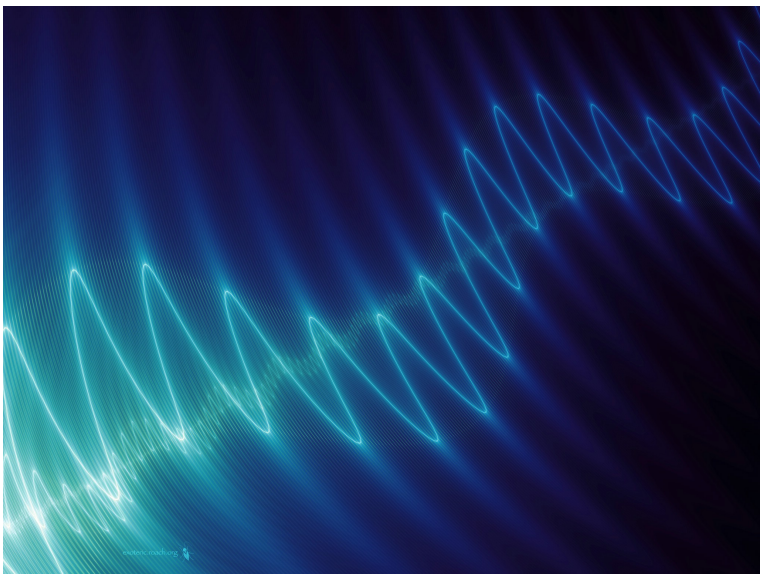
Université de Liège

Travail de fin d'études

Année académique : 2011-2012

Modélisation du comportement non-linéaire d'un haut-parleur pour sonorisation instrumentale



Travail de fin d'études réalisé par Thomas Schmitz , en vue de l'obtention du grade de Master Ingénieur Civil Electricien

Promoteur : Embrechts Jean-Jacques

Faculté des Sciences Appliquées

" *La musique est une mathématique sonore, la mathématique une musique silencieuse.* "

Edouard Herriot

Remerciements :

– JJ.Embrechts pour m'avoir permis de choisir cette thèse qui me tenait à coeur, pour sa disponibilité, son implication dans le projet, ses conseils et son écoute.

– A.Calderon pour le partage de son expérience, sa disponibilité et son aide lors de mes expérimentations au laboratoire d'acoustique.

– J.Osmalsky et Q.Gemine pour leur soutient informatique lors de la création du plugin .

– M.Andrien et B.Schmitz les relecteurs de ce mémoire .

Résumé :

Pour un guitariste, le haut-parleur qu'il utilise fait partie intégrante de sa chaîne d'instrumentation, au même titre que son ampli, ses effets ou sa guitare. Le haut-parleur agit comme un filtre constitué d'un grand nombre de subtilités qu'il est fort difficile de copier même à l'aide d'un égaliseur de précision.

L'objectif de ce travail est de remplacer le haut-parleur électrodynamique et son enceinte par une simulation numérique dans le but d'une utilisation "studio". Une attention particulière sera portée sur le comportement non linéaire du haut-parleur afin d'inclure ses non-linéarités dans le modèle de simulation.

En premier lieu, nous ferons le tour des techniques existantes pour simuler un haut-parleur. En deuxième lieu, nous établirons la théorie nécessaire à notre simulation (hypothèses de modélisation, choix de la méthode de modélisation, limites du modèle). Ensuite, nous prendrons des mesures caractérisant le haut-parleur et vérifierons si le modèle ainsi reconstruit est bien équivalent au baffle réel. Finalement , nous effectuerons quelques tests d'écoute afin de déterminer si la simulation est fiable ou non .

Pour effectuer les tests audio, nous développerons un plugin capable de s'insérer dans un grand nombre de séquenceurs audio. Il nous permettra de simuler le comportement de l'enceinte en temps réel pendant qu'un guitariste joue sur son ampli. Nous aurons aussi besoin d'un système de loadbox permettant d'adapter le niveau du signal disponible en sortie d'un amplificateur guitare vers un niveau ligne capable de rentrer dans la carte son d'un ordinateur.

1 Introduction

1.1 Simulation d'un haut-parleur

Le but de ce travail est de remplacer le haut-parleur électrodynamique d'un guitariste par une simulation numérique.

Les besoins des guitaristes

Une grande part des guitaristes jouent sur des amplis à lampes aux comportements fortement non-linéaires [1] et dont la puissance est prévue pour une salle de concert. Le son ainsi obtenu est fortement dépendant du niveau de puissance fournit par l'ampli. On entend souvent dire que le son des amplis à lampes est meilleur quand ils fonctionnent au-delà de la moitié de leur puissance maximale.

Lors de l'enregistrement d'un album en studio, le guitariste se voit dans l'obligation de jouer avec le même niveau sonore que dans sa salle de répétition afin d'obtenir la même sonorité. Pour un guitariste muni d'un amplificateur 100W branché sur un cab 4*12 [2], la pression acoustique à 1m serait de 123 db_{spl} [3], à 2m la pression serait encore de 117 db_{spl} [4], ce qui reste un niveau peu supportable [5].

L'idée est donc de reprendre le signal en sortie de l'amplificateur, de l'adapter vers un niveau ligne pour pouvoir l'enregistrer à l'aide d'une carte son et d'émuler numériquement la réponse du haut-parleur . Le son serait alors retransmis vers un système hifi (casque ou enceinte) ce qui permet de jouer avec le niveau maximal de l'ampli tout en

1. voir [46]
2. 4 haut-parleurs de 12 pouces et de 8 ohms chacun. Deux couples de haut-parleurs sont mis en parallèles où un couple représente la mise en série de 2 haut-parleurs => impédance totale du dipôle = 8 ohms
3. En se basant sur le rendement du haut-parleur Eminence Wizard qui est de 103db/W/m et en considérant que chaque haut-parleur se répartit une puissance de 25W, alors nous avons 4 haut-parleurs fournissant une pression Lp = 103 +10log(25)=117 db. Nous obtenons donc pour l'ensemble des haut-parleurs un niveau Lp=117+10log(4)=123 db_{spl} (voir hypothèse p10 [17])
4. -6db par doublement de distance (onde sphérique en champ libre voir p16 [17])
5. Le seuil de douleur est à 120db_{spl}

gardant un niveau acoustique agréable.

Les avantages sont les suivants :

- L'ampli peut travailler dans sa plage optimale (qualité du son) tout en gardant un niveau acoustique agréable.
- Le guitariste ne doit transporter que son ampli au studio et non son baffle. Un 4*12 pèse entre 35 et 50 kg et il est souvent très difficile à transporter seul vu ses grandes dimensions (voir figure 1.1) .
- La simulation ne nécessite plus de micro coûteux (à positionner devant le baffle) puisque le signal électrique est repris en sortie de l'ampli.
- L'enregistrement d'une piste peut se faire en appartement sans risque de gêner les voisins(le volume sonore est contrôlé par le plugin).
- La qualité de l'enregistrement est indépendante du bruit ambiant présent dans la pièce. En effet, le signal enregistré est directement repris du signal électrique sortant de l'ampli.
- Le musicien à accès à une grande quantité d'enceintes à moindre coût et ce, indépendamment de la taille de son studio .

Figure 1.1 – Photo d'un cabinet 4*12

Convolution non-linéaire

L'originalité de la méthode présentée dans ce travail est d'avoir modélisé le comportement non-linéaire du haut-parleur à l'aide de la théorie des séries de Volterra [6].

6. voir section 2.4 page 16

Plan du mémoire

1. Le chapitre 2 présente un rappel sur :
   (a) le haut parleur électrodynamique,
   (b) la théorie des systèmes,
   (c) les modélisations de haut-parleurs existantes.
2. Un développement de la théorie nécessaire à la caractérisation des noyaux de Volterra sera présent au chapitre 3 .
3. Le chapitre 4 est consacré à l'analyse de nos différentes mesures.
4. Le chapitre 5 est consacré aux différents tests audio effectués .
5. Le chapitre 6 discute des perspectives de futurs travaux qu'amène le présent TFE.
6. Une conclusion sera finalement donnée au chapitre 7 .
7. Viennent ensuite une série d'annexes :
   (a) Annexe A : présentation du code Matlab.
   (b) Annexe B : présentation du plugin et de ses caractéristiques.
   (c) Annexe C : développement de quelques étapes mathématiques jugées trop "lourdes" pour être insérées dans le corps du texte.
   (d) Annexe D : présentation du matériel utilisé .

2 Etat de l'art

2.1 Le haut-parleur électrodynamique

Le haut-parleur électrodynamique (voir figure 2.1) est un transducteur acoustique permettant de transformer un signal électrique en vibration acoustique.

Le principe de fonctionnement est le suivant : une bobine attachée à une membrane et pouvant se déplacer selon une direction axiale est plongée dans un champ d'induction magnétique radial (provoqué par un aimant permanent). Lorsqu'un signal électrique (provenant d'un amplificateur) traverse la bobine de longueur L, un courant circule dans celle-ci [1]. Il s'ensuit une force de Lorentz $\vec{F} = L\vec{I} \wedge \vec{B}$ entraînant la membrane dans la direction axiale.

2.1.1 Équations de fonctionnement [18],[16]

Le courant circulant dans la bobine induit une force de Lorentz :

$$\vec{F} = L\vec{I} \wedge \vec{B} \tag{2.1}$$

En contre-partie, le déplacement de la bobine dans un champ d'induction magnétique crée une force contre-électromotrice F_{cem} tel que :

$$F_{c.e.m} = B.L.\dot{x} \tag{2.2}$$

Où $\dot{x}$ est la vitesse d'oscillation de la membrane. En modélisant le haut-parleur comme un système mécanique "masse, ressort, amortisseur", l'équation de Newton nous donne pour un régime sinusoïdal :

$$F = B.L.I = jwm'_M\dot{x} + R'_M\dot{x} + K_M\frac{\dot{x}}{jw} \tag{2.3}$$

1. Perpendiculairement au champ d'induction magnétique $\vec{B}$

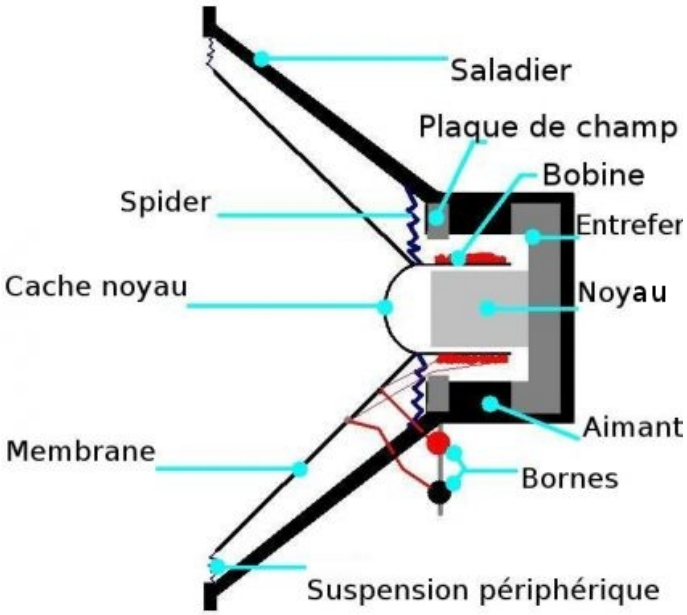


Figure 2.1 – Vue en coupe du haut-parleur électrodynamique

Où :

- w est la pulsation du signal
- m'_M un terme de masse comprenant :
  - la masse de la bobine,
  - la masse de la membrane,
  - la charge d'air déplacé par la membrane (partie imaginaire de l'impédance mécanique. elle est due à la résistance au déplacement de l'air et est modélisée comme un terme de masse voir [16]).
- R'_M est la résistance par frottement incrémentée d'un terme constitué de la partie réelle de l'impédance mécanique due à la résistance au déplacement de l'air.
- K_M est l'inverse de la compliance mécanique, c'est-à-dire la rigidité de l'ensemble.

D'autre part, d'un point de vue électrique, nous savons que la tension aux bornes de la bobine vaut :

$$\begin{aligned} e_g - F_{cem} &= e_g - BL\dot{x} \\ &= (Z_g + R_E + jwL_E)i \\ &= Z_{es}.i \end{aligned} \tag{2.4}$$

Où :

- e_g est la tension envoyée par l'ampli connecté au haut-parleur
- Z_g est son impédance de sortie
- R_E est la résistance électrique de la bobine
- L_E est l'inductance de la bobine
- Z_{es} est l'impédance statique c'est a dire l'impédance composée de Z_g, R_E, L_E .

En tirant i de l'équation (2.4) et en remplaçant dans l'équation (2.3) , nous obtenons la vitesse de vibration de la membrane[18] [16] :

$$\dot{x} = e_g \left(\frac{BL}{Zes}\right) \left[\frac{1}{jwm'_M + (R'_M + \frac{(BL)^2}{Z_{es}}) + \frac{K_M}{jw}}\right] \tag{2.5}$$

Que nous pouvons relier à la pression [2] dans l'axe du haut-parleur pour une distance suffisamment grande [3] par :

$$p(r,t) = j\frac{\rho_0 a^2 w}{2r}\dot{x}e^{j(wt-kr)} \tag{2.6}$$

Où k est le nombre d'onde et ρ_0 est la masse volumique de l'air.

2.1.2 Non-linéarité du haut-parleur

Nous sommes intéressés par les variables de l'équation (2.5) qui peuvent dépendre de la tension e_g, créant ainsi des non-linéarités en fonction de l'amplitude du signal provenant de l'amplificateur.

2.1.2.1 Non-linéarité du champ magnétique B

Le facteur de force BL(x) décrit le couplage entre la partie mécanique et la partie électrique. Ce facteur dépend le la longueur du fil de la bobine qui intercepte le champ d'induction magnétique [37]. Lorsque la bobine avance, la longueur L de la bobine plongée dans le champ d'induction magnétique diminue et par conséquent le facteur de force BL(x) aussi, ce qui a pour conséquences :

1. la force de Lorentz F=BL(x) i dépend de la position de la bobine
2. La force contre électromotrice induite dépend du carré de la position de la bobine $(BL(x)^2)$ [37]

2.1.2.2 Non-linéarité de la rigidité K_M des suspensions

Le haut-parleur utilise des suspensions pour centrer la bobine autour du noyau et générer une force de rappel sur le cône. Cette force de rappel peut s'exprimer comme celle d'un ressort, à l'aide d'un coefficient de raideur K(x) qui dépend de la l'écart entre

2. voir p 41 de [16]
3. la distance $r \gg a$ où a est le rayon de la membrane

la position de la bobine et sa position au repos.

Le coefficient de raideur totale, est une combinaison des coefficients de raideur du spider et des suspensions. Il croit avec l'amplitude, varie avec la fréquence et peut être soumis à des phénomènes d'hystérésis. De plus, il n'est pas symétrique par rapport à la direction du déplacement [37].

2.1.2.3 Non-linéarité de l'inductance de la bobine

L'inductance L_e de la bobine varie avec la position car elle est liée à la reluctance du milieu par $L = \frac{N^2}{\Re}$ (où N est le nombre de spire) . Or la reluctance $\Re$ du milieu n'est pas homogène. En effet, lorsque la bobine recule, elle s'approche du saladier où le milieu est de moindre reluctance (car le saladier est constitué de métal et que $\Re_{metal} < \Re_{air}$ [37]).

2.1.2.4 Non-linéarité de la résistance de la bobine

La résistivité du conducteur de la bobine varie avec la température de celle-ci, or la température de la bobine d'un haut-parleur peut monter à plus de 160°C en fonction du courant qui la traverse. La figure 4.23 du chapitre 4 nous montre la perte de rendement du haut-parleur lorsque nous l'excitons avec un sinus de grande amplitude pendant 25 secondes.

2.2 Modélisations des systèmes

2.2.1 Propriétés des systèmes

Un système est *sans mémoire* si sa sortie pour chaque valeur des variables indépendantes à un moment fixé ne dépend que de l'entrée au même moment [41] .

Un système est dit *causal* si sa sortie ne dépend que des entrées présentes et passées du signal d'entrée [41] .

Un système est *linéaire* s'il vérifie le principe de superposition. C'est-à-dire que si l'entrée est la somme pondérée de deux signaux, la sortie est la même somme pondérée, des deux réponses correspondantes [41],[50].

Un système est donc linéaire si :

- la réponse de $x_1(t) + x_2(t) = y_1(t) + y_2(t)$ (additivité)
- la réponse de ax(t) est ay(t) (homogénéité)

Un système est *invariant* (dans le temps), lorsque la loi qu'il établit entre entrées et sortie ne change pas au cours du temps. Un décalage dans le temps de t secondes du signal d'entrée conduit à un décalage de t secondes du signal de sortie [41],[50].

