pour convertir ces valeurs en fréquences, il suffit de multiplier le temps par la fréquence d'échantillonnage puis de diviser par la durée totale du signal. Ici, $0.05s \cdot 44100Hz/5s = 441Hz$, soit quasiment 440Hz, la fréquence du La. Quant au deuxième pic, celui-ci correspond simplement à la symétrie du pic initial, du au phénomène de repliement de spectre.

Figure 13: TFD d'un La à 440Hz (durée: 5 secondes, fréquence d'échantillonnage: 44100Hz). En rouge, zoom sur les pics.

Sur la TFI d'un La simple (Figure 14), on constate qu'on retombe simplement sur le signal d'origine (Figure 2).

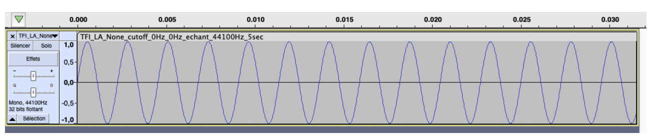


Figure 14: TFI d'un La à 440Hz (durée: 5 secondes, fréquence d'échantillonnage: 44100Hz).

7.2 Sur un accord

La TFD d'un accord devrait correspondre à quelque chose de ce genre (sans repliement de spectre):

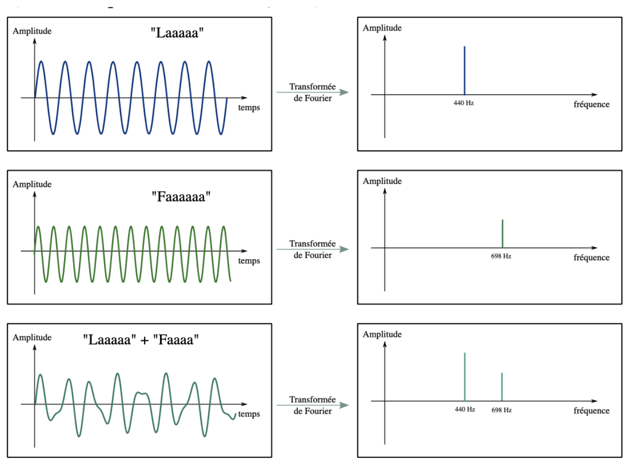


Figure 15: Représentation de la TFD d'un accord comprenant un La pur (440Hz) et un Fa pur (698Hz). [3]

Prenons à présent la TFD de notre accord en Do majeur. Sur celui-ci on observe six pics en tout. On va particulièrement s'intéresser aux trois premeirs puisque les trois derniers reflètent simplement le phénomène de repliement de spectre. Le premier pic se situe à 0.0295s, soit 260Hz, fréquence correspondant à un Do (voir Figure 1). Le deuxième pic se situe à 0.0375s, soit 330Hz la fréquence d'un Mi. Et enfin, le troisième pic se trouve à 0.0445s, ce qui équivaut à 392Hz la fréquence d'un sol. *Note:* la valeur temps manque de précision c'est pourquoi on ne retombe pas exactement sur la fréquence de la note originale.

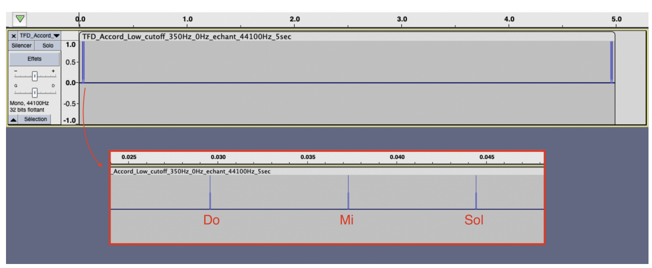


Figure 16: TFD d'un accord en Do majeur

La TFI de cet accord, sans filtre appliqué, correspond simplement au signal de l'accord avant de l'avoir transformé.

