



PROJET SIGNAL

Accordeur électronique de guitare avec MATLAB

Colin Mourard <colin.mourard@ensica.isae.fr>
Anthony Pagliaï <anthony.pagliai@ensica.isae.fr>

ISAE - Formation ENSICA - Année 2013-2014

Table des matières

II	Analyse de signaux réels	3
1	Introduction	4
2	Elaboration de l'accordeur électronique	5
2.1	Observation des signaux fournis	5
2.2	Nouvelle stratégie de traitement	7
2.2.1	Schéma bloc de la stratégie	7
2.2.2	Développement des 6 filtres passe-bande	7
2.2.3	Détermination de la corde jouée	16
2.2.4	Détermination de l'action à effectuer sur la corde	16
2.3	Acquisition du signal	17
3	Que pourrions-nous améliorer dans notre accordeur ?	18
4	Conclusion	19

Deuxième partie

Analyse de signaux réels

Chapitre 1

Introduction

Au cours de toute la première partie, nous avons travaillé sur l'analyse de signaux synthétisés. L'avantage que présentaient ces signaux résidait dans le fait qu'ils étaient parfaits, c'est à dire que les signaux n'étaient composé que d'une seule harmonique.

Par ailleurs, nous avons tenté à la fin de la première partie de construire des signaux synthétisés qui se rapprochent des signaux réels. En effet, nous avons construit des signaux comportant plusieurs fréquences fondamentales (mais avec différentes amplitudes) ce qui nous avaient permis de soulever une problématique qui est essentielle dans cette deuxième partie, à savoir : Comment extraire la bonne fondamentale correspondant à la fréquence d'une corde accordée ?

Le but de cette partie est donc d'analyser des signaux réels et de proposer un traitement pour accorder les cordes. Dans un premier temps, nous étalonnerons notre accordeur sur des signaux pré-enregistrés. Dans un second temps, nous testerons notre accordeur sur des signaux que nous enregistrerons en direct via un microphone. Le microphone sera de suffisamment bonne qualité pour nous permettre de ne pas mettre en place de traitement spécifique pour le bruit.

Chapitre 2

Elaboration de l'accordeur électronique

Pour cette deuxième partie de projet, nous décidons de créer une toolbox. Ainsi, notre travail sera plus structuré et plus facile d'accès pour un intervenant extérieur. La toolbox comportera deux dossiers, **BIB** contenant les fonctions créées que nous implémentons (par exemple *sav2lin.m*) et **DATA** contenant les signaux que nous utiliserons avec notre accordeur de guitare. De plus, notre projet contient deux M-Files : une fonction *startup.m* qui nous permet d'initialiser notre environnement de travail et le script *accordeur.m* qui constitue le traitement de notre signal à proprement parler.

2.1 Observation des signaux fournis

Nous souhaitons observer temporellement et fréquemment les signaux qui nous sont fournis. Dans un premier temps, il nous faut charger les-dits signaux. C'est pourquoi nous utilisons les fonctions **load** et **uigetfile**. La fonction **load** nous permet de charger un fichier et la fonction **uigetfile** nous ouvre la boîte de dialogue dont on indique le chemin d'accès en paramètre, ce qui permet à l'utilisateur de choisir son signal à charger. Ainsi, nous n'écrivons pas en dur dans notre script quel signal nous chargeons.

Nous avons une grande quantité de signaux à notre disposition. Nous choisissons de ne représenter temporellement et fréquemment que le signal contenu dans le fichier *LA1* correspondant à la note *LA*, 110 Hz, *Correctement Accordée*, à titre d'exemple.

La FIGURE 2.1. présentée ci-dessous représente l'allure temporelle de la note LA1 :

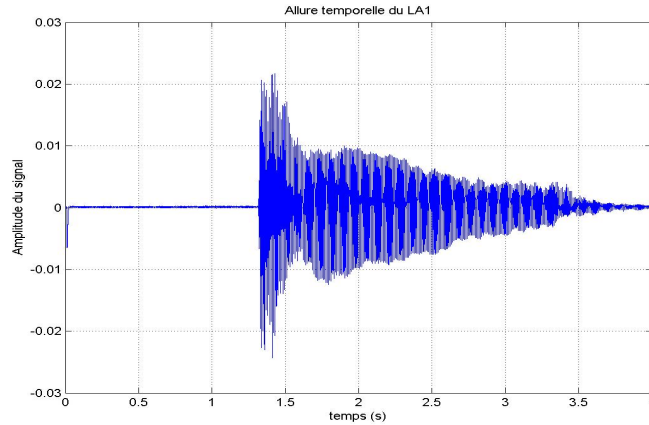


FIGURE 2.1 – Allure temporelle du LA1

Pour la représentation fréquentielle du signal, nous avons calculé la transformée de Fourier du signal LA1 à l'aide de la fonction `fft` et nous l'avons centrée en zéro à l'aide de la fonction `fftshift`. Pour améliorer la lecture, nous avons réduit la fenêtre d'affichage avec la fonction `xlim` en divisant par 10 l'intervalle de fréquence initial du signal chargé. La FIGURE 2.2. représente l'allure fréquentielle du LA1 :