2.2.2 Produit de convolution

Le produit de convolution peut être vu comme une généralisation de l'idée de la moyenne glissante, la convolution de x(t) par h(t) est définie comme :

$$\int_{-\infty}^{\infty} h(\tau)x(t-\tau)d\tau \tag{2.7}$$

Ou de manière discrète par :

$$\sum_{\tau=-\infty}^{\infty} h[\tau]x[n-\tau] \tag{2.8}$$

Pour un système linéaire et invariant, nous pouvons représenter un signal x[n] par sa convolution avec l'impulsion de dirac tel que :

$$x[n] = \sum_{\tau=-\infty}^{\infty} x[\tau]\delta[n-\tau] \tag{2.9}$$

Le signal x[n] est alors représenté comme une combinaison linaire d'impulsions décalées $\delta[n-\tau]$ associée à un poids $x[\tau]$, où l'impulsion de Dirac est définie comme :

$$\delta[n] = 0, \forall n \neq 0 \text{ et } \delta[n] = 1 \textit{ pour } n = 0. \tag{2.10}$$

En continu, l'impulsion de Dirac est définie par l'approche au limite comme[50] :

$$\delta(t) = 0, \forall t \neq 0 \text{ et } \int_{-\infty}^{+\infty} \delta(t)dt = 1 \tag{2.11}$$

2.2.3 Réponse impulsionnelle

La réponse impulsionnelle d'un système est la sortie obtenue lorsque l'entrée est une impulsion, si l'application qui transforme l'entrée en la sortie est linéaire et invariante alors elle caractérise entièrement le système et y[n] peut être calculée pour n'importe quel entrée x[n] par la relation :

$$y[n] = \sum_{\tau=-\infty}^{\infty} h[n-\tau]x[\tau] = \sum_{\tau=-\infty}^{\infty} x[n-\tau]h[\tau] \tag{2.12}$$

En effet, par (2.9) nous pouvons représenter le signal d'entrée x[n] comme une combinaison linéaire d'impulsion décalée. Si on note $h[n-\tau]$ la réponse du système à l'impulsion $\delta[n-\tau]$ alors par la propriété de superposition du système linéaire, la sortie y[n] n'est autre qu'une combinaison de ces réponses $h[n-\tau]$ associées à un poids $x[\tau]$.

2.3 Obtention de la réponse impulsionnelle

De manière générale la sortie d'un système LTI [4] peut s'écrire comme :

$$y(t) = h(t) \circledast x(t) \tag{2.13}$$

Où $\circledast$ est le symbole du produit de convolution définit par l'équation (2.8).

En remplaçant le signal d'entrée par une impulsion de Dirac, nous obtenons la réponse impulsionnelle du système :

$$y(t) = h(t) \circledast \delta(t) = h(t) \tag{2.14}$$

La difficulté d'obtenir une réponse impulsionnelle de cette manière réside dans la génération d'une impulsion de Dirac parfaite. L'impulsion créée sera déformée par le système qui la génère.

Une des méthode utilisée pour générer une impulsion de Dirac consiste à créer un son bref et de grande intensité tel qu'un coup de pistolet ou un éclatement de ballon. A titre purement informatif, la figure 2.2 nous montre la réponse fréquentielle d'une pièce lors d'un éclatement d'un ballon. La réponse est loin d'être uniforme et nous avons détecté des différences de plus de 7 db entre deux tests successifs.

2.3.1 Déconvolution

La déconvolution est un procédé algorithmique destiné à inverser l'effet d'une convolution. Il permet d'isoler la solution h d'une équation de la forme :

$$y[n] = x[n] \circledast h[n] \tag{2.15}$$

En fréquentiel, la déconvolution consiste à multiplier le signal reçu par le filtre inverse du signal émis de sorte que :

4. Linear Time Invariant

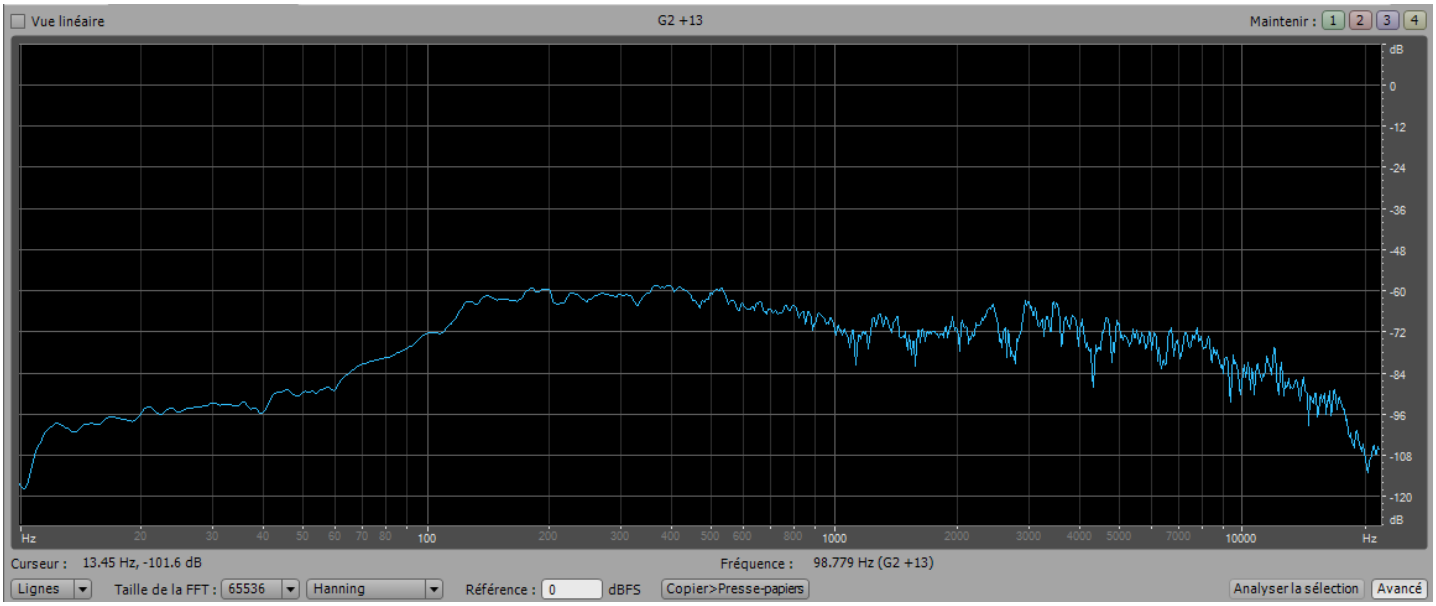


Figure 2.2 – Transformée de Fourier de la réponse impulsionnelle d'un ballon de baudruche explosant dans une pièce (chambre à coucher).

$$Y(jw).\frac{1}{X(jw)} = \frac{X(jw)}{X(jw)}.H(jw) = H(jw) \tag{2.16}$$

Où :

- H(jw) est le spectre de la réponse impulsionnelle du système
- Y(jw) est le spectre du signal de sortie du système
- X(jw) est le spectre du signal d'excitation.
- $\frac{1}{X(jw)}$ noté aussi F(jw) est le filtre inverse de X(jw) tel que F(jw)*X(jw)=1.

Pour obtenir une réponse fréquentielle sur un spectre compris entre 20 hertz et 20 khertz (bande passante des signaux audio), il faut choisir un signal X(jw) qui excite ces différentes fréquences.

Nous avons une certaine liberté face au choix du signal x[n] à utiliser. La seule contrainte étant que le signal utilisé doit exciter toutes les fréquences comprises dans la bande passante que nous cherchons à déterminer.

Les méthodes suivantes sont équivalentes d'un point de vue théorique pour un système linéaire invariant et non bruité. Cependant, chacune d'elles possède ses avantages et ses inconvénients lorsqu'il s'agit de faire apparaître les non linéarités du système ou de diminuer l'influence des bruits parasites [5].

5. Voir aussi les méthodes décrites dans les articles suivant : [31],[15],[23],[22], [14],[52].

2.3.2 Sinesweep

Le sine sweep est un sinus dont la fréquence varie au cours du temps, il peut s'exprimer mathématiquement comme :

$$x(t) = A\ sin(z(t)) \tag{2.17}$$

Dans ce travail, nous utiliserons deux fonctions z(t) différentes pour faire varier la fréquence du sweep :

- l'une fera évoluer la fréquence de manière linéaire avec le temps,
- l'autre fera varier la fréquence de façon exponentielle.

2.3.2.1 Sweep linéaire

C'est un sweep avec une variation linéaire de la fréquence. La fréquence instantanée est donnée par :

$$\frac{dz(t)}{dt} = w_1 + \frac{w_2 - w_1}{T} t \tag{2.18}$$

Où :

- w_1 est la fréquence de départ
- w_2 est la fréquence finale
- T est la durée du sweep.

En intégrant par rapport à t, nous obtenons :

$$z(t) = w_1 * t + \frac{w_2 - w_1}{T} \frac{t^2}{2}$$

2.3.2.2 Sweep logarithmique

C'est un sweep avec une augmentation exponentielle de la fréquence, de manière similaire au cas linéaire, il peut être montré que [21] :

$$x(t) = A\ sin\left[\frac{w_1 T}{ln(\frac{w_2}{w_1})}\ (e^{\frac{t}{T} ln(\frac{w_2}{w_1})} - 1)\right] \tag{2.19}$$

Un des avantages du sweep logarithmique est qu'il regroupe les différentes harmoniques selon leur ordre en des impulsions décalées de Δt par rapport à la réponse principale (voir figure 2.3).

En recherchant quel délai est nécessaire pour que la fréquence instantanée du signal soit un multiple N d'elle même, c'est à dire en cherchant le Δt qui satisfait :

$$N\frac{dz(t)}{dt} = \frac{dz(t+\Delta t)}{dt}$$

Nous obtenons [6] :

$$\Delta t = T\frac{ln(N)}{ln(\frac{w_2}{w_1})} \tag{2.20}$$

Avec :

- Δt le délai entre la réponse impulsionnelle d'ordre 1 et d'ordre N.
- T la durée totale du sweep.
- N l'ordre harmonique recherché.
- w_1 la pulsation initiale.
- w_2 la pulsation finale .

Un exemple d'application de cette formule est donné à la figure 2.3 pour un sweep d'une durée de 20 secondes, balayant les fréquences comprises entre 20 et 20 000 Hz.

Le sweep logarithmique permet donc d'isoler les réponses correspondants aux différents ordres de distorsions.

6. Voir [21] et [7].

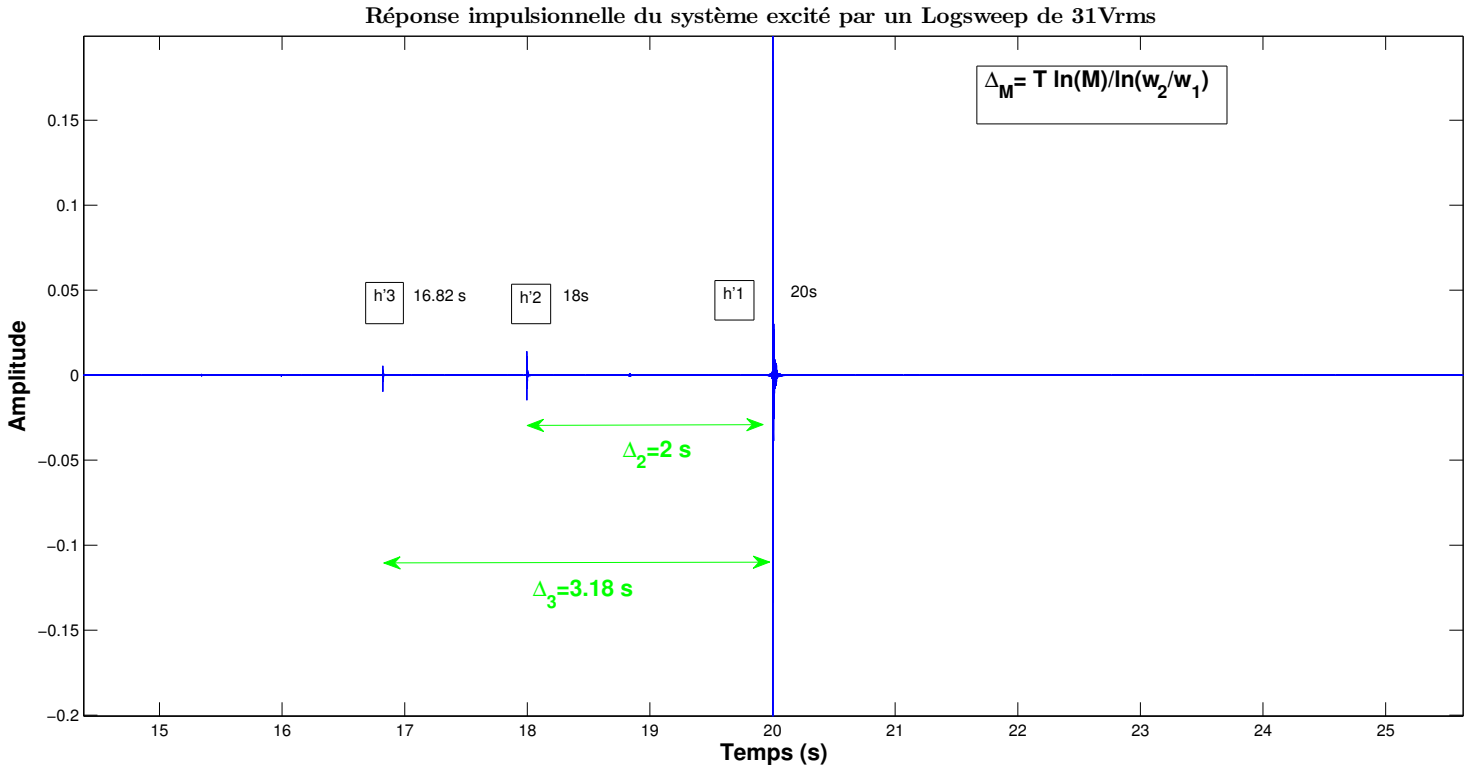


Figure 2.3 – Réponse impulsionnelle du haut parleur obtenue par déconvolution de la réponse du système lorsqu'on lui applique un sweep logarithmique de 20 secondes et application de la formule (2.20)

2.3.3 Maximum length sequence

La technique MLS consiste à récupérer la réponse du système excité par un bruit pseudo-aléatoire. Le bruit est pseudo aléatoire car il est périodique. il peut être généré par une série de registres à décalages avec feedback. La périodicité du signal dépend directement du nombre de registres (n) utilisés et vaut $2^n - 1$.

La réponse impulsionnelle peut être obtenue en calculant la fonction d'inter-corrélation entre les signaux envoyés et reçus. La longueur de la séquence doit être plus grande que la réponse impulsionnelle pour éviter tous problèmes d'aliasing. Des techniques ont été développées pour réduire le temps de calcul causé par la fonction d'inter-corrélation [11], ces techniques sortent du cadre de ce travail et ne seront donc pas décrites plus longuement.

2.4 Système non-linéaire et série de Volterra

Théorie provenant essentiellement de [47].

Un système est dit non-linéaire si sa réponse n'est pas proportionnelle au signal d'en-

trée sur tout son domaine, en pratique l'étude des systèmes non-linéaire est de grand intérêt puisque aucun système physique ne peut être modélisé par un système linéaire sur un domaine du signal d'entrée suffisamment grand.

Si le système est non-linéaire mais invariant dans le temps et possède certaines restrictions [7], nous pouvons montrer que la relation entre l'entrée et la sortie du système peut être exprimée par une série de Volterra tel que :

$$\left\{ y(t) = \sum_{n=1}^{\infty} \int_{-\infty}^{+\infty} ... \int_{-\infty}^{+\infty} h_n(t_1, t_2, ..., t_n) x(t - t_1)...x(t - t_n) dt_1 dt_2 ... dt_n \right. \tag{2.21}$$

Où :

– h_n est le noyau de Volterra d'ordre n

Condition de causalité : Comme tout système physique réalisable, le signal de sortie ne peut dépendre des futures valeurs du signal d'entrée. Par conséquent, les noyaux h_n sont nuls si au moins une des variables $t_1, ..., t_n$ est négative. Le système (2.21) devient :

$$\left\{ y(t) = \sum_{n=1}^{\infty} \int_{0}^{+\infty} ... \int_{0}^{+\infty} h_n(t_1, t_2, ..., t_n) x(t - t_1)...x(t - t_n) dt_1 dt_2 ... dt_n \right. \tag{2.22}$$

Condition de stabilité : La réponse linéaire est BIBO [8] stable si et seulement si

$$\int_{-\infty}^{\infty} |h_1(\tau)| d\tau < \infty \tag{2.23}$$

La condition est nécessaire et suffisante et implique l'existence de la transformée de Fourier de la réponse impulsionnelle h_1 [9] . Les conditions de stabilité des noyaux supérieurs sont discutées dans [47], nous retiendrons que si les noyaux d'ordre supérieur sont bornés au sens de l'équation (2.23) ; alors ils sont stables. C'est une condition suffisante mais non nécessaire.