7.3 Sur une gamme

Lorsqu'on applique la TFD sur la gamme chromatique du La (précédemment créée), on obtient un signal à 7 pics, correspondant aux 7 notes de la gamme (La-Si-Do-Ré-Mi-Fa-Sol).

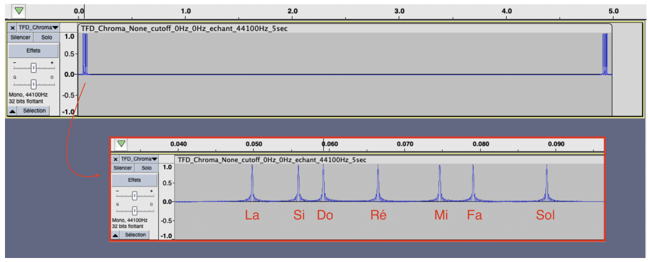


Figure 17: TFD de la gamme chromatique du La (durée = 5 secondes, fréquence d'échantillonnage = 44100Hz).

Pour la gamme piano (provenant du fichier GammePiano.wav), un résultat comparable peut être observés. On obtient 8 pics, correspondant aux huit notes de la gamme (Do-Ré-Mi-Fa-Sol-La-Si-Do).

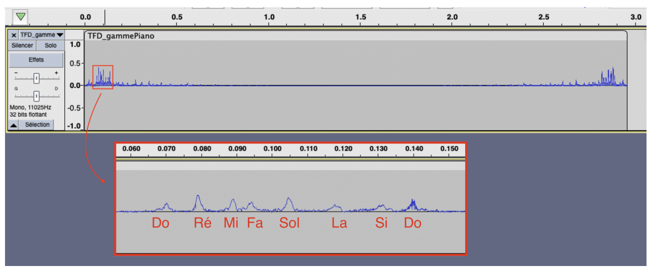


Figure 18: TFD de la gamme piano (durée = 2.95 secondes, fréquence d'échantillonnage = 11025Hz).

Sur le signal de la TFD de la gamme piano, on observe que les pics sont moins nets que sur la gamme chromatique du La créée artificiellement. Cela est du au fait que la gamme piano contient du bruit, les notes sont moins nettes et cela se reflète sur sa TFD.

Pour les deux gammes (chromatique et piano), leur TFI ne fait que nous donner le signal initial de la gamme. La TFI devient intéressante lorsqu'on applique un filtre sur la TFD d'un signal.

8 Les filtres

8.1 Passe bas

Un filtre passe bas est un filtre qui laisse passer uniquement les fréquences en dessous d'un certain seuil, appelé fréquence de coupure. Il prend donc en entrée la TFD d'un signal, trouve l'indice de la fréquence de coupure, et met ensuite à zéro toutes les valeurs après cet indice en faisant attention à garder les valeurs symétriques à celles conservées (voir Figure 19).

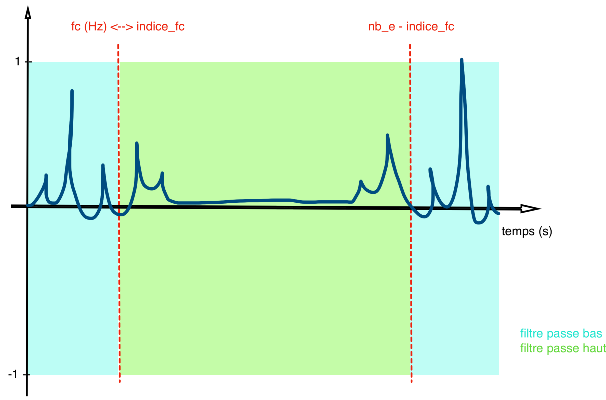


Figure 19: Schéma conceptuel de l'effet engendré par un filtre passe bas ou un filtre passe haut sur les données de la TFD d'un signal. En bleu: la zone du signal conservée par le filtre passe bas. En vert: la zone du signal conservée par le filtre passe haut.