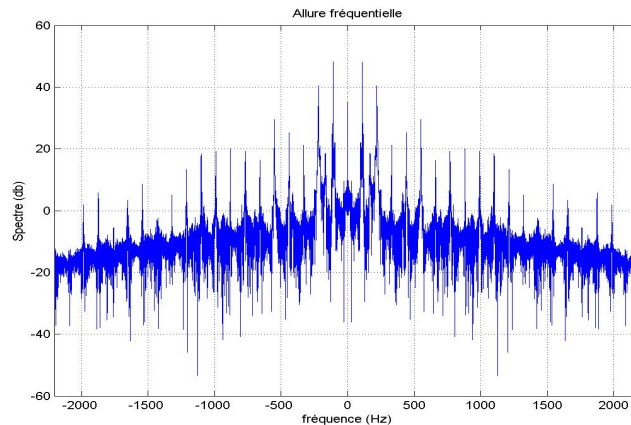


FIGURE 2.2 – Allure fréquentielle du LA1

En regardant de plus près l'allure de la FIGURE 2.2., nous pouvons remarquer que le pic maximum du signal correspond bien, **dans ce cas**, à la fréquence du fondamental. Cependant, notons bien que **ce n'est pas toujours le cas** ! En effet, l'allure fréquentielle du signal MI1 représenté sur la FIGURE 2.3. (page suivante) constitue un parfait contre-exemple puisque la fréquence du fondamental pour la corde associée à MI1 est 82,407 Hz alors que le maximum en sortie de

FFT est compris (lecture graphique sur la FIGURE 2.3.) entre 150 et 200 Hz. Par conséquent, la méthode consistant en une simple recherche par maximum en sortie de FFT n'est pas appropriée, car elle ne donnera pas toujours le résultat escompté.

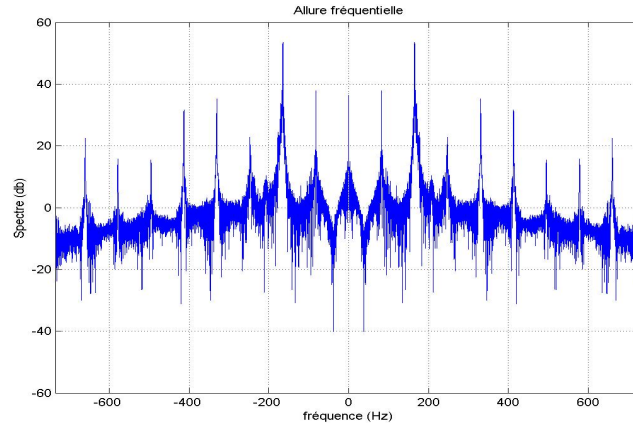


FIGURE 2.3 – Allure fréquentielle du MI1

La suite de cette partie consiste donc à trouver une alternative plus efficace.

2.2 Nouvelle stratégie de traitement

Nous sommes partis de la piste que nous avons entrevue en fin de première partie, à savoir une recherche de l'amplitude maximale du signal. Cependant, comme cette simple solution ne nous a pas apporté satisfaction, nous avons décidé de garder cette idée directrice, mais de l'adapter en conséquence pour obtenir une stratégie de traitement efficace et fonctionnelle.

2.2.1 Schéma bloc de la stratégie

La FIGURE 2.4. que vous trouverez à la page suivante constitue le principe de fonctionnement de notre accordeur électronique de guitare. Par ailleurs, vous trouverez ce schéma dans le dossier `.zip` s'il ne vous paraît pas suffisamment lisible dans ce compte-rendu. Nous ne donnons pas plus de précisions à propos du schéma bloc pour l'instant puisque notre démarche, quoique déjà détaillée sur la FIGURE 2.4., sera expliquée précisément dans la suite de la section 2.2. *Nouvelle stratégie de traitement.*

2.2.2 Développement des 6 filtres passe-bande

Nous avons vu précédemment que les maximum (en amplitude) ne coïncident pas toujours avec le fondamental d'une corde, c'est pourquoi nous choisissons maintenant de ne garder qu'une partie du signal suffisamment resserrée autour du fondamental de chaque corde. Comme la guitare comporte 6 cordes, nous