La série de Volterra est infinie en temps et en ordre, ce qui rend son implémentation pratique difficile. Pour notre modélisation, elle sera tronquée à l'ordre M et à une mémoire de longueur N [10].

On obtient en temps discret :

7. voir section 3.3 page 33
8. Bounded input bounded output
9. voir [47] page 24
10. Il reste à déterminer la valeur de N et M qui donne lieu à une erreur "acceptable".

$$\left\{ y(t) = \sum_{n=1}^{M} \sum_{\tau_1=0}^{N} \dots \sum_{\tau_n=0}^{N} h_n[\tau_1, \dots, \tau_n]\ x[n-\tau_1] \dots x[n-\tau_n] \right. \tag{2.24}$$

2.5 Simulation d'un haut-parleur

Le haut-parleur pour *ampli guitare* est un haut-parleur dont la bande passante est généralement comprise entre 100 et 5000 Hz. Par conséquent, le son enregistré directement en sortie de l'ampli sans passer par le baffle est agressif [11]. L'effet du haut parleur en tant que filtre passe bas est d'autant plus marqué si on ajoute une pédale de distorsion dans la chaîne d'instrumentation du musicien [12].

2.5.1 Le simulateur analogique de haut-parleur

Le simulateur de haut-parleur analogique est composé de filtres du second ordre [13] recréant grossièrement les fréquences de coupures du haut-parleur. Le filtre ne suit pas rigoureusement les subtilités de la réponse en fréquence du haut-parleur, mais reconstitue l'allure générale de sa courbe de réponse.

La figure 2.4 nous montre un exemple de simulateur analogique de haut-parleur. La courbe en trait plein de la figure 2.5 présente la réponse fréquentielle du montage alors que la courbe en pointillé représente la courbe de réponse du haut-parleur que les concepteurs cherchaient à émuler.

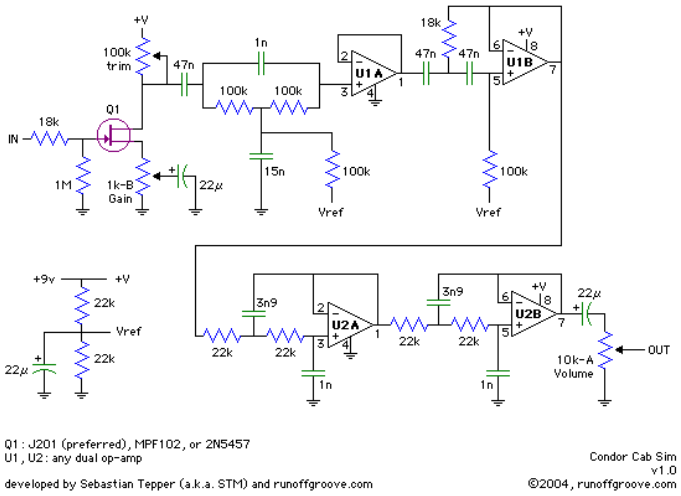


Figure 2.4 – Simulateur de cab analogique [45]

11. Chargé en hautes fréquences
12. La pédale de distorsion ajoutée crée des harmoniques renforçant le haut du spectre du signal.
13. Décroissance du filtre en -6db/octave après sa fréquence de coupure

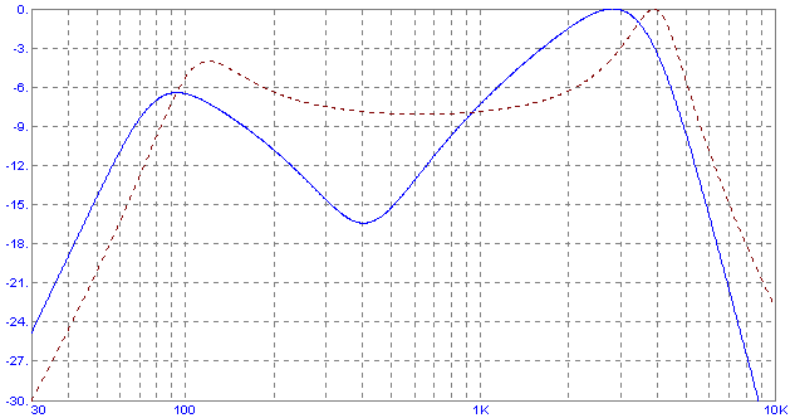


Figure 2.5 – Courbe de réponse fréquentielle du simulateur de cab analogique (trait plein) et courbe de la réponse fréquentielle du cabinet que les concepteurs cherchaient à simuler (pointillé)

2.5.2 Le simulateur numérique à convolution de haut-parleur

La réponse impulsionnelle h(t) d'un système est en quelque sorte l'empreinte du système. En convoluant la réponse impulsionnelle avec le signal d'entrée, x(t), nous obtenons un signal de sortie y(t) qui est tel que serait le signal s'il passait directement dans le système réel.

La précision du système obtenu dépend alors de la qualité de la réponse impulsionnelle et de sa longueur (idéalement, une longueur infinie). La longueur de la réponse impulsionnelle utilisée dépend de la puissance de calcul disponible.

En effet, l'opération de convolution est une opération extrêmement coûteuse en nombre d'opérations. Pour une réponse impulsionnelle d'une longueur = 1s et en travaillant à la fréquence d'échantillonnage fs = 44100 Hz, il faut pour chaque échantillon de sortie, faire 44100 multiplications et 44100 additions. Nous devons donc avoir une puissance de calcul de 44100*(44100+44100)=3.88 Gops (Giga opérations par seconde).

Fort heureusement, la longueur des réponses impulsionnelles mesurée en chambre anéchoïque est de l'ordre de quelques dixièmes de seconde soit 0.35 Gops pour une réponse de 0.3 seconde . Il est donc de notre intérêt de réduire la longueur de la réponse impulsionnelle pour minimiser la charge du *CPU*.

Nos mesures des réponses impulsionnelles seront réalisées en chambre anéchoïque, ce qui permet d'obtenir une réponse courte et indépendante de l'espace acoustique. Un simulateur de réverbération à convolution peut ainsi être ajouté en sortie du système pour choisir l'empreinte acoustique de la pièce désirée.

La méthode de capture de la réponse impulsionnelle sera discutée au chapitre 3 et une étude subjective sur la longueur de la réponse impulsionnelle à utiliser sera menée au chapitre 5.

2.5.3 Le simulateur de haut parleur non-linéaire

Le premier système de convolution non-linéaire fut proposé par la société Syntefex en 2001 avec son *Replicator FX8000*. Le principe [14] est de prendre une série de réponses impulsionnelles à différents niveaux et de choisir la réponse à convoluer avec le signal d'entrée en fonction du niveau de celui-ci.

Le défaut de cette technique est qu'il n'y a pas de continuité entre les réponses impulsionnelles des différents niveaux (bien que l'on puisse interpoler la réponse impulsionnelle entre 2 niveaux).

Une autre société appelée Acoustica, à créé un plugin audio (Nebula3) capable de réaliser des convolutions non-linéaires (sur base d'une série de réponses impulsionnelles de différents niveaux). Comme le montre la figure 2.6, leur plugin permet également la prise de nouvelles réponses en choisissant le pas, en décibel, entre le niveau des réponses impulsionnelles.

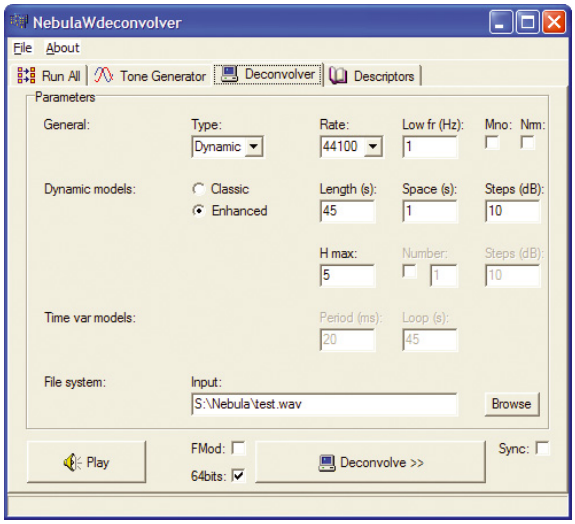


Figure 2.6 – Écran de contrôle pour la prise de nouveaux impulses dynamiques dans le plugin Nebula

Finalement [15], l'équipe de chez *Two-notes* (2008) développe des simulateurs de haut-parleur à convolutions. Bien que leurs méthodes restent inconnues, il n'est pas impossible qu'ils effectuent des convolutions non-linéaires sur base des séries de Volterra, puisque l'idée d'utiliser les séries de Volterra pour simuler le haut parleur électrodynamique me vient d'un de leur membre.

14. Similaire au principe utilisé pour simuler un compresseur dans cet article [32]
15. Voir aussi [54]

3 Modélisation non linéaire du haut parleur, étude théorique.

3.1 Obtention des noyaux de Volterra

Soit le Système de Wiener [1] de la figure 3.1 [2].

Où :

- h_a est la réponse impulsionnelle linéaire du haut-parleur
- h_b la réponse impulsionnelle de la pièce
- le bloc non-linéaire placé entre les deux consistes en une série de puissances sans mémoire.
- x[n] et y[n] sont respectivement l'entrée et la sortie du système
- w[n] est la sortie après la réponse linéaire du haut-parleur
- z[n] est la sortie du haut-parleur non-linéaire

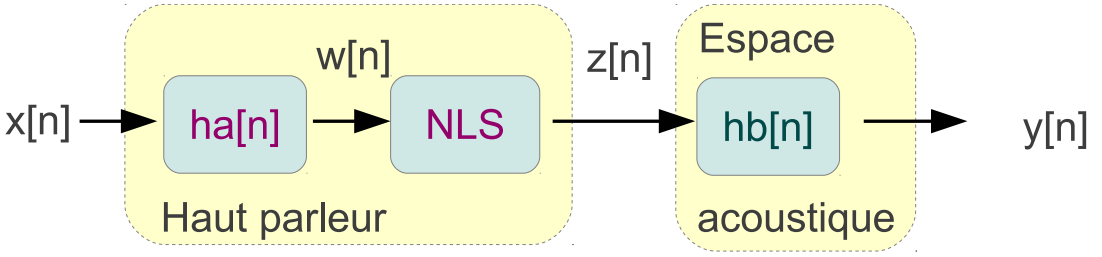


Figure 3.1 – Modèle non-linéaire numéro 1 du haut-parleur et de son espace acoustique.

La sortie y[n] du système complet vaut [3] :

1. La classe de Wiener décrit tous les systèmes physiques invariants en temps et avec mémoire finie.
2. Dans ce modèle, les réponses impulsionnelles du microphone et de l'amplificateur générant le sweep ne sont pas représentées.
3. Voir développement Appendix C section 10.1.1 page 109

$$\begin{cases} y[n] = z[n] \circledast h_b[n] \\ \quad = a_1 h_a[n] \circledast x[n] \circledast h_b[n] \\ \quad + a_2 \sum_{k=0}^{\infty} h_b[k] \sum_{\tau_1=0}^{\infty} h_a[\tau_1] x[n-k-\tau_1] \sum_{\tau_2=0}^{\infty} h_a[\tau_2] x[n-k-\tau_2] \\ \quad + a_3 \sum_{k=0}^{\infty} h_b[k] \sum_{\tau_1=0}^{\infty} \sum_{\tau_2=0}^{\infty} \sum_{\tau_3=0}^{\infty} h_a[\tau_1] h_a[\tau_2] h_a[\tau_3] x[n-k-\tau_1] x[n-k-\tau_2] x[n-k-\tau_3] \end{cases} \tag{3.1}$$

note : nous nous limitons au développement des trois premiers noyaux de Volterra.

Finalement, en posant $\tau_i' = k + \tau_i$ pour le système (3.1), nous pouvons identifier les noyaux h_i de l'équation Volterra 2.24.

Nous obtenons :

$$\begin{cases} h_1[n] = a_1 h_a[n] \circledast h_b[n] \\ h_2[\tau_1', \tau_2'] = a_2 \sum_{k=0}^{\infty} h_b[k] h_a[\tau_1' - k] h_a[\tau_2' - k] \\ h_3[\tau_1', \tau_2', \tau_3'] = a_3 \sum_{k=0}^{\infty} h_b[k] h_a[\tau_1' - k] h_a[\tau_2' - k] h_a[\tau_3' - k] \end{cases} \tag{3.2}$$

Où :

– h_i est le noyau de Volterra d'ordre i.

Le professeur Farina fait l'hypothèse [7] que seule la partie linéaire présente un effet de mémoire et donc que seuls les termes $\tau_1 = \tau_2$ apportent une contribution significative [4] au double produit $h_a[\tau_1] h_a[\tau_2]$, réduisant ainsi les différents noyaux à leur diagonale principale.

Le système se met alors sous la forme de la figure 3.2. Comme nous le verrons à la section 3.2 page 27 , cette hypothèse est peut être trop forte dans certains cas.

Établissons le lien entre la réponse impulsionnelle mesurée (exemple figure 4.13) et les noyaux de Volterra. En temps discret la sortie du système y[n] peut s'écrire de manière générale comme :

4. Un cas particulier est développé en Annexe C section 10.1.2 page 109

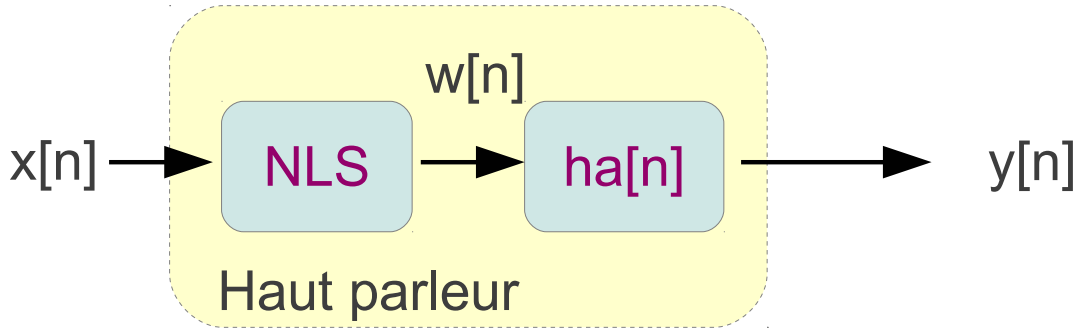


Figure 3.2 – Modèle non-linéaire numéro 2 du haut parleur.

$$\begin{aligned} y[n] = & \sum_{\tau_1=-\infty}^{\infty} h_1[\tau_1]x[n-\tau_1] \\ & + \sum_{\tau_1=-\infty}^{\infty}\sum_{\tau_2=-\infty}^{\infty} h_2[\tau_1,\tau_2]x[n-\tau_1]x[n-\tau_2] \\ & + \sum_{\tau_1=-\infty}^{\infty}\sum_{\tau_2=-\infty}^{\infty}\sum_{\tau_3=-\infty}^{\infty} h_3[\tau_1,\tau_2,\tau_3]x[n-\tau_1]x[n-\tau_2]x[n-\tau_3] \\ & + \ldots \end{aligned} \tag{3.3}$$

Si nous considérons la simplification apportée par le modèle 2 (voir figure 3.2), le système devient :

$$\begin{aligned} y[n] = & \sum_{\tau_1=-\infty}^{\infty} h_1[\tau_1]x[n-\tau_1] \\ & + \sum_{\tau_2=-\infty}^{\infty} h_2[\tau_2]x^2[n-\tau_2] \\ & + \sum_{\tau_3=-\infty}^{\infty} h_3[\tau_3]x^3[n-\tau_3] \\ & + \ldots \end{aligned} \tag{3.4}$$

Qui peut aussi s'écrire comme :

$$y[n] = h1[n] \circledast x[n] + h2[n] \circledast x^2[n] + h3[n] \circledast x^3[n] + \ldots \tag{3.5}$$

La réponse impulsionnelle du système peut alors s'obtenir en remplaçant le signal d'entrée x[n] par un sweep logarithmique puis en convoluant le signal de sortie y[n] avec son filtre inverse f[n] tel que $x[n] \circledast f[n] \Rightarrow \delta[n]$.

Afin de pouvoir convoluer les puissances de x[n] avec le filtre inverse, exprimons celle ci comme une somme pondérée de x[n].