8.2 Passe haut

Le filtre passe haut correspond à l'inverse du filtre passe bas. Il fonctionne comme indiqué sur la Figure 19.

8.3 Passe bande

Le filtre passe bande, permet de conserver les fréquences situées entre une première fréquence de coupure et une deuxième (voir Figure 20).

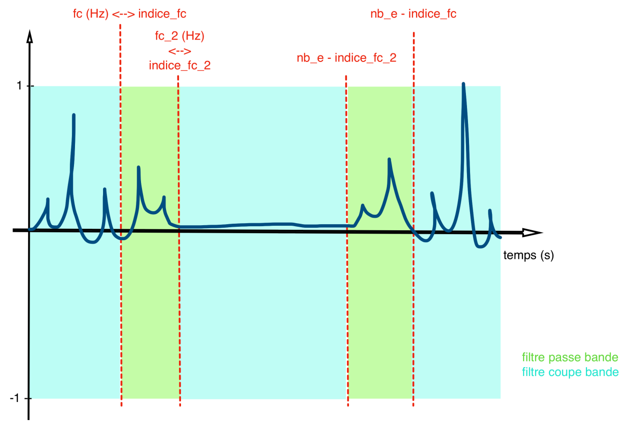


Figure 20: Schéma conceptuel de l'effet engendré par un filtre coupe bande ou un filtre passe bande sur les données de la TFD d'un signal. En bleu: la zone du signal conservée par le filtre coupe bande. En vert: la zone du signal conservée par le filtre passe bande.

8.4 Coupe bande

Le filtre coupe bande correspond à l'inverse du filtre passe bande, sur la Figure 20, on peut voir comment il fonctionne.

8.5 Butterworth

Le filtre de Butterworth est un peu différent des quatre précédents. C'est un filtre dit linéaire qui possède ce qu'on appelle un gain. Ce gain (mesuré en dB) reste relativement constant dans la bande passante du filtre, et tend vers 0 en dehors de cette région. Pour ce projet, on implémente un filtre discret d'ordre 3 et voici son équation sous forme récursive:

$$F(z) = \frac{\alpha^3(z^{-3} + 3z^{-2} + 3z^{-1} + 1)}{A(\alpha) + z^{-1}B(\alpha) + z^{-2}C(\alpha) + z^{-3}D(\alpha)}$$

avec

$$\alpha = \pi \cdot \frac{fc}{fe}$$

9 Filtrer les signaux

On va à présent filtrer nos TFD pour ensuite leur appliquer une TFI et obtenir un signal audio filtré.

9.1 Avec un filtre passe bas

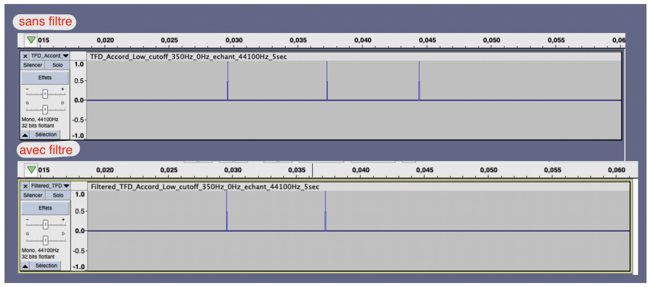


Figure 21: TFD avant et après application d'un filtre passe bas sur le signal d'un accord en Do majeur (fréquence de coupure = 350Hz, durée = 5 secondes, fréquence d'échantillonnage = 44100Hz).

Sur la figure ci-dessus, on constate que les fréquences au delà de 350Hz ont été filtrées. Cela signifie que notre accord ne comporte plus qu'un Do et un Mi une fois filtré et retransformé en signal sonore (le Sol ayant une fréquence de 392Hz dans notre cas).