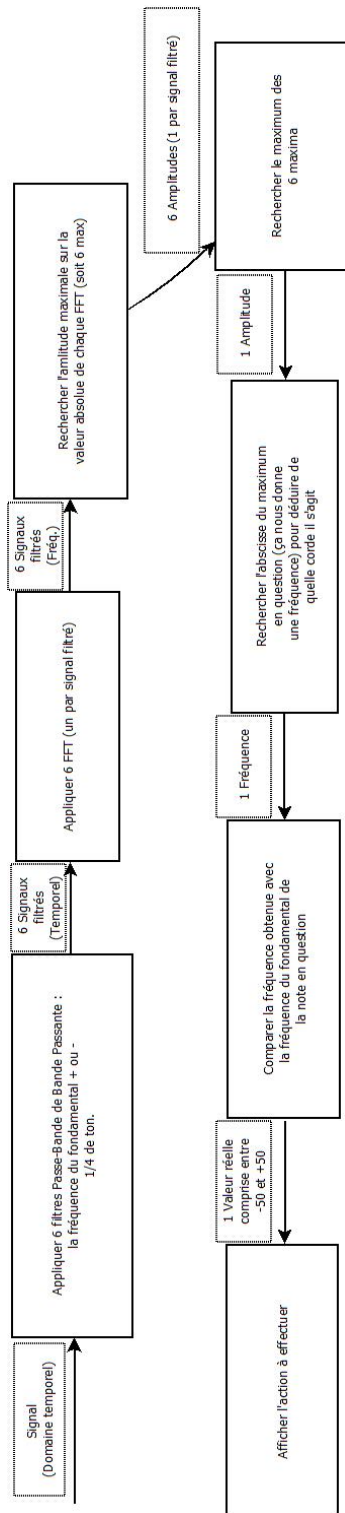




FIGURE 2.4 – Principe de fonctionnement de l'accordeur

avons décidé d'élaborer 6 filtres passe-bande. Chacun de ces filtres est caractérisé par une donnée identique : la bande passante est centrée sur la fréquence du fondamental de la corde, et mesure un demi-ton. En effet, ce choix s'est rapidement imposé à nous étant donné que nous avons fait l'hypothèse (cf Partie 1) que les cordes sont au maximum désaccordées d'un quart de ton.

Pour le choix des filtres, nous avons opté pour des **filtres de Butterworth**. En effet, ces filtres sont utilisés lorsque toutes les fréquences de la bande passante doivent posséder le même gain ; ce  est notre cas. Nous ne souhaitons pas déformer les amplitudes des signaux, sans quoi nous risquons de détecter un mauvais pic par la suite. Par soucis de clarté, nous avons écrit une fonction **filtrage** qui nous permet d'appliquer un filtre de Butterworth d'ordre 4 à un signal (dans le domaine temporel) autour d'une fréquence qui sera successivement, pour notre accordeur, la fréquence du fondamental de chaque corde.

Nous allons maintenant expliciter chacun des filtres. Les filtres de Butterworth sont des filtres à **réponse impulsionnelle infinie (RII)**. Nous avons fait le choix d'utiliser des RII plutôt que des filtres à réponse impulsionnelle finie (RIF) car les RII sont plus sélectifs que les RIF du même ordre. Ainsi, la transition entre la bande passante et la bande rejetée du RII est plus rapide, ce qui nous permet (toujours à même ordre donné) de ne garder que la partie du signal comprise autour du fondamental de chaque corde en éliminant de manière suffisamment efficace le reste du signal.

Ensuite, le choix de l'ordre du filtre est une étape primordiale puisque ce dernier est déterminant dans l'efficacité du filtrage. En effet, la pente de coupure est proportionnelle à l'ordre du filtre (elle vaut *ordre du filtre * 20 dB/dec.*). A priori, plus l'ordre du filtre est important, plus notre filtre sera proche d'une porte, ce qui nous convient puisque nous ne voulons garder qu'une partie du signal lors de chaque filtrage. Cependant, nous devons veiller à ne pas choisir l'ordre du filtre trop important puisque plus on augmente l'ordre du filtre, plus on augmente nos chances d'obtenir un filtre qui sera instable, ce qui n'est pas bon. Finalement, nous avons choisi (par l'expérience) d'utiliser des filtres **d'ordre**  4.

N.B. Comme l'échelle des octaves n'est pas une échelle linéaire, notons que l'écart entre la borne inférieure et le fondamental n'est pas le même que celui entre la borne supérieure et le fondamental. Pour obtenir la bande passante normalisée (i.e. telle que représentée sur les figures grâce à la fonction **fvtool**) il suffit de multiplier cette bande-passante par $\frac{2}{f_s}$, où f_s représente la fréquence d'échantillonnage du signal.

Filtre du MI1

La FIGURE 2.5. représentée ci-dessus donne une allure du filtre associé à la corde MI1 dont la fréquence du fondamental est 82,407 Hz. La bande passante de ce filtre est à $\pm \frac{1}{4}$ de ton par rapport à la fréquence du fondamental. La bande passante est donc le segment de fréquences [80,061 ; 84,822].