Rappelons qu'en temps discret, le sweep logarithmique x[n] peut s'exprimer comme :

$$\begin{aligned} x[n] =& A\ sin(\phi[n]) \\ =& A\ sin\left[\frac{w_1 T}{ln(\frac{w_2}{w_1})}\ (e^{\frac{n}{N} ln(\frac{w_2}{w_1})} - 1)\right] \end{aligned} \tag{3.6}$$

Avec :

- n est l'échantillon courant
- N est le nombre de samples total.

Nous pouvons alors réécrire les puissances de x[n] [5] comme [6] :

$$\begin{cases} x^2[n] = \frac{A^2}{2}\left[1 + \cos(B)\sin(\phi[n+\Delta_2]) \circledast h_H[n] - \sin(B)\sin(\phi[n+\Delta_2])\right] \\ x^3[n] = \frac{A^3}{4}[3sin(\phi[n]) - \cos(2B)\sin(\phi[n+\Delta 3]) - sin(2B)\sin(\phi[n+\Delta_3]) \circledast h_H[n]] \\ x^4[n] = \frac{A^4}{8}[3 + 4cos(B)sin(\phi[n+\Delta_2]) \circledast h_H[n] - 4sin(B)sin(\phi[n+\Delta_2]) \\ \qquad - cos(3B)sin(\phi[n+\Delta_4]) \circledast h_H[n] + sin(3B)sin(\phi[n+\Delta_4])] \\ x^5[n] = \frac{A^5}{16}[10sin(\phi[n]) - 5cos(2B)sin(\phi[n+\Delta_3]) - 5sin(2B)sin(\phi[n+\Delta_3]) \circledast h_H[n] \\ \qquad + cos(4B)sin(\phi[n+\Delta_5]) + sin(4B)cos(\phi[n+\Delta_5]) \circledast h_H[n]] \end{cases} \tag{3.7}$$

Où :

$$B = \frac{T w_1}{ln(\frac{w_2}{w_1})} \tag{3.8}$$

$$\Delta_M = N\frac{ln(M)}{ln(\frac{w_2}{w_1})}\ , avec\ M \in \aleph \tag{3.9}$$

5. En nous limitant à la 5ème puissance.
6. Voir développement Appendix C : section 10.1.3 page 110

En remplaçant dans (3.5) et en convoluant le résultat par le filtre inverse f[n], nous obtenons :

$$y[n] \circledast f[n] = h_1[n] \circledast x[n] \circledast f[n] + h_2[n] \circledast x^2[n] \circledast f[n] + h_3[n] \circledast x^3[n] \circledast f[n] + ... \quad (3.10)$$

Soit [7] [8] :

$$\begin{cases} y[n] \circledast f[n] = h[n] \\ \qquad = h_1[n] \\ \qquad + \frac{A}{2} h_2[n] \circledast [A \circledast f[n] + cos(B) h_H[n + \Delta_2] - sin(B)\delta[n + \Delta_2]] \\ \qquad + \frac{A^2}{4} h_3[n] \circledast [3\delta[n] - cos(2B)\delta[n + \Delta_3] - sin(2B) h_H[n + \Delta_3]] \\ \qquad + \frac{A^3}{8} h_4[n] \circledast [3A \circledast f[n] + 4cos(B) h_H[n + \Delta_2] - 4sin(B)\delta[n + \Delta_2] \\ \qquad\qquad - cos(3B) h_H[n + \Delta_4] + sin(3B)\delta[n + \Delta_4]] \\ \qquad + \frac{A^4}{16} h_5[n] \circledast [10\delta[n] - 5cos(2B)\delta[n + \Delta_3] - 5sin(2B)\delta[n + \Delta_3] \\ \qquad\qquad + cos(4B)\delta[n + \Delta_5] + sin(4B)\delta[n + \Delta_4]] \end{cases} \quad (3.11)$$

Remarquons que le terme :

$$1 \circledast f[n] = \sum_{k=-\infty}^{\infty} f[k] 1[n - k] = \sum_{k=-\infty}^{\infty} f[k]$$

est en fait la composante continue du filtre inverse f[n], que l'on peut raisonnablement supposer nulle. Il en va de même pour le terme $3 \circledast f[n]$.

Comme le montre l'équation (3.11), la mesure via *LogSweep* permet de décaler les différents ordres de distorsion par rapport à la réponse principale $h_1[n]$. Les termes débutant en $n = -\Delta_i$ ou présentant une contribution prépondérante autour de $n = -\Delta_i$ seront identifiés à la mesure $h'i[n + \Delta_i]$.

En regroupant ensemble les termes décalés d'un même Δ_i de l'équation (3.11), nous obtenons :

7. Les 2 premiers termes sont développés dans l'Appendix C section 10.1.4 page 113

8. Par rapport à l'équation 8 de l'article de Torras-Rosell [53],il y a une différence de signe sur le sin(2B) dans le terme d'ordre 3

$$\begin{cases} h'_1[n] = h_1[n] + \dfrac{3A^2}{4}h_3[n] + \dfrac{5A^4}{8}h_5[n] \\ h'_2[n+\Delta_2] = \dfrac{A}{2}[cos(B)h_2[n] \circledast h_H[n+\Delta_2] - sin(B)h_2[n+\Delta_2]] \\ \qquad + \dfrac{A^3}{2}[cos(B)h_4[n] \circledast h_H[n+\Delta_2] - sin(B)h_4[n+\Delta_2]] \\ h'_3[n+\Delta_3] = \dfrac{-A^2}{4}[cos(2B)h_3[n+\Delta_3] + sin(2B)h_3[n] \circledast h_H[n+\Delta_3]] \\ \qquad - \dfrac{5A^4}{16}[cos(2B)h_5[n+\Delta_3] + sin(2B)h_5[n] \circledast h_H[n+\Delta_3]] \\ h'_4[n+\Delta_4] = \dfrac{A^3}{8}[-cos(3B)h_4[n] \circledast h_H[n+\Delta_4] - sin(3B)h_4[n+\Delta_4]] \\ h'_5[n+\Delta_5] = \dfrac{A^4}{16}[cos(4B)h_5[n+\Delta_5] + sin(4B)h_5[n] \circledast h_H[n+\Delta_5]] \end{cases} \tag{3.12}$$

Comme le montre le système d'équation (3.12) la réponse h'_1 mesurée ne contient pas que la réponse linéaire mais contient aussi une part de ses harmoniques impaires. Les répliques aperçues le long de la réponse impulsionnelle [9] sont donc une superposition des différents noyaux de Volterra.

Finalement en inversant les relations (3.12) et en passant dans le domaine fréquentiel, nous obtenons la transformée de Fourier des différents noyaux de Volterra [10] :

$$\begin{cases} H_1(w) = H'_1(w) + 3e^{2jB}H'_3(w) + 5e^{4jB}H'_5(w) \\ H_2(w) = \dfrac{2}{A}\, je^{jB}[H'_2(w) + 4e^{2jB}H'_4(w)] \\ H_3(w) = \dfrac{-4}{A^2}\, e^{2jB}[H'_3(w) + 5e^{2jB}H'_5(w)] \\ H_4(w) = \dfrac{8}{A^3}\left(-je^{3jB}\right) H'_4(w) \\ H_5(w) = \dfrac{16}{A^4}H'_5(w)e^{4jB} \end{cases} \tag{3.13}$$

9. Voir figure 2.3 page 16.
10. Développement dans l'Appendix C section 10.1.5 114

Les relations (3.13) permettent de trouver les transformées de Fourier des noyaux de Volterra pour w>0. Pour les fréquences négatives, deux options sont possibles :

- Soit remarquer que si $h_i[n] \in \Re \Rightarrow H_i(-w) = H_i^*(w)$
- Soit dériver de nouvelles relations de (10.14) avec $H_H(w) = j$

On remarque également que les relations (3.13) diffèrent de celles proposées par le professeur Farina. [11]

3.2 Artefact dans la réponse impulsionnelle mesurée

Dans la section précédente, nous avons considéré que l'effet de mémoire n'était présent que dans la partie linéaire du système, réduisant ainsi les matrices des noyaux de Volterra à leur diagonale principale. En procédant de cette manière, nous nous sommes aperçu que certains artéfacts qui apparaissaient sur la réponse impulsionnelle (voir figure 4.13) restaient inexpliqués par la théorie développée jusqu'ici.

3.2.1 Cas général modèle Torras-Rossel et Jacobsen [53]

En repartant du modèle 1 (figure 3.1) et de l'équation (10.3) tout en considérant que l'espace acoustique (chambre anéchoïque) n'influence pas la mesure, il peut être montré que les différents noyaux se distribuent le long de la réponse impulsionnelle [53] de façon plus complexe.

3.2.2 Position des répliques additionnelles

Si l'on reprend la théorie générale de l'article de Torras et Jacobsen, des contributions γ_{20},γ_{22} ,γ_{31} et γ_{33} [12] dues à la distorsion sont également générées à des instants $\Delta_{20}, \Delta_{22}, \Delta_{31}$ et Δ_{33}.

Pour que ces contributions soient situées à des instants précis non dilués dans le temps, il faut que γ_{20} soit de la forme [13] :

$$\gamma_{20} = -\tau_1 + \Delta_{20} \tag{3.14}$$

Avec Δ_{20} un délai constant.

Or si $\tau_2 = \tau_1 + \Delta$ avec $\Delta > 0$, alors :

11. eq 10 de [7]
12. Voir éq 5 et 9 de [53].
13. Pour que les Δ_{20} et Δ_{22} de l'équation 8 de l'article [53] mènent à une convolution de h_2 et de $\delta(n - \Delta_{20})$ ce qui donnera lieu à une réplique $h_2[n - \Delta_{20}]$ (même raisonnement pour Δ_{22})

$$\begin{cases} \gamma_{20} = N \dfrac{ln\left[\left(\frac{w_2}{w_1}\right)^{\frac{-\tau_1}{N}} - \left(\frac{w_2}{w_1}\right)^{\frac{-\tau_2}{N}}\right]}{ln\left(\frac{w_2}{w_1}\right)} = N \dfrac{ln\left[\left(\frac{w_2}{w_1}\right)^{\frac{-\tau_1}{N}} - \left(\frac{w_2}{w_1}\right)^{\frac{-\tau_1-\Delta}{N}}\right]}{ln\left(\frac{w_2}{w_1}\right)} \\ = N \dfrac{ln\left[\left(\frac{w_2}{w_1}\right)^{\frac{-\tau_1}{N}}\left(1 - \left(\frac{w_2}{w_1}\right)\right)^{\frac{-\Delta}{N}}\right]}{ln\left(\frac{w_2}{w_1}\right)} = N \dfrac{ln\left(\frac{w_2}{w_1}\right)^{\frac{-\tau_1}{N}} + ln\left[1 - \left(\frac{w_2}{w_1}\right)^{\frac{-\Delta}{N}}\right]}{ln\left(\frac{w_2}{w_1}\right)} \\ = -\tau_1 + N \dfrac{ln\left[1 - \left(\frac{w_2}{w_1}\right)^{\frac{-\Delta}{N}}\right]}{ln\left(\frac{w_2}{w_1}\right)} \end{cases} \tag{3.15}$$

Par identification du deuxième terme de l'équation (3.15) avec l'équation (3.14), nous obtenons :

$$\Delta_{20} = N \dfrac{ln\left[1 - \left(\frac{w_2}{w_1}\right)^{\frac{-\Delta}{N}}\right]}{ln\left(\frac{w_2}{w_1}\right)} \tag{3.16}$$

Si cette hypothèse est vraie, alors le même délai Δ conduit à un délai supplémentaire dans γ_{22} :

$$\begin{cases} \gamma_{22} = N \dfrac{ln\left[\left(\frac{w_2}{w_1}\right)^{\frac{-\tau_1}{N}} + \left(\frac{w_2}{w_1}\right)^{\frac{-\tau_1-\Delta}{N}}\right]}{ln\left(\frac{w_2}{w_1}\right)} \\ = -\tau_1 + N \dfrac{ln\left[1 + \left(\frac{w_2}{w_1}\right)^{\frac{-\Delta}{N}}\right]}{ln\left(\frac{w_2}{w_1}\right)} \end{cases} \tag{3.17}$$

De sorte que :

$$\Delta_{22} = N \dfrac{ln\left[1 + \left(\frac{w_2}{w_1}\right)^{\frac{-\Delta}{N}}\right]}{ln\left(\frac{w_2}{w_1}\right)} \tag{3.18}$$

Remarque :

- Δ_{20} est négatif [14] par conséquent la réplique se situera **après** la réponse linéaire
- Δ_{22} est positif [15] et plus petit que Δ_2, la réplique se situera **entre** *la réplique d'ordre 2* et la réponse linéaire.

3.2.3 Application des résultats de la section 3.2 à un cas concret

La figure 3.3 présente la réponse impulsionnelle du haut parleur éminence wizard 16 [1], obtenue en excitant le système à l'aide d'un logsweep de 20 secondes balayant les fréquences comprises entre 20 et 20 000 Hz. Nous y observons plusieurs répliques, tout d'abords les répliques principales prédites par la formule :

$$\Delta_M = N\frac{ln(M)}{ln\left(\frac{w_2}{w_1}\right)}$$

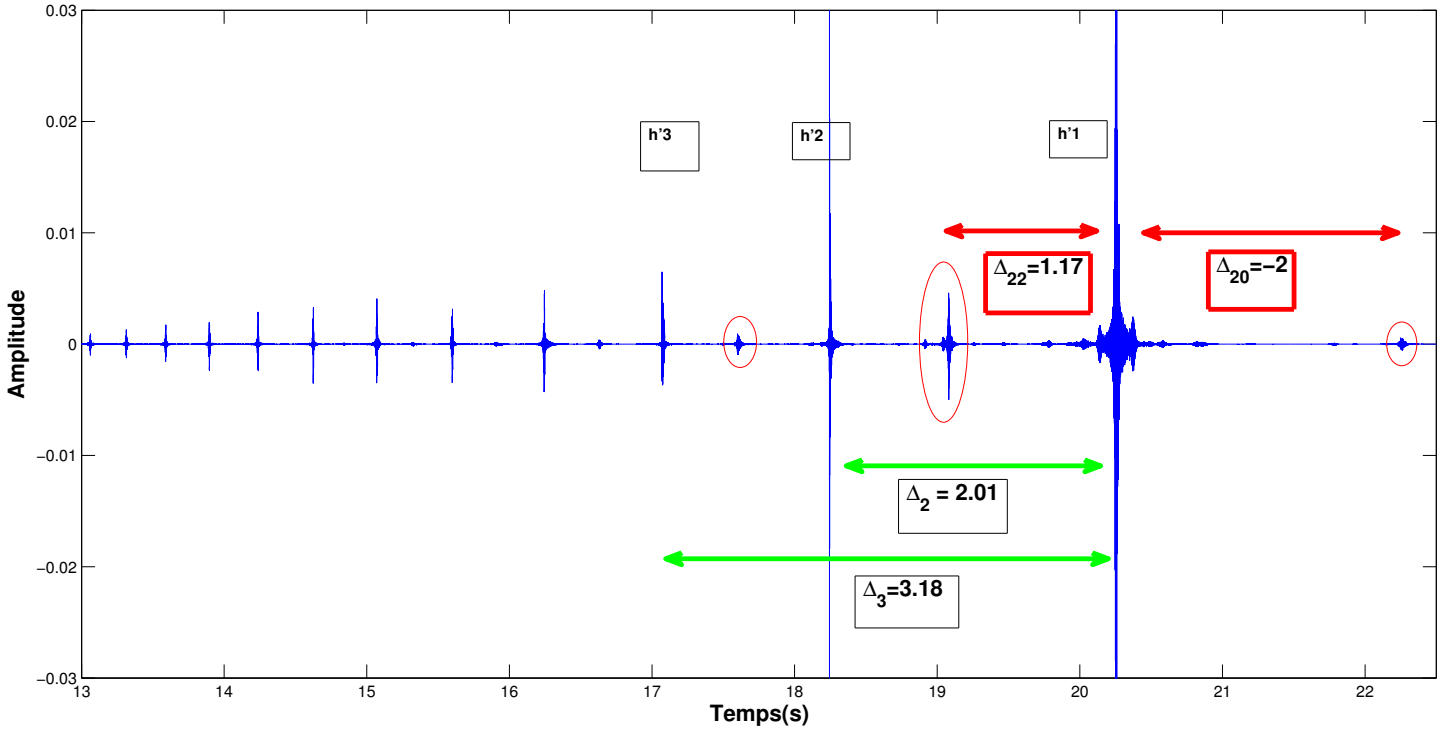


Figure 3.3 – Réponse impulsionnelle du haut-parleur obtenue en déconvoluant le signal de sortie du système lorsqu'on lui applique un sweep logarithmique à son entrée.