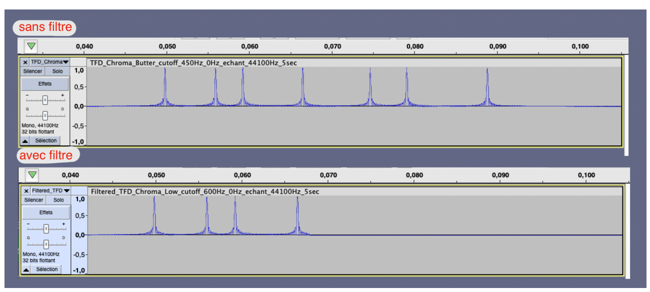


Figure 22: TFD avant et après application d'un filtre passe bas sur le signal de la gamme chromatique en La (fréquence de coupure = 600Hz, durée = 5 secondes, fréquence d'échantillonnage = 44100Hz).

Sur la figure ci-dessus, on comprend que notre gamme a été filtrée d'une façon à ce que les notes après le Ré ne soient pas jouées (en écoutant la TFI de ce signal filtré, on entendra que les 4 premières notes de l'accord).

9.2 Avec un filtre passe haut

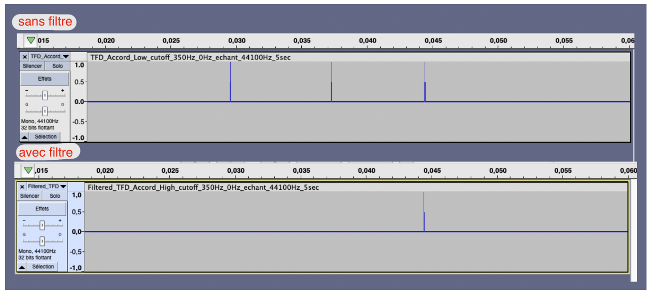


Figure 23: TFD avant et après application d'un filtre passe haut sur le signal d'un accord en Do majeur (fréquence de coupure = 350Hz, durée = 5 secondes, fréquence d'échantillonnage = 44100Hz).

Sur l'image ci-dessus, l'effet est l'inverse de ce qu'on peut observer à la Figure 21

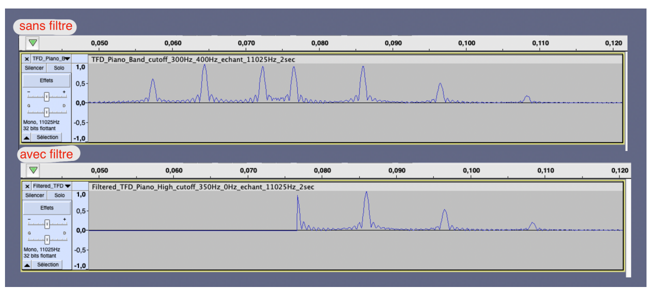


Figure 24: TFD avant et après application d'un filtre passe haut sur le signal de la gamme piano (fréquence de coupure = 350Hz, durée = 2,4 secondes, fréquence d'échantillonnage = 11025Hz).

9.3 Avec un filtre passe bande

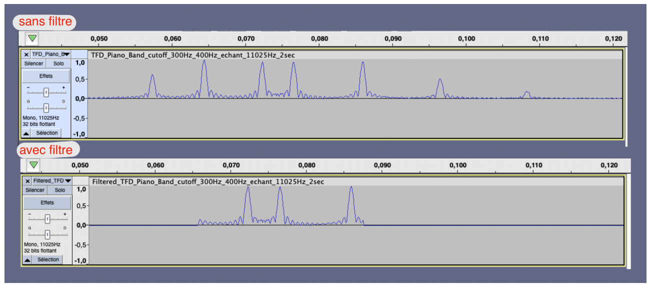


Figure 25: TFD avant et après application d'un filtre passe bande sur le signal de la gamme piano (fréquence de coupure n°1 = 300Hz, fréquence de coupure n°2 = 400Hz, durée = 2.4 secondes, fréquence d'échantillonnage = 11025Hz).