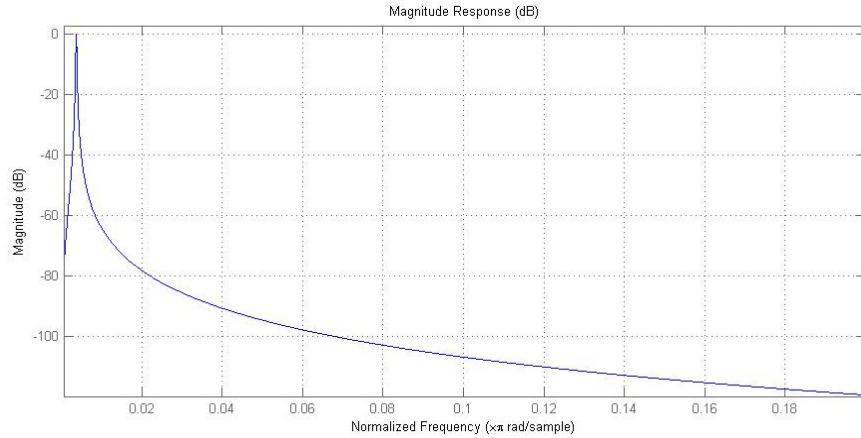


FIGURE 2.5 – Diagramme de Bode (en gain en dB) du filtre du MI1 en fonction de la fréquence normalisée

Comme les filtres passe-bande que nous utilisons sont des RII, il nous incombe d'en vérifier la stabilité. Pour ce faire, nous vérifions que les pôles de la fonction de transfert du filtre a bien tous ses pôles compris dans le cercle unité. Ceux-ci sont représentés sur la FIGURE 2.6. ci-dessous. Pour des soucis de lisibilité, la figure a été zoomée autour des pôles, donc le cercle unité n'est pas entièrement visible. Mais chaque cas, si tous les pôles se trouvent dans la partie concave du cercle unité (ce qui est le cas !) c'est que notre filtre est stable.

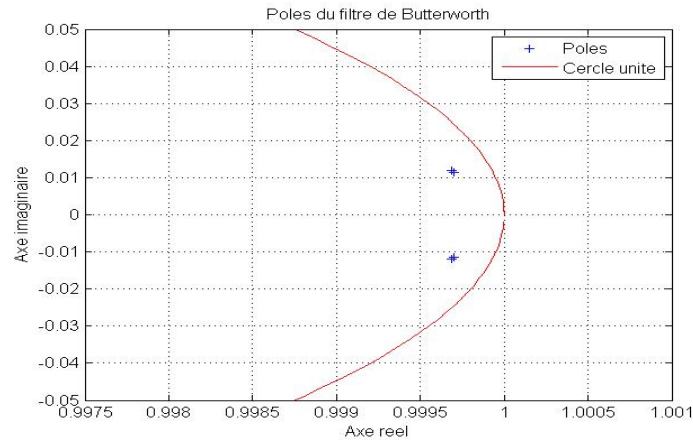


FIGURE 2.6 – Représentation des pôles de la fonction de transfert du filtre du MI1

Filtre du LA1

De la même manière, la FIGURE 2.7. représentée ci-dessus donne une allure du filtre associé à la corde MI1 dont la fréquence du fondamental est 110 Hz. La bande passante est le segment de fréquences $[106,869 ; 113,223]$.

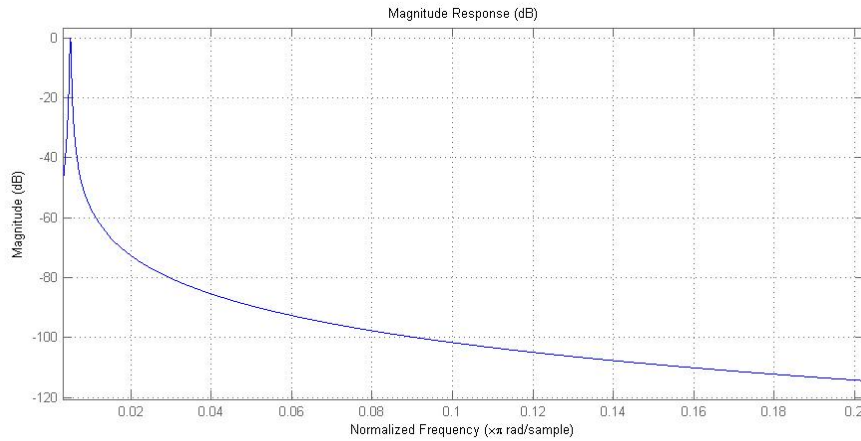


FIGURE 2.7 – Diagramme de Bode (en gain en dB) du filtre du LA1 en fonction de la fréquence normalisée

Nous vérifions que les pôles de la fonction de transfert du filtre a bien tous ses pôles compris dans le cercle unité. Ceux-ci sont représentés sur la FIGURE 2.8. ci-dessous. Tous les pôles se trouvent dans la partie concave du cercle unité, donc notre filtre est stable.