14. l'argument du logarithme est < 1
15. $1 < 1 + \left(\frac{w_2}{w_1}^{\frac{-\Delta}{N}}\right) < 2$

– la réponse dite *linéaire* ou d'ordre 1, en t = 20.25 que l'on notera h'_1 ou hp_1

– la réplique d'ordre 2,h'_2, en t =18.24s, $\Delta_2 = 2.01 = T\frac{ln(2)}{ln(\frac{20000}{20})}$.

– la réplique d'ordre 3,h'_3, en t =17.07s,$\Delta_3 = 3.18 = T\frac{ln(3)}{ln(\frac{20000}{20})}$.

– ...

Mais aussi une réplique en t=19.08s. Si cette réplique est celle de Δ_{22}, alors :

$$\Delta_{22} = (20.25 - 19.08) * fs = 51597$$

Où fs est la fréquence d'échantillonnage et vaut 44100Hz.

En remplaçant dans l'équation (3.18) on trouve que :

$$\frac{\Delta}{N} = -\frac{ln\left(\left(\frac{w_2}{w_1}\right)^{\frac{\Delta_{22}}{N}} - 1\right)}{ln(\frac{w_2}{w_1})}$$

Cette valeur de Δ implique une autre réplique en Δ_{20} tel que en remplaçant la valeur trouvée de $\frac{\Delta}{N}$ dans l'équation (3.16), nous trouvons :

$$\Delta_{20} = N\ \frac{ln\left[1 - \left(\frac{w_2}{w_1}\right)^{-0.101}\right]}{ln\left(\frac{w_2}{w_1}\right)} = -87926 \tag{3.19}$$

Une réplique devrait donc exister en :

$$t = 20.25 - (-\frac{87926}{44100}) = 22.25s$$

Ce qui est confirmé par la mesure de la réponse impulsionnelle présentée à la figure 3.3 .

Ces répliques, proviennent de l'équation 8 de l'article [53] dans le cas ou τ_2 est lié à τ_1 par un délai constant Δ. Nous pouvons donc dire [16] qu'il y a un délai Δ autre que $\Delta = 0$ qui donne lieu à un maxima dans la fonction d'auto-corrélation de la réponse linéaire du système.

La figure 3.4 présente la fonction d'auto-corrélation de la réponse linéaire du haut parleur. Cette fonction présente un maximum pour un $\Delta = 88748$ soit un $\frac{\Delta}{N} = \frac{\Delta}{20*44100} = 0.1003$. D'après les formules (3.16) et (3.18), nous devrions obtenir des répliques en :

– Δ_{20}=-88258 soit -2.001 seconde si la fréquence d'échantillonnage vaut 44100 Hz

16. Voir eq (10.3)

– Δ_{22}= 51689 soit 1.172 seconde.

Nous retrouvons donc bien nos répliques artéfacts mesurées à la figure 3.3. De plus cette méthode permet une identification plus précise du Δ que la méthode proposée juste avant et qui consistait à mesurer l'écart entre la réplique et le noyaux d'ordre 1 afin d'en déduire le Δ en inversant les relations (3.16) et (3.18).

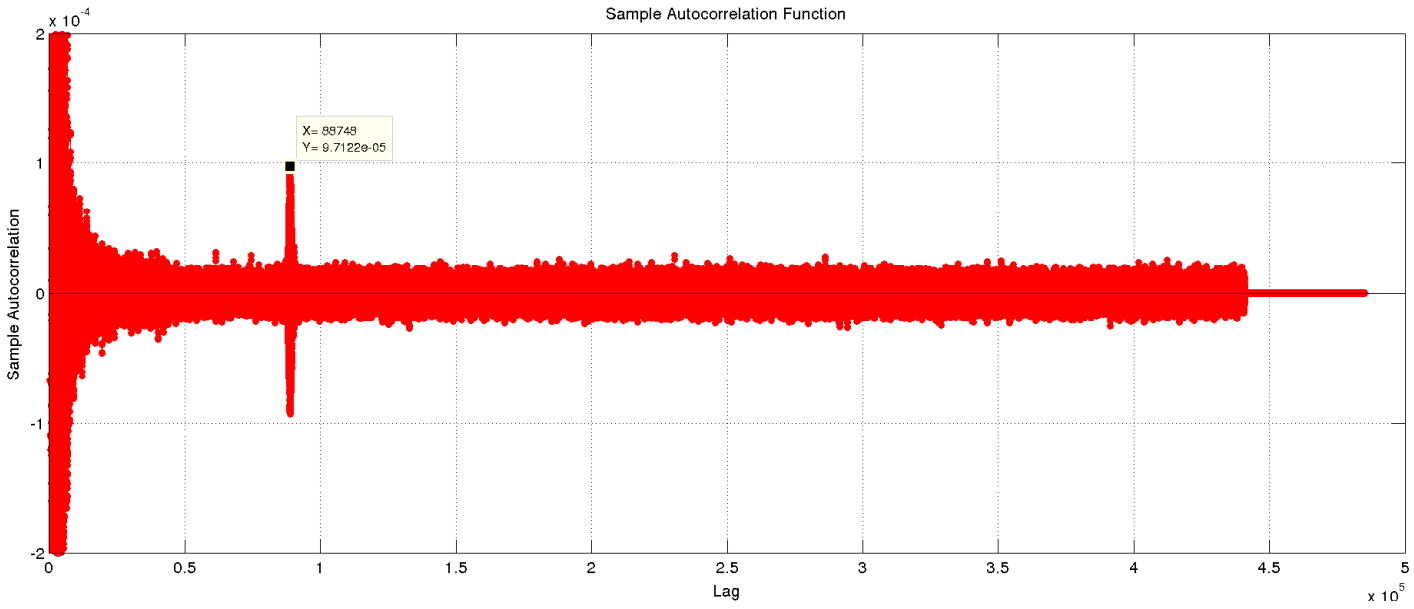


Figure 3.4 – Fonction d'auto-corrélation de la réponse impulsionnelle linéaire du haut parleur.

En suivant le même principe, nous pouvons nous demander quelles sont les répliques provoquées par γ_{31} et γ_{33}. L'article de Torras et Jacobsen nous donne les formules suivantes :

$$\gamma_{31} = N \frac{ln\left[\left|\left(\frac{w_2}{w_1}\right)^{\frac{-\tau_1}{N}} - \left(\frac{w_2}{w_1}\right)^{\frac{-\tau_2}{N}} + \left(\frac{w_2}{w_1}\right)^{\frac{-\tau_3}{N}}\right|\right]}{ln\left(\frac{w_2}{w_1}\right)} \tag{3.20}$$

$$\gamma_{33} = N \frac{ln\left[\left(\frac{w_2}{w_1}\right)^{\frac{-\tau_1}{N}} + \left(\frac{w_2}{w_1}\right)^{\frac{-\tau_2}{N}} + \left(\frac{w_2}{w_1}\right)^{\frac{-\tau_3}{N}}\right]}{ln\left(\frac{w_2}{w_1}\right)} \tag{3.21}$$

Remplaçons dans les équations (3.20) et (3.21), τ_1, τ_2, τ_3 par une des combinaisons suivantes :

$$\begin{cases} \tau_2 = \tau_1 \\ \tau_3 = \tau_1 + \Delta(\Delta > 0) \end{cases} \tag{3.22}$$

ou

$$\begin{cases} \tau_3 = \tau_1 \\ \tau_2 = \tau_1 + \Delta(\Delta > 0) \end{cases} \tag{3.23}$$

La première de ces combinaisons donne lieu à une réplique située **après** la réponse linéaire :

$$\gamma_{31}^{a} = N \frac{ln\left[\left(\frac{w_2}{w_1}\right)^{\frac{-(\tau_1+\Delta)}{N}}\right]}{ln\left(\frac{w_2}{w_1}\right)} = -\tau_1 + \Delta \tag{3.24}$$

et

$$\Delta_{31}^{a} = -\Delta \tag{3.25}$$

La seconde combinaison donne lieu à une réplique située entre h_2' et h_1' :

$$\begin{cases} \gamma_{31}^{b} = N \dfrac{ln\left(\frac{w_2}{w_1}\right)^{\frac{-\tau_1}{N}} + ln\left[2 - \left(\frac{w_2}{w_1}\right)^{\frac{-\tau_2}{N}}\right]}{ln\left(\frac{w_2}{w_1}\right)} \\ \quad = -\tau_1 + N \dfrac{ln\left[2 - \left(\frac{w_2}{w_1}\right)^{\frac{-\Delta}{N}}\right]}{ln\left(\frac{w_2}{w_1}\right)} \end{cases} \tag{3.26}$$

et

$$\Delta_{31}^{b} = N \frac{ln\left(2 - \left(\frac{w_2}{w_1}\right)^{\frac{-\Delta}{N}}\right)}{ln\left(\frac{w_2}{w_1}\right)} \tag{3.27}$$

Finalement, les deux combinaisons donnent lieu à une réplique située entre h_3' et h_2' :

$$\begin{cases} \gamma_{33} = N \dfrac{ln\left(\frac{w_2}{w_1}\right)^{\frac{-\tau_1}{N}} + ln\left[2 + \left(\frac{w_2}{w_1}\right)^{\frac{-\tau_2}{N}}\right]}{ln\left(\frac{w_2}{w_1}\right)} \\ \qquad = -\tau_1 + N \dfrac{ln\left[2 + \left(\frac{w_2}{w_1}\right)^{\frac{-\Delta}{N}}\right]}{ln\left(\frac{w_2}{w_1}\right)} \end{cases} \tag{3.28}$$

et

$$\Delta_{33} = N \frac{ln\left(2 + \left(\frac{w_2}{w_1}\right)^{\frac{-\Delta}{N}}\right)}{ln\left(\frac{w_2}{w_1}\right)} \tag{3.29}$$

En remplaçant dans ces formules le Δ obtenu dans la fonction d'auto-corrélation, nous obtenons :

- Δ_{31}^{a}=-2.012 secondes
- Δ_{31}^{b}= 1.176 seconde
- Δ_{33}= 2.652 secondes

Nous constatons que Δ_{31}^{a} et Δ_{20} ainsi que Δ_{31}^{b} et Δ_{22} ont des valeurs identiques ce qui rend l'identification de ces répliques difficile. Par contre la répliques γ_{33} est bien visible [17]

3.3 Limitation des séries de Volterra

La série de Volterra possède essentiellement deux problèmes :

1. Le premier problème consiste en la mesure des différents noyaux, ils sont difficiles à séparer et à identifier dans la réponse totale du système.
2. Le second problème concerne la convergence de la série, celle-ci peut ne converger que sur un certain domaine de l'amplitude du signal d'entrée. La série de Volterra est intimement liée à la série de Taylor [47]. Par conséquent, les limitations de l'une entraînent les limitations de l'autre.

   En effet, reprenons le modèle non-linéaire 1 (figure 3.1), constitué d'une réponse impulsionnelle classique puis d'un système non-linéaire sans mémoire, qui peut être représenté en un développement en série de Taylor.

17. réplique située entre h'1 et h'2 avec un Δ_{33}= 2.653 secondes.

Si la série diverge pour certaines valeurs w[n] de son entrée et que x[n] est non borné, alors, la série divergera. En particulier, il n'existe pas de série de Taylor convergente pour un système du type z[n]=V_{sat} *sign*(y[n]). Par conséquent la série de Volterra ne peut exprimer les phénomènes de saturations.

4

Mesure de réponses impulsionnelles sur un haut-parleur

Dans ce chapitre, nous présenterons :

- le matériel de mesure utilisé et ses non-idéalités,
- les différentes mesures effectuées lors de mes expérimentations en laboratoires,
- la reconstruction des noyaux de Volterra caractérisant 2 haut-parleurs différents.

4.1 Enceinte à caractériser

Deux enceintes sont modélisées :

- Une enceinte open back de dimension [60,48,34] cm, montée avec un haut-parleur Eminence, le *wizard 16* [1].
- Une enceinte closed back de dimension [47,47,28] cm, le BH112 [19] de chez *Blackheart*, monté avec un haut-parleur de chez Eminence le *1216B*

Les 2 haut-parleurs sont prévus pour une puissance de 75 W_{rms} et possèdent une impédance de 16 ohms. La réponse fréquentielle du Wizard16 [1] est fournie par sa datasheet et est présentée à la figure 4.1. Sur cette même figure, la courbe grise représente l'impédance du haut-parleur en fonction de la fréquence.

Note :

Le haut-parleur *wizard16* possède un rendement très élevé pour un haut-parleur guitare (103 dbspl/W/m). Par conséquent, pour une même puissance électrique fournie par l'ampli, le signal recueilli au niveau du micro est plus important pour le haut-parleur *wizard* que pour le *1216B*.

1. La datasheet du haut-parleur 1216B de chez eminence est introuvable.

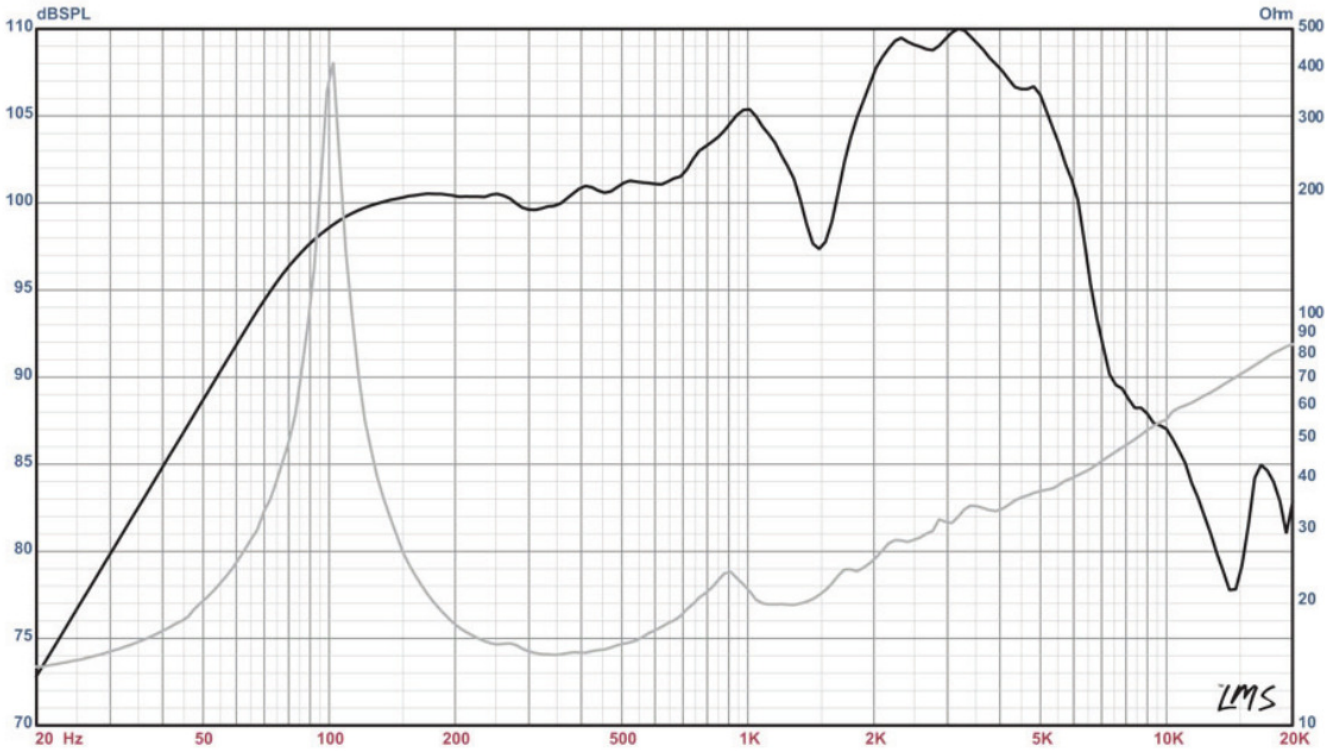


Figure 4.1 – La courbe noire représente la réponse en fréquence du haut-parleur "Wizard16" (voir condition de mesure dans la datasheet [1]) et la courbe grise, l'impédance du haut parleur.

4.2 Système de mesure

Le schéma général de mesure est présenté à la figure 4.2. La carte son de l'ordinateur sert à la fois pour générer le signal d'excitation du système (sweep) et pour enregistrer le signal provenant du microphone placé devant le haut-parleur .

Matériel de mesure

Dans ce travail, nous voulons étudier les non-linéarités du haut-parleur seul. Malheureusement le résultat de nos mesures est, en réalité, le fruit de la mise en cascade des différents appareils utilisés. Nos mesures sont donc entachées d'une certaine quantité d'erreurs, relatives aux non-idéalités de l'appareillage. Il convient de caractériser le plus précisément ces non-idéalités afin de *justifier* les écarts observés entre la théorie et la pratiques.