9.4 Avec un filtre coupe bande

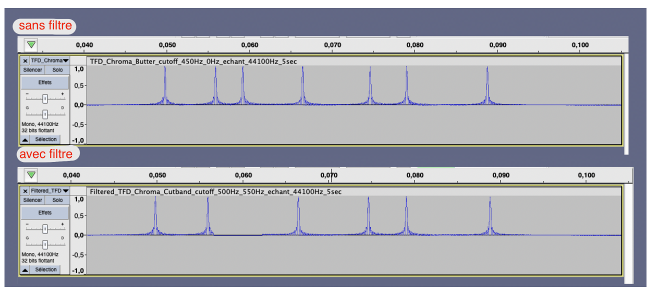


Figure 26: TFD avant et après application d'un filtre coupe bande sur le signal de la gamme chromatique en La (fréquence de coupure n°1 = 500Hz, fréquence de coupure n°2 = 550Hz, durée = 5 secondes, fréquence d'échantillonnage = 44100Hz).

Sur l'image ci-dessus, on constate que seule une note a été supprimée grâce au filtre coupe bande, le Do.

9.5 Avec un filtre de Butterworth

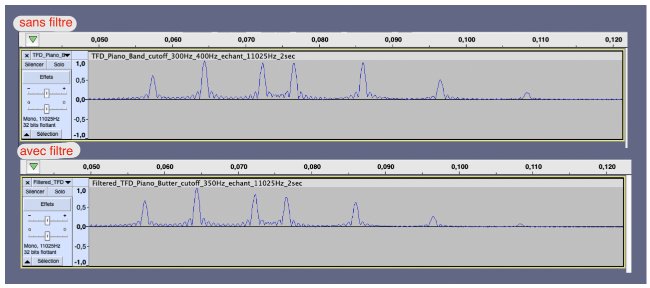


Figure 27: TFD avant et après application d'un filtre de Butterworth sur le signal de la gamme piano (fréquence de coupure = 350Hz, durée = 2.4 secondes, fréquence d'échantillonnage = 11025Hz).

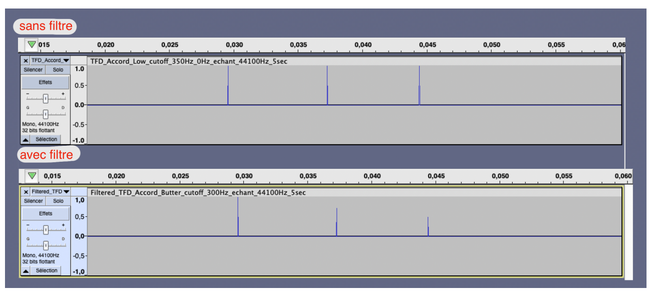


Figure 28: TFD avant et après application d'un filtre de Butterworth sur le signal d'un accord en Do majeur (fréquence de coupure = 300Hz, durée = 5 secondes, fréquence d'échantillonnage = 44100Hz).

Sur les deux exemples ci-dessus, on observe que les pics de fréquences diminuent graduellement une fois la fréquence de coupure atteinte.

References

[1] *3.1 Transformation de Fourier : rappels.* URL: `https://glq2200.clberube.org/chapitres/docs/signal-fourier`.

[2] Futura. *Transformée de Fourier : qu'est-ce que c'est ?* Apr. 2012. URL: `https://www.futura-sciences.com/sciences/definitions/mathematiques-transformee-fourier-11880/`.

[3] Pol Grasland-Mongrain. *La transformée de Fourier - Les Petites Curies du Net.* Feb. 2015. URL: `http://polgm.free.fr/petitescuriesdunet/index.php?post/2015/02/La-transform%C3%A9e-de-Fourier`.

[4] Wikipedia contributors. *Note de musique*. Oct. 2022. URL: https://fr.wikipedia.org/wiki/Note_de_musique.

[5] Wikipedia contributors. *Transformation de Fourier*. Nov. 2022. URL: https://fr.wikipedia.org/wiki/Transformation_de_Fourier.