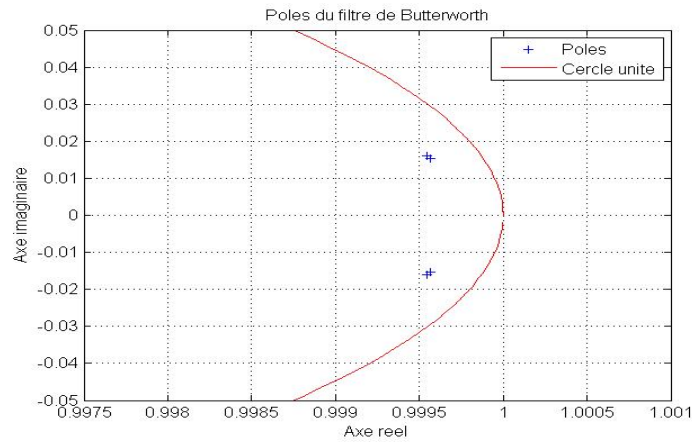


FIGURE 2.8 – Représentation des pôles de la fonction de transfert du filtre du LA1

Filtre du RE2

De la même manière, la FIGURE 2.9. représentée ci-dessus donne une allure du filtre associé à la corde RE2 dont la fréquence du fondamental est 146,83 Hz. La bande passante est le segment de fréquences [142,65 ; 151,13].

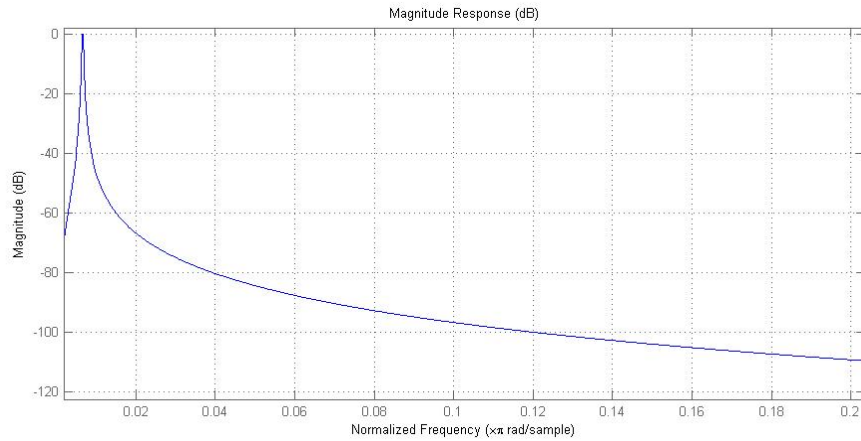


FIGURE 2.9 – Diagramme de Bode (en gain en dB) du filtre du RE2 en fonction de la fréquence normalisée

Nous vérifions que les pôles de la fonction de transfert du filtre a bien tous ses pôles compris dans le cercle unité. Ceux-ci sont représentés sur la FIGURE 2.10. ci-dessous. Tous les pôles se trouvent dans la partie concave du cercle unité, donc notre filtre est stable.

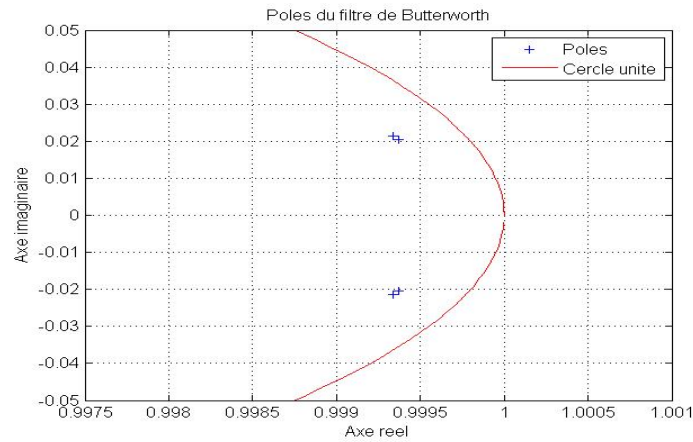


FIGURE 2.10 – Représentation des pôles de la fonction de transfert du filtre du RE2

Filtre du SOL2

De la même manière, la FIGURE 2.11. représentée ci-dessus donne une allure du filtre associé à la corde SOL2 dont la fréquence du fondamental est 196 Hz. La bande passante est le segment de fréquences $[190, 420 ; 201, 743]$.

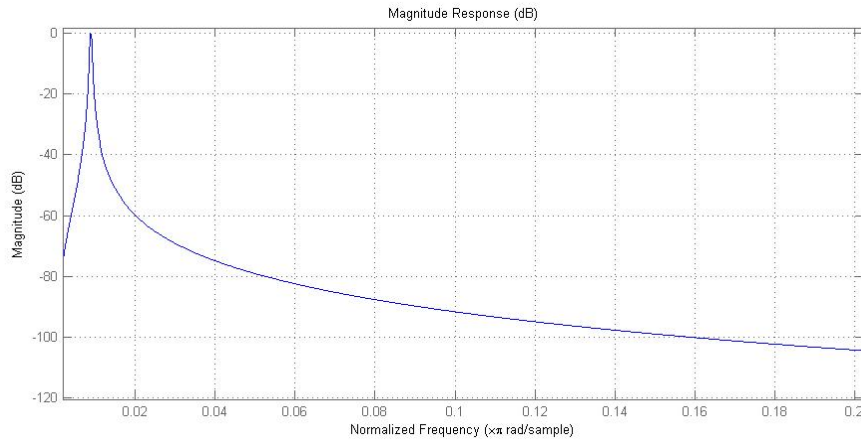


FIGURE 2.11 – Diagramme de Bode (en gain en dB) du filtre du SOL2 en fonction de la fréquence normalisée