Afin de ne pas alourdir la lecture, la description de l'appareillage de mesure et ses non-idéalités sont présentés dans l'appendix D section 11.1 page 116. Néanmoins , la section suivante décrit les non-idéalités de la carte son utilisée pour générer le sweep car cet exemple nous permettra de mettre en évidence certains problèmes liés à la comparaison de réponses impulsionnelles obtenues par excitation du système via un sweep logarithmique

ou un sweep linéaire.

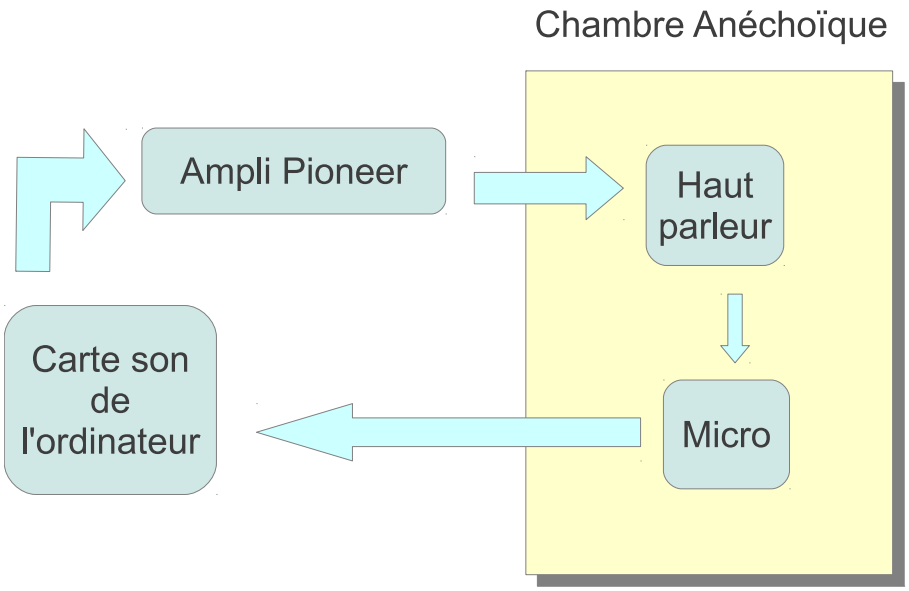


Figure 4.2 – Schéma de mesure pour la prise de réponses impulsionnelles

4.2.1 Carte son 1 : waveterminal 2496 [20]

Description des convertisseurs de la carte son :

- 24-Bit/96kHz A/D converter ; 100dB Dynamic range and 64 x oversampling
- 24-Bit/96kHz D/A converter ; 110dB Dynamic range and 128 x oversampling
- Signal-to-Noise Ratio (D/A) ; 110dB

Description de la bande passante de la carte son :

Afin de vérifier la bande passante de la carte son, nous analyserons la transformée de Fourier de sa réponse impulsionnelle. Pour générer le sweep , nous utilisons le séquenceur Adobe Audition avec le plugin Aurora [4] (générateur de sweep). La réponse impulsionnelle de la carte son est obtenue [2] en envoyant le sweep sur une des sorties de la carte et en réinjectant ce signal dans une des entrées de la carte son. La courbe de réponse est présentée à la figure 4.3.

Nous remarquons :

- Une bonne linéarité de la carte son dans la bande passante de la guitare (80 Hz - 20 kHz).
- Une oscillation de la courbe de réponse entre 20 et 30 Hz [3]

2. Voir section 2.3 page 12
3. Voir section 4.2.1.2

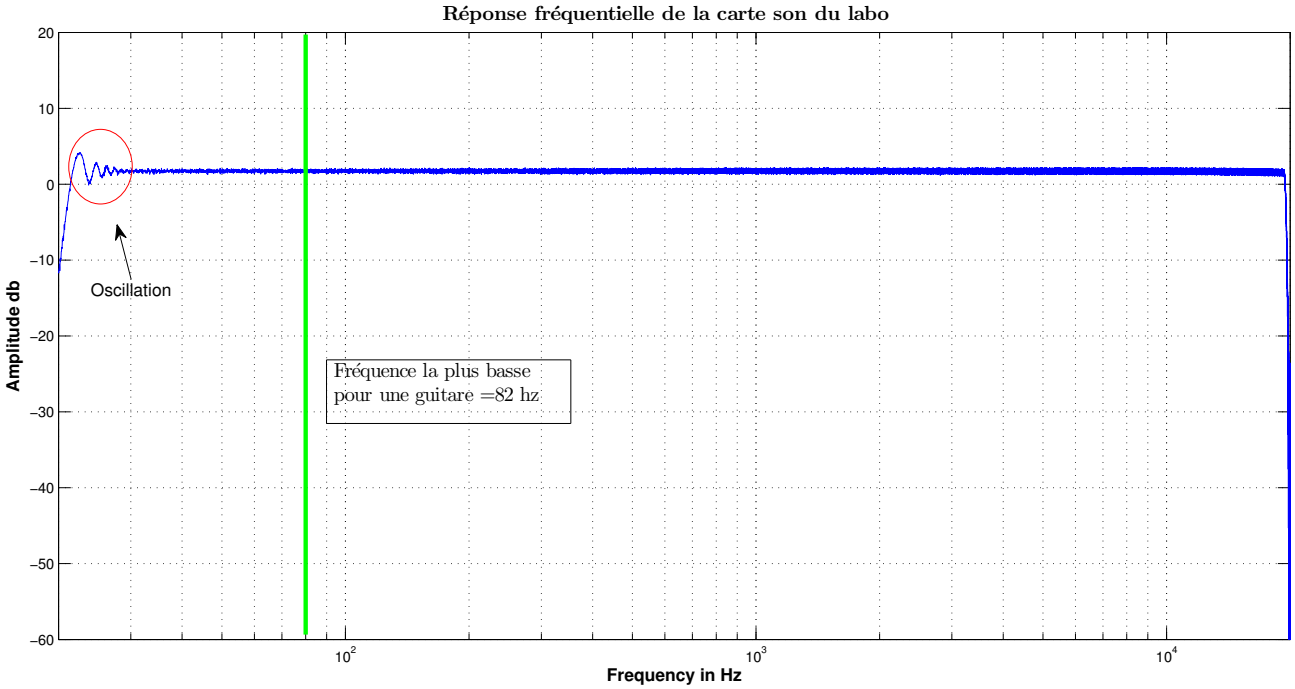


Figure 4.3 – Réponse fréquentielle de la carte son Waveterminal (*type d'excitation : Logsweep*

Pour nos futures mesures, il est important de vérifier que la réponse impulsionnelle obtenue en excitant un système linéaire à l'aide d'un sweep logarithmique ou linéaire conduit bien à la même courbe de réponse fréquentielle du système [4]. La réponse fréquentielle de la carte son obtenue lorsque le signal d'excitation est un sweep linéaire est présentée à la figure 4.4.

Nous remarquons :

- La fréquence de coupure est plus haute que pour une excitation en logsweep.
- En basse fréquence, le signal semble bruité.

note :

Les sweeps utilisés (linéaire et logarithmique) possèdent les mêmes paramètres et balaient les fréquences comprises entre 20 et 20 000 Hz en 20 secondes.

4. En effet, si le système est linéaire, les contributions h'3 et h'5 sont très faibles et nous pouvons les négliger de prime abord.

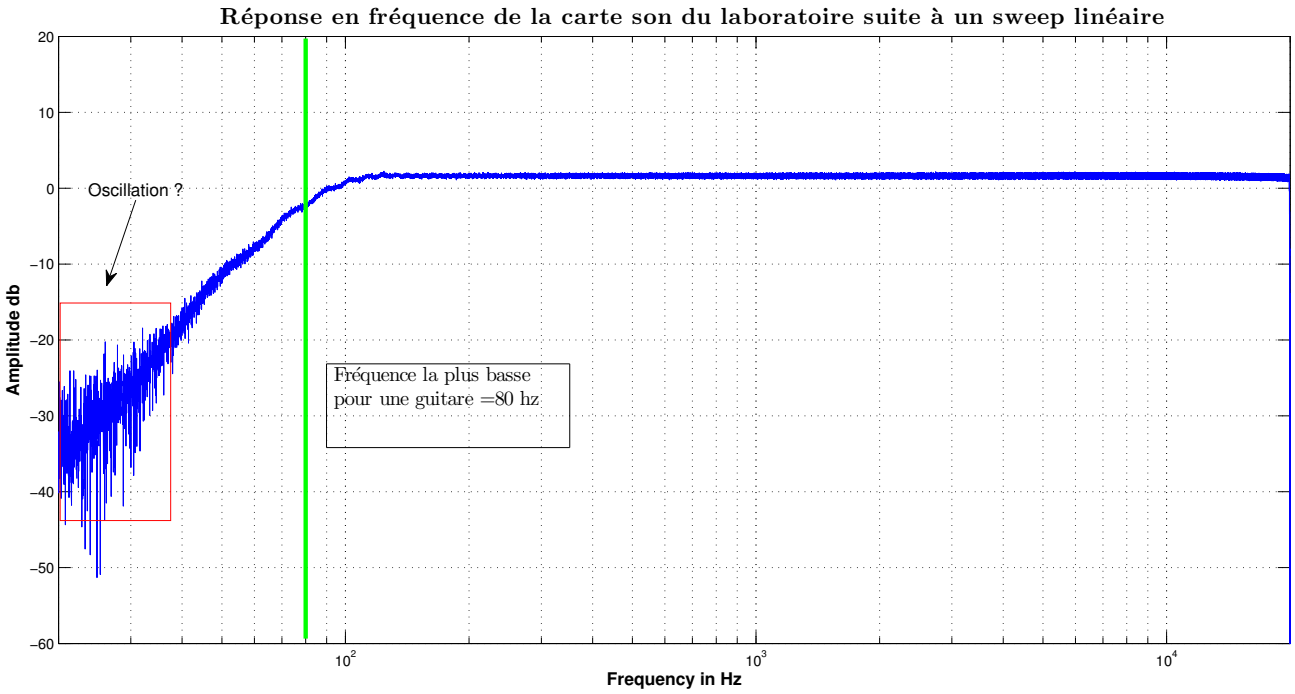


Figure 4.4 – Réponse fréquentielle de la carte son Waveterminal (*type d'excitation : sweep linéaire*)

4.2.1.1 Comparaison Lin et Log sweep

Comme nous venons de le voir à la section précédente, Lin et Log sweep ne sont en pratique, pas équivalents pour un même système linéaire, contrairement à la théorie développée à la section 3.10.

Pour mieux comprendre la source de ce problème,intéressons nous à :

1. la transformée de Fourier d'un Logsweep déconvolué par son filtre inverse (figure 4.5)
2. la transformée de Fourier d'un Linsweep déconvolué par son filtre inverse (figure 4.6)

Ces réponses ne dépendent pas d'un matériel Hardware et sont générées directement dans Adobe Audition. Comme nous le constatons, la fréquence de coupure dans le cas d'une excitation linéaire est toujours plus haute que dans le cas d'une excitation logarithmique.

Nous constatons également que les oscillations ont pratiquement disparu. Cette dernière différence s'explique par l'application d'une fenêtre de Hann lors de la FFT dans Adobe Audition, alors qu'il n'y a pas de fenêtrage appliqué dans mes fonctions matlab. (voir [40] pour plus d'explications sur l'utilité du fenêtrage lors de la FFT).

Notons d'ores et déjà que dans le cas d'une excitation en Linsweep, la fréquence de coupure du système varie avec la longueur du sweep (diminution de la fréquence de coupure mais augmentation des oscillations). Comme nous pouvons le constater à la figure 4.7 où la durée du sweep est passée de T=20 secondes(figure 4.6) à T=100 secondes). La fréquence de coupure descend alors aux alentours de 40 hz au détriment de la linéarité dans la bande passante comme nous le verrons à la section 4.2.1.2 .

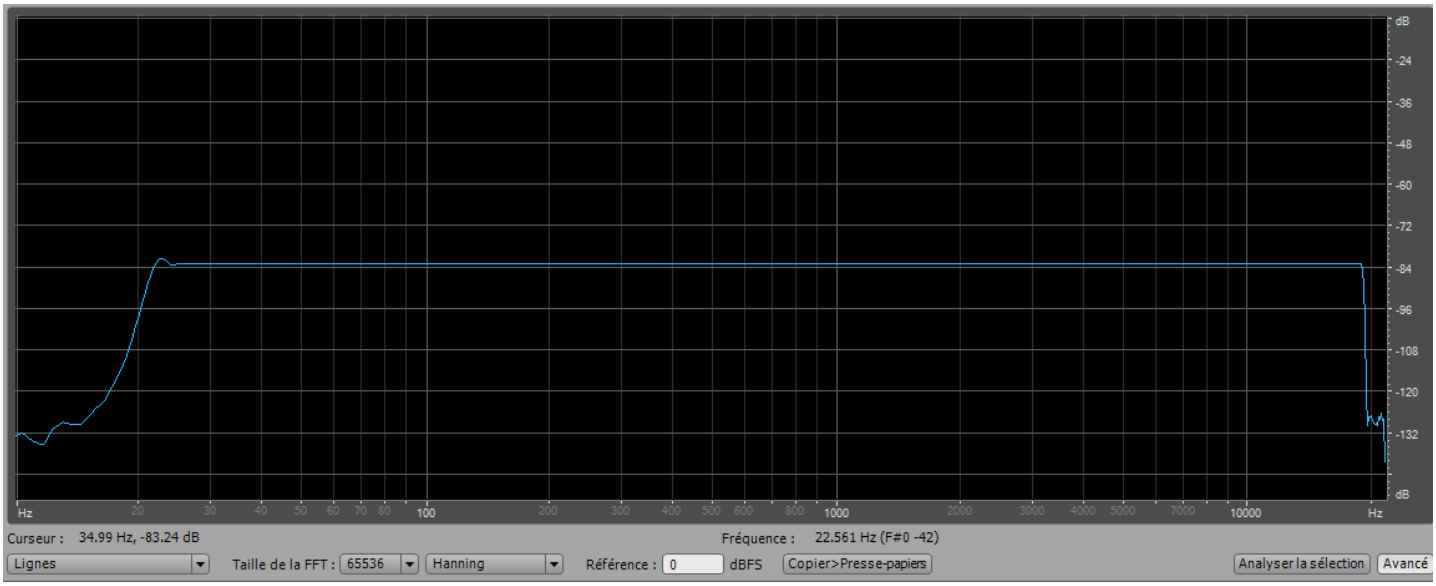


Figure 4.5 – Transformée de Fourier du Logsweep convolué avec son filtre inverse

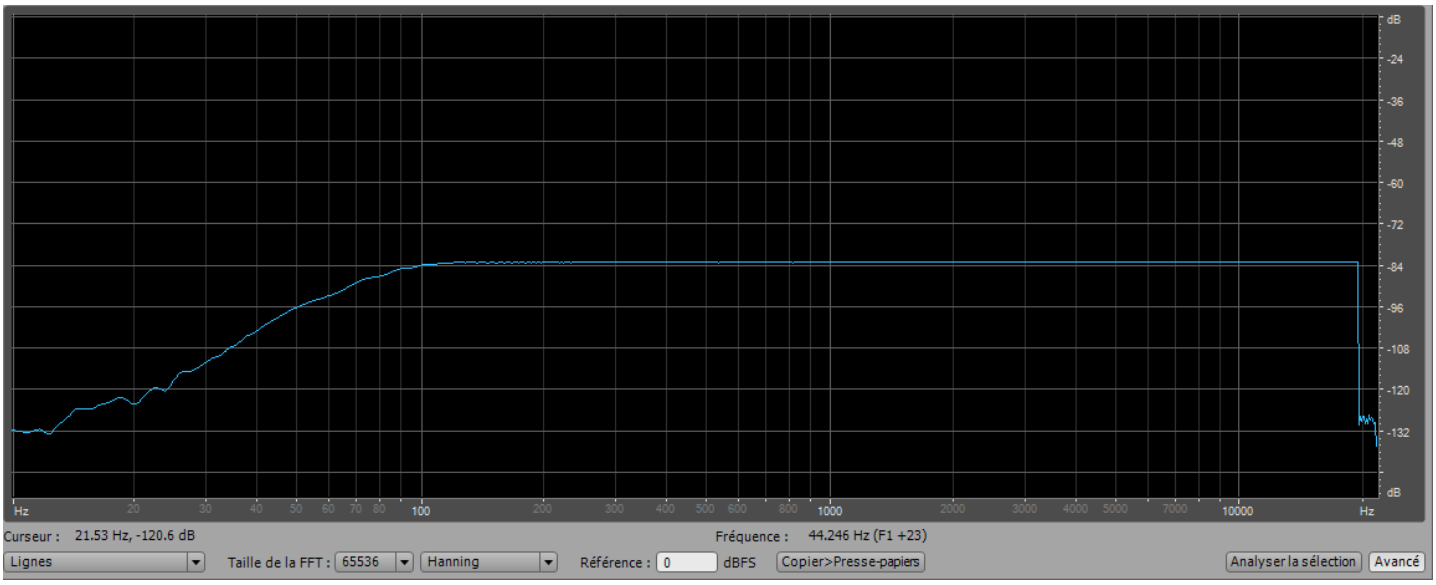


Figure 4.6 – Transformée de Fourier d'un sweep linéaire convolué avec son filtre inverse