Nous vérifions que les pôles de la fonction de transfert du filtre a bien tous ses pôles compris dans le cercle unité. Ceux-ci sont représentés sur la FIGURE 2.12. ci-dessous. Tous les pôles se trouvent dans la partie concave du cercle unité, donc notre filtre est stable.

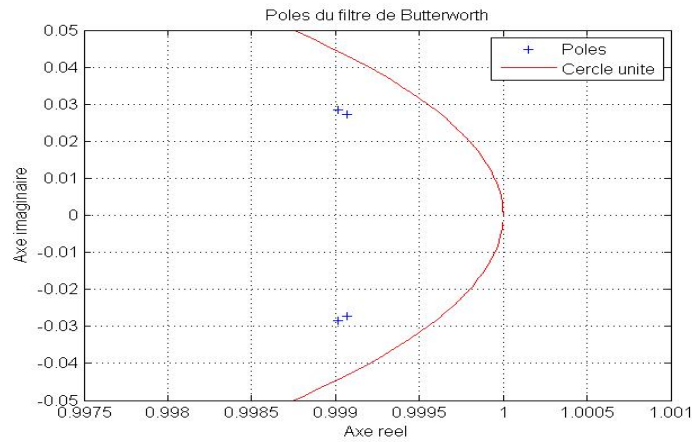


FIGURE 2.12 – Représentation des pôles de la fonction de transfert du filtre du SOL2

Filtre du SI2

De la même manière, la FIGURE 2.13. représentée ci-dessus donne une allure du filtre associé à la corde SI2 dont la fréquence du fondamental est 246,94 Hz. La bande passante est le segment de fréquences [239,91 ; 254,18].

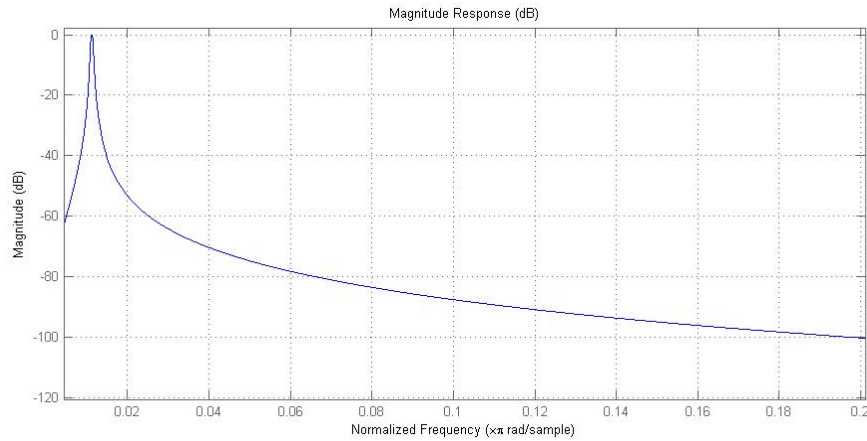


FIGURE 2.13 – Diagramme de Bode (en gain en dB) du filtre du SI2 en fonction de la fréquence normalisée

Nous vérifions que les pôles de la fonction de transfert du filtre a bien tous ses pôles compris dans le cercle unité. Ceux-ci sont représentés sur la FIGURE 2.14. ci-dessous. Tous les pôles se trouvent dans la partie concave du cercle unité, donc notre filtre est stable.

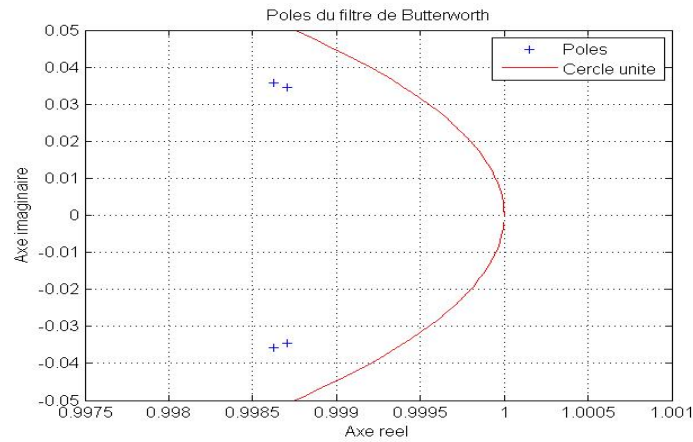


FIGURE 2.14 – Représentation des pôles de la fonction de transfert du filtre du SI2

Filtre du MI3

De la même manière, la FIGURE 2.15. représentée ci-dessus donne une allure du filtre associé à la corde MI3 dont la fréquence du fondamental est 329,63 Hz. La bande passante est le segment de fréquences [320,25 ; 339,29].

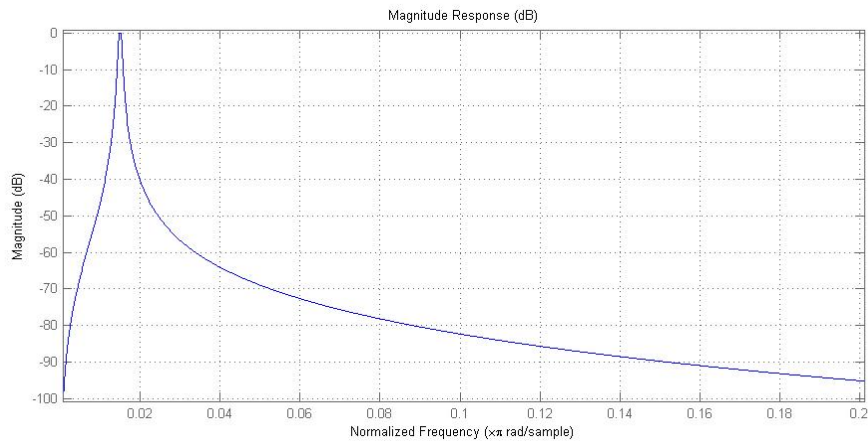


FIGURE 2.15 – Diagramme de Bode (en gain en dB) du filtre du MI3 en fonction de la fréquence normalisée

Nous vérifions que les pôles de la fonction de transfert du filtre a bien tous ses pôles compris dans le cercle unité. Ceux-ci sont représentés sur la FIGURE 2.16. ci-dessous. Tous les pôles se trouvent dans la partie concave du cercle unité, donc notre filtre est stable.

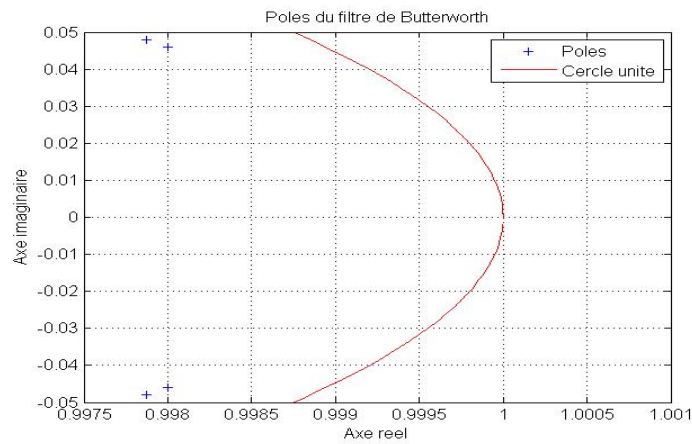


FIGURE 2.16 – Représentation des pôles de la fonction de transfert du filtre du MI3



2.2.3 Détermination de la corde jouée

Une fois notre opération de filtrage terminée, nous obtenons 6 signaux filtrés que nous décidons de traiter dans le domaine fréquentiel. Pour ce faire, nous appliquons une FFT à l'aide de la fonction `fft` comportant un nombre de points égal au nombre de points que comporte le signal temporel discrétisé. Nous décidons donc volontairement de ne pas effectuer de zéro-padding. En effet, un calcul simple de la **précision** (cf Partie 1 pour la formule) nous montre qu'elle est pour notre accordeur **de 0,25 Hz**. Certes cette précision n'est pas suffisante pour obtenir une forme de sinus cardinal en zoomant sur les pics de notre FFT, mais elle est amplement suffisante pour remplir le cahier des charges de l'accordeur puisqu'elle nous permet de bien avoir un écart entre deux points calculés plus petit que l'écart entre deux paliers de l'accordeur (par exemple entre +10 et +20).

Nous utilisons ensuite la fonction `fftshift` pour centrer notre observation fréquentielle, et pour que notre étude soit correcte (nous recherchons un maximum en valeur absolue !) nous n'oublions pas d'appliquer la fonction `abs` qui prend la valeur absolue du signal.

Enfin, nous recherchons sur chaque signal filtré l'amplitude maximum (en valeur absolue) et nous stockons cette valeur ainsi que l'abscisse du maximum (c'est une fréquence !). Cette opération est réalisée grâce à la fonction `max` sur chaque signal filtré. Cela nous donne 6 maxima. Nous ne gardons ensuite que le maximum des maxima, ce qui nous permet de savoir quelle corde est à accorder, ainsi que la fréquence associée à ce maximum.