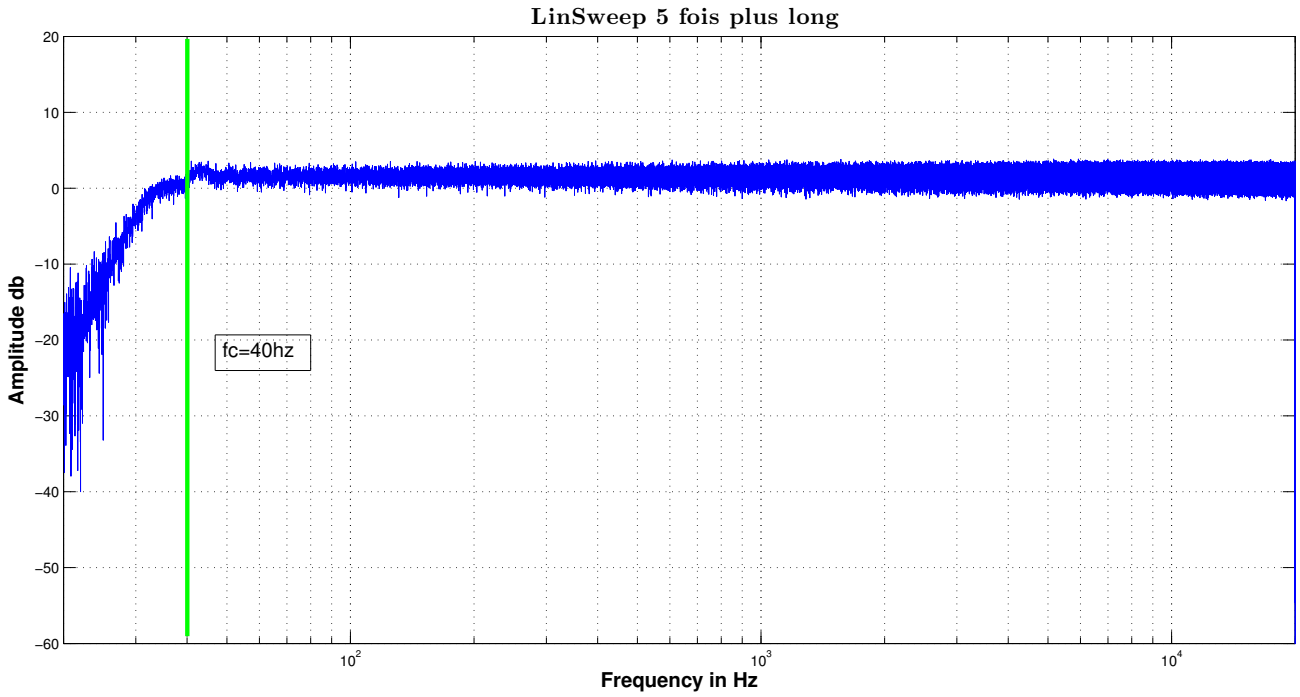


Figure 4.7 – Transformée de Fourier du Linsweep de durée T=100 secondes (soit une durée 5 fois plus importante par rapport au sweep utilisé pour la figure 4.6) convolué avec son filtre inverse.

4.2.1.2 Le fenêtrage

Soit un signal de type sweep :

$$x(t) = Asin(f(t))$$

Le sweep est borné dans le temps, il commence en t=0 et fini en t=T (T = 20 secondes pour nos sweeps) ce qui revient à fenêtrer le signal x(t) par une fonction rectangle h(t).

Où :

$$h(t) = \begin{cases} 1 & \text{si t} \in [0,T] \\ 0 & \text{sinon} \end{cases} \tag{4.1}$$

Nous étudions donc le signal $x_h(t) = x(t)h(t)$ et non le signal x(t), en fréquentiel nous avons :

$$X_h(w) = X(w) \circledast H(w)$$

Où :

– X(w) est la transformée de Fourier du signal x(t)

– H(w) est la transformée de Fourier de la fenêtre utilisée.

Idéalement, nous voudrions que la transformée de Fourier de la fenêtre h(t) utilisée soit une impulsion de Dirac de sorte que :

$$X_h(w) = X(w) \circledast \delta(t) = X(w)$$

Or la transformée de Fourier d'une fenêtre rectangulaire z(t) de largeur 2T centrée en zéro est la fonction sync tel que :

$$Z(jw) = 2\frac{sin(wT)}{w}$$

Cette fonction est tracée à la figure 4.8 pour des fenêtres de plus en plus large (T=1,2,3). L'amplitude du lobe principal vaut 2T et les 2 premiers zéros de la fonction valent $-\frac{\pi}{T}$ et $\frac{\pi}{T}$. Par conséquent, plus le sweep est long (T est grand), plus le lobe principal devient fin et de grande amplitude. Le lobe principal se rapproche alors de la forme d'une impulsion de Dirac.

Cependant, plus le lobe principal est étroit et plus les lobes secondaires sont d'amplitudes élevées. Ces lobes secondaires sont responsables d' oscillations dans la bande passante [5].

Il nous appartient de choisir un compromis entre la largeur du lobe principal et l'amplitude des lobes secondaires, à titre d'exemple, quelques fenêtres communes sont tracées à la figure 4.9 à l'aide de l'outil *wvtool* de Matlab. La fenêtre rectangle est celle qui possède le lobe principal le plus fin. En contre partie, elle possède des lobes secondaires importants. A contrario, la fenêtre de Blackman possède les lobes secondaires les moins importants mais le lobe principal le plus large.

Le choix de la fenêtre optimal à utiliser n'a pas été déterminé dans cette étude et la fenêtre choisie est celle par défaut du plugin Aurora, à savoir : la fenêtre de *Hann*.

La fenêtre de Hann va donc réduire l'amplitude du signal au début et à la fin du sweep. L'effet net est une réduction des basses et hautes fréquences en fonction de la portion du sweep (en temporel) atténué par la fenêtre.

Exemple :

Considérons une fenêtre diminuant l'amplitude du signal x(t) sur 1 dixième de seconde soit 4410 samples :

5. Cette dernière remarque explique la variation d'amplitude des oscillations dans la bande passante lorsque nous passons de la figure 4.6 (Lin sweep de 20 secondes) à la figure 4.7 (Lin sweep de 100 secondes)

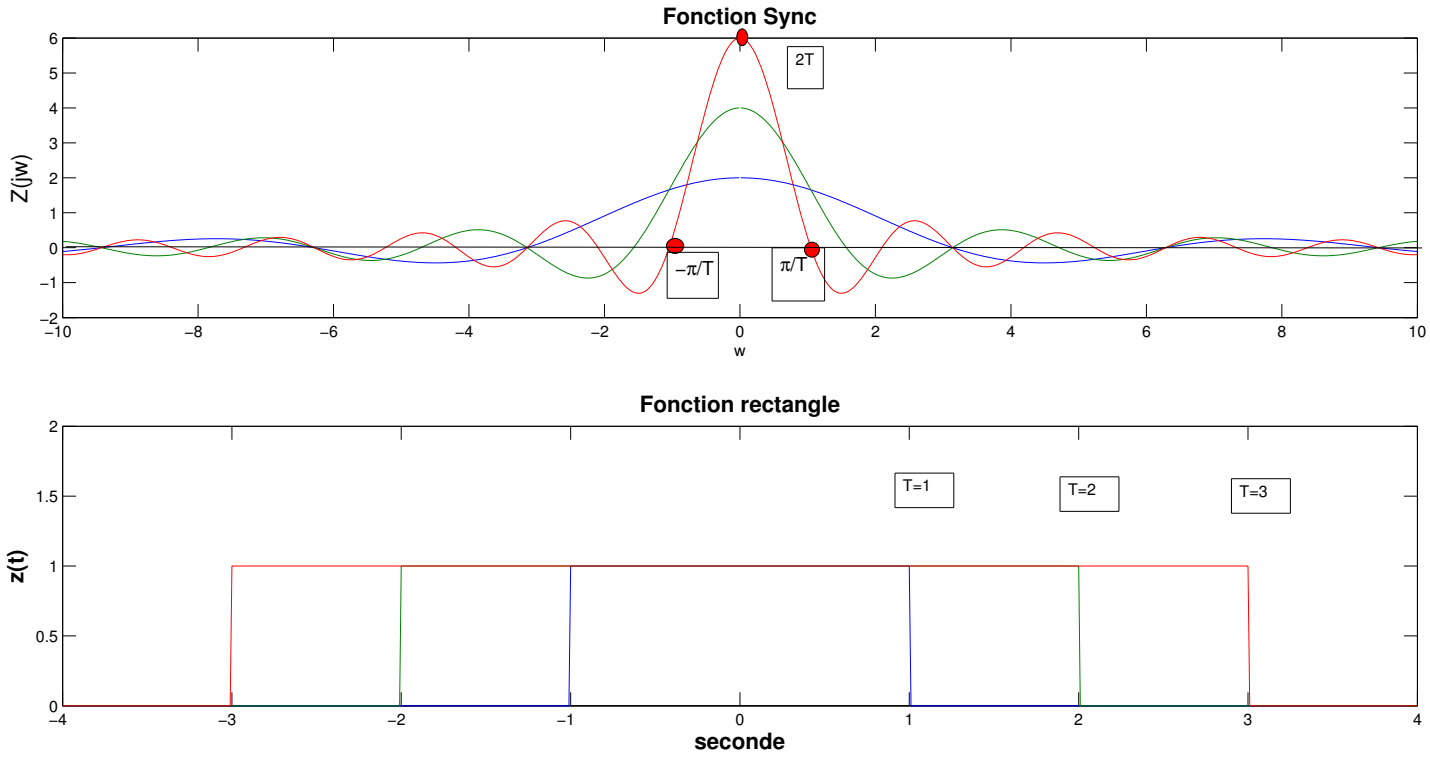


Figure 4.8 – Transformée de Fourier de la fonction rectangle, pour des longueurs de rectangles variant entre 2 et 6 secondes

Pour un Linsweep, la formule (2.18) pour t = 0.1 seconde nous donne la fréquence instantanée du sweep linéaire :

$$w(t) = w_1 + \frac{w_2 - w_1}{T} t = 2\pi 20 + \frac{20000 * 2\pi - 20 * 2\pi}{20} 0.1 = 753 rad/s \qquad (4.2)$$

Alors que pour un Logsweep, la fréquence instantanée s'obtient en dérivant l'argument du sinus de la formule (4.14) par rapport au temps, soit :

$$w(t) = \left[\frac{w_1 T \frac{1}{T} ln(\frac{w_2}{w_1})}{ln(\frac{w_2}{w_1})} \left(e^{\frac{t}{T} ln(\frac{w_2}{w_1})}\right) \right] = w_1 \left(e^{\frac{t}{T} ln(\frac{w_2}{w_1})}\right) = 130 rad/s \qquad (4.3)$$

Il s'ensuit que pour une même fenêtre de Hann appliquée à différents types de sweep, le résultat sur le spectre ne sera pas le même. En effet pour un Linsweep, les fréquences en dessous de 119Hz seront atténuées, alors que pour le Logsweep seules les fréquences inférieures à 20.69 Hz seront atténuées [6].

Pour illustrer cet exemple, la figure 4.10 présente la réponse en fréquence d'un sweep linéaire convolué avec son filtre inverse dans le cas d'une fenêtre ayant un *fade in* de 0.01

6. 753 rad/s = 119Hz et 130 rad/s = 20.69 Hz

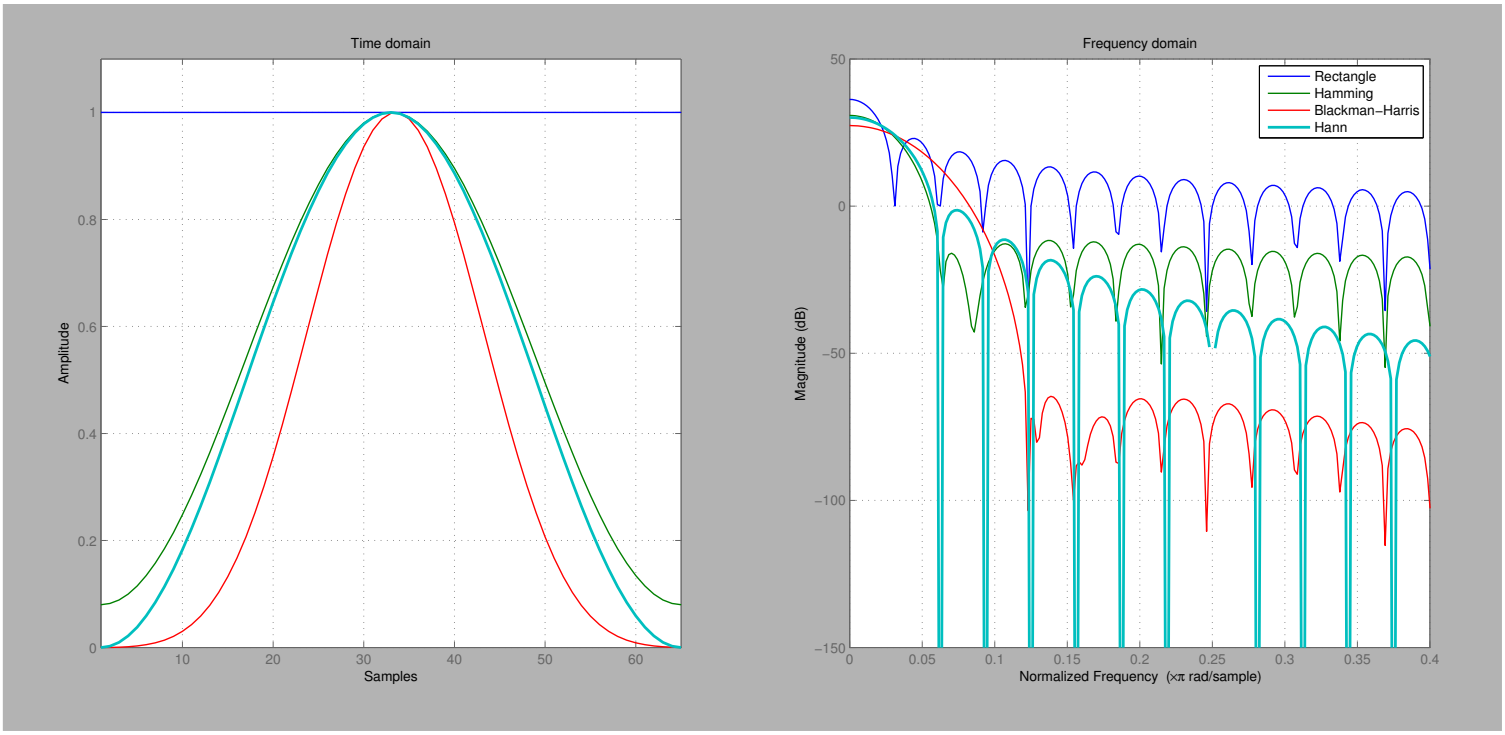


Figure 4.9 – Représentation temporelle et spectrale de quelques fenêtres communes

seconde, au lieu de 0.1 seconde comme sur la figure 4.6. En reprenant l'équation (4.3) pour t=0.01, nous obtenons w(0.01)=188 rad/s, soit une atténuation des fréquences en-dessous de 30 hertz.

La figure 4.11 nous présente l'interface du plugin Aurora servant à générer les différents sweeps.