A ce stade de notre stratégie de traitement, **nous connaissons donc quelle est la corde qu'il faut accorder, et à quelle fréquence est la corde à accorder**. Il nous reste maintenant à comparer cette fréquence à celle du fondamental de la même corde, ce qui nous permettra ensuite de décider de quelle action effectuer sur la corde. La corde en question s'affiche dans la *Command Window*.

2.2.4 Détermination de l'action à effectuer sur la corde

Maintenant que nous connaissons quelle corde il faut accorder ainsi que la fréquence de la note jouée, nous comparons ces deux valeurs. Si la fréquence de la note jouée est supérieure à la fréquence du fondamental, c'est qu'il faut desserrer la corde (l'échelle des paliers +10, +20, ... est définie dans le cahier des charges). Dans le cas contraire, il faut resserrer la corde. Un message s'affiche dans la *Command Window* pour informer l'utilisateur de l'action à effectuer.

Si en revanche la corde est désaccordée de moins d' $\frac{1}{20}$ de ton (valeur imposée par le cahier des charges) nous considérons que notre corde est accordée et un message s'affiche également dans la *Command Window*.

2.3 Acquisition du signal

Pour l'acquisition du signal, nous avons choisi d'intégrer un menu contextuel avec deux choix. Ce menu est très rudimentaire et propose à l'utilisateur soit de choisir un signal déjà enregistré dans la toolbox *DATA* soit d'enregistrer un signal avec un microphone. Avec cette deuxième option, l'utilisateur est informé à l'aide d'instructions qui s'affichent dans la *Command Window* de quand il peut commencer à enregistrer son signal. De plus, pour que l'étude soit correcte, le temps d'enregistrement est tel qu'on a une FFT avec la même précision de 0,25 Hz qu'avec les signaux pré-enregistrés.

Une fois le signal enregistré, le traitement du signal et les étapes qui ont été décrites précédemment sont appliquées scrupuleusement, et de la même manière. L'utilisateur est donc informé de l'action à effectuer, de la même manière qu'avec un son pré-enregistré.

Chapitre 3

Que pourrions-nous améliorer dans notre accordeur ?

Le travail sur notre accordeur électronique de guitare avec MatLab arrive maintenant à son terme, mais ce n'est pas pour autant que c'est un logiciel parfait et il est intéressant selon nous de mettre l'accent sur quelques points que nous aurions éventuellement pu améliorer avec un peu plus de temps.

1. Le traitement d'un signal en temps réel

Pour le confort de l'utilisateur, il aurait pu être intéressant de développer une fonctionnalité permettant de jouer des sons sans avoir à relancer à chaque fois l'accordeur. En effet, au jour d'aujourd'hui, pour accorder notre guitare, nous lançons l'accordeur, nous jouons une note, et il nous affiche à l'écran quelle action effectuer. Et ceci, autant de fois que nécessaire.

2. La polyvalence de notre corde

Notre accordeur a été développé pour accorder une guitare, mais il serait peu adapté pour accorder un autre instrument. Pour intégrer cette nouvelle fonctionnalité (qui n'est pas exigée par le cahier des charges) il faudrait éventuellement paramétrer différemment notre code pour y intégrer un nombre variable de cordes ainsi que des valeurs variables pour le fondamental de chaque corde.

3. La précision de notre accordeur

Pour améliorer le rendu de nos observations graphiques (notamment fréquentielles) nous aurions pu améliorer la précision de la FFT en utilisant la méthode du zéro-padding. Je rappelle que nous ne l'avons volontairement pas fait étant donné que la précision obtenue sans zéro-padding nous permettait de remplir les exigences du cahier des charges.

4. L'interface graphique de notre accordeur

Très rudimentaire, elle est certes efficace, mais manque cruellement de soin esthétique. En effet, nous n'avons pas eu le temps de développer une interface plus aboutie, mais une interface graphique créée avec l'éditeur GUIDE aurait été de toute évidence plus agréable pour l'utilisateur. En effet, cet éditeur est non seulement plus ergonomique et esthétique, mais il permet également d'afficher des graphes en temps réel.

Chapitre 4

Conclusion

Cet algorithme présente l'avantage de déterminer la fréquence de la note jouée, permettant ainsi à l'utilisateur d'accorder sa guitare avec une précision suffisante, tant que les cordes ne sont pas désaccordées de plus d'un quart de ton. Cette hypothèse fondamentale est notamment utilisée lors du filtrage, pour ne garder que la partie intéressante du signal à chaque fois. Sans cette hypothèse, utiliser un filtre ne serait pas une solution technologique envisageable puisqu'alors une trop grande partie du signal ne serait pas coupée, nous confrontant ainsi aux problèmes rencontrés dans la Partie 1 de ce projet.

Cependant, nous sommes bien évidemment conscients que d'autres solutions que celle que nous avons utilisée sont possibles, et tout autant efficace si ce n'est plus, c'est pourquoi nous gardons un oeil critique sur les propriétés de notre accordeur qui est, bien que fonctionnel et abouti, un outil perfectible.