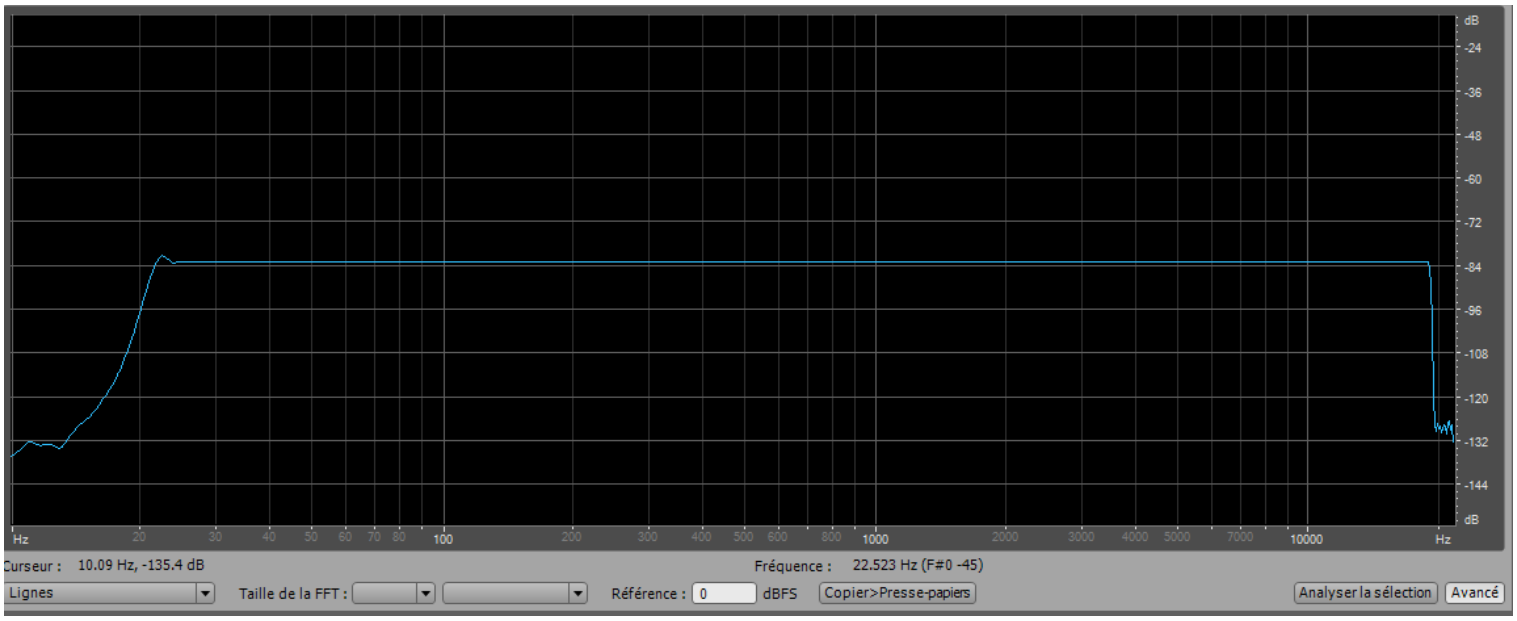


Figure 4.10 – Transformée de Fourier d'un sweep linéaire convolué avec son filtre inverse, avec fade in de 0.01 seconde au lieu de 0.1 s (comme à la figure 4.6).

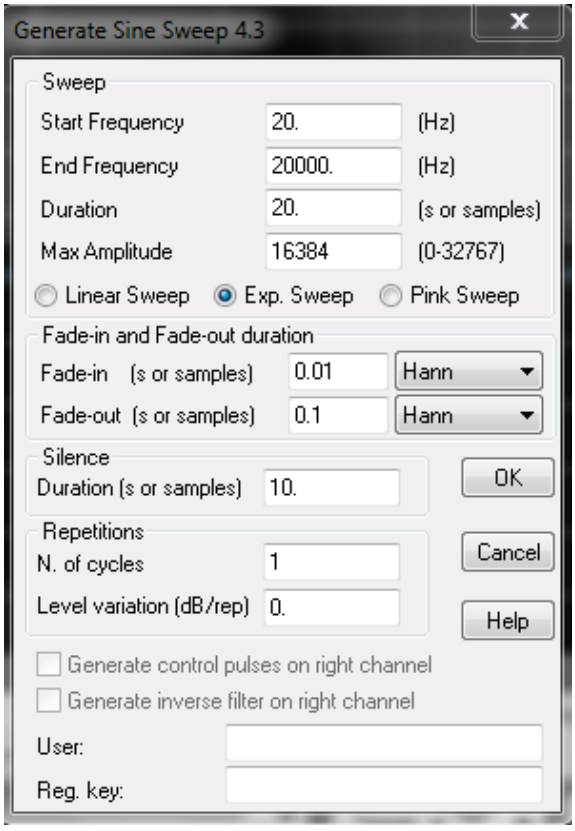


Figure 4.11 – Générateur de sweep Aurora dans adobe audition

4.3 Reconstructions des noyaux de Volterra

4.3.1 Les étapes de l'obtention des noyaux de Volterra.

Étapes à mettre en relation avec le schéma de mesure de la figure 4.2

1. Génération du sweep x(t) et de son filtre inverse f(t) [7], le signal est envoyé sur une sortie de la carte son *Waveterminal* .
2. Amplification du signal x(t) via l'amplificateur Kenwood afin d'atteindre une amplitude de 31Vrms aux bornes du haut-parleur.
3. Prise du signal acoustique émis par le haut-parleur, via le microphone metravib.
4. Enregistrement d'un signal y(t) provenant du microphone, via la carte son *Waveterminal* et le séquenceur *Adobe Audition*.
5. Déconvolution $\Rightarrow y(t) \circledast f(t) = h(t) \Rightarrow$ obtention de la réponse impulsionnelle.
6. Séparation des noyaux h'_i et reconstitution des noyaux de Volterra via les équations (3.13) [8]

4.3.2 Objectif

L'objectif fixé est de pouvoir reconstruire à partir d'une réponse impulsionnelle h(t), toutes les réponses impulsionnelles du haut-parleur pour tous les niveaux d'entrées.

Nous comparerons donc :

- la transformée de Fourier, de la réponse impulsionnelle obtenue en déconvoluant la réponse du système lorsqu'on applique un sweep linéaire à un niveau donné sur son entrée,
- la réponse fréquentielle reconstituée au moyen de la série de Volterra et de ses noyaux hv_i pour ce même niveau d'entrée.

En pratique, il est impossible de comparer toutes les réponses fréquentielles pour tous les niveaux d'entrés. Nous nous limiterons à 2 cas particulier :

1. Reconstitution et comparaison des réponses à **bas** niveaux (voir paragraphe suivant).
2. Reconstitution et comparaison des réponses à **haut** niveaux .

- Le **bas** niveau, représente la tension à appliquer au haut-parleur pour que celui-ci puisse être considéré comme un système linéaire.

  Cependant le niveau à appliquer au haut-parleur pour que ses non-linéarités disparaissent est tellement faible que la réponse impulsionnelle est noyée dans le bruit.

7. Aurora plug-ins d'Adobe Audition
8. Voir section 8.2 page 94 pour l'utilisation du code Matlab permettant de calculer les noyaux de Volterra sur base du signal h(t)

Une amélioration du rapport signal à bruit est possible en répétant la mesure N fois et en additionnant les résultats [9] , pour autant que le système soit invariant [10] [11]. Il est difficile d'appliquer cette propriété vu la dépendance de la sensibilité du haut-parleur en fonction de la température [12].

Dans ce travail, nous considérons qu'un signal dont l'amplitude est inférieure à 2 V_{rms} est un signal de niveau **bas**.

– Le **haut** niveau représente la tension maximale admissible par le haut-parleur. Comme il s'agit de haut-parleur de 75 W_{rms} et de 16 ohms, nous obtenons : $V_{Hp} = \sqrt{(75 * 16)} \simeq 34.6 Vrms$. Par mesure de sécurité, nous nous limiterons à 31 V_{rms} soit 60 W_{rms} [13].

4.3.3 Reconstruction des noyaux de Volterra pour le haut-parleur Blackheart

La figure 4.12 compare les réponses fréquentielles à bas et haut niveau pour le haut-parleur *Blackheart*.

Où : Hlin x représente la transformée de Fourier de la réponse impulsionnelle du haut-parleur obtenue par déconvolution d'un sweep linéaire d'une amplitude de x volt. Comme nous pouvons le constater, les différences entre les 2 réponses fréquentielles restent faibles pour cette plage de puissance [0.25 60] W_{rms}.

Nous remarquons :

– la présence d'un pic à 50 hertz,
– des oscillations dans la bande dues à l'augmentation de la durée de sweep [14] (50 secondes dans ce cas ci),
– une différence d'environs 1 db entre les réponses dans la plage [150-600] Hz,
– une différence de 3 à 7 db aux alentours de 1800 hertz.

9. le SNR augmente de 3 db à chaque doublement du nombre de mesure [52],[53]
10. voir section 2.2.1
11. Dérive de l'horloge entre le playback et l'enregistrement [2] Cette notion sort du cadre de l'étude et ne sera pas approfondie.
12. Voir section 4.3.3.4 page 57
13. le modèle n'est donc valable que pour des puissances inférieurs à 60 W_{rms}
14. voir 4.2.1.2 page 41

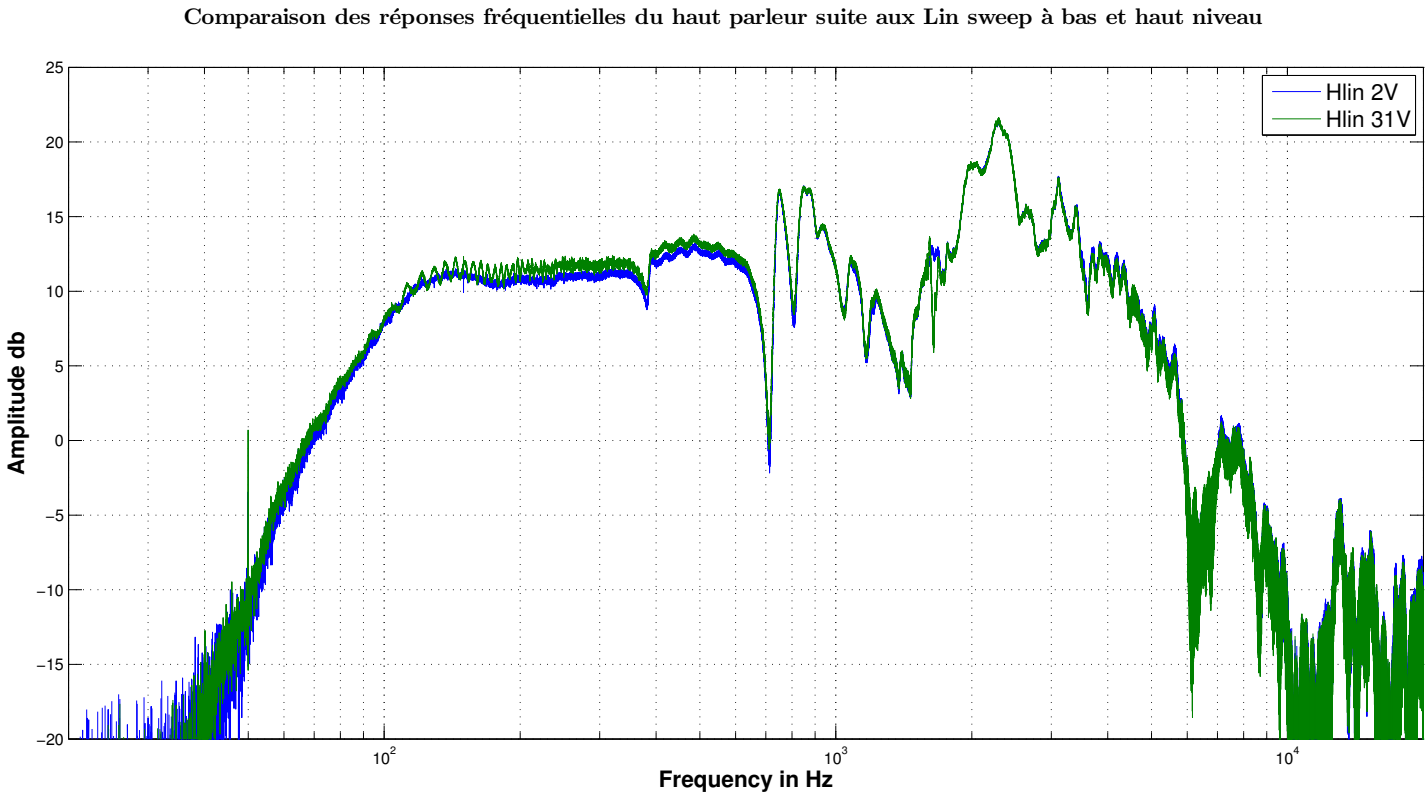


Figure 4.12 – Transformée de Fourier de la réponse impulsionnelle du haut-parleur Blackheart obtenue par déconvolution d'un sweep linéaire de bas et haut niveau appliqués en entrée.

Identification des noyaux de Volterra

La figure 4.13 nous montre la réponse impulsionnelle du haut-parleur obtenue en déconvoluant un Logsweep de $31 V_{rms}$ (voir figure 4.14) à l'aide de son filtre inverse (voir figure 4.15).

Remarquons :

– la décroissance du filtre inverse dans les basses fréquences (voir figure 4.16) : cette décroissance compense la différence d'énergie qui existe entre basses et hautes fréquences lors de la génération d'un Logsweep (vu la croissance exponentielle de la fréquence, l'énergie en basse fréquence est plus importante [21]) ;

– le *fade in* de 0.1 seconde lors de la génération du Logsweep (figure 4.14) , comme nous l'avons vu à la section 4.2.1.2.

Pour compenser la différence d'énergie entre basse et haute fréquence lors de la

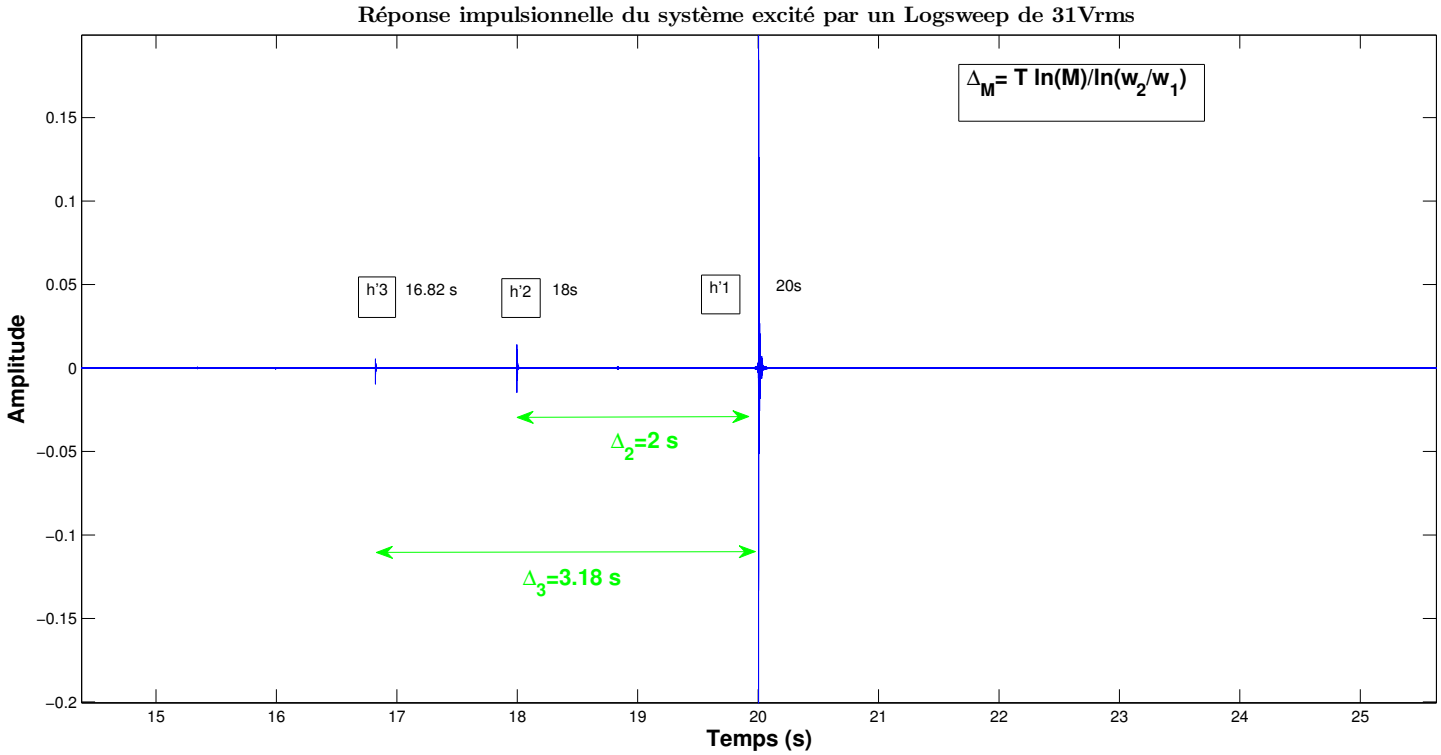


Figure 4.13 – Réponse impulsionnelle du système obtenue en déconvoluant un sweep logarithmique de 31 *Vrms* appliqué en entrée.

génération du logsweep ainsi que le fade in lors de la génération du logsweep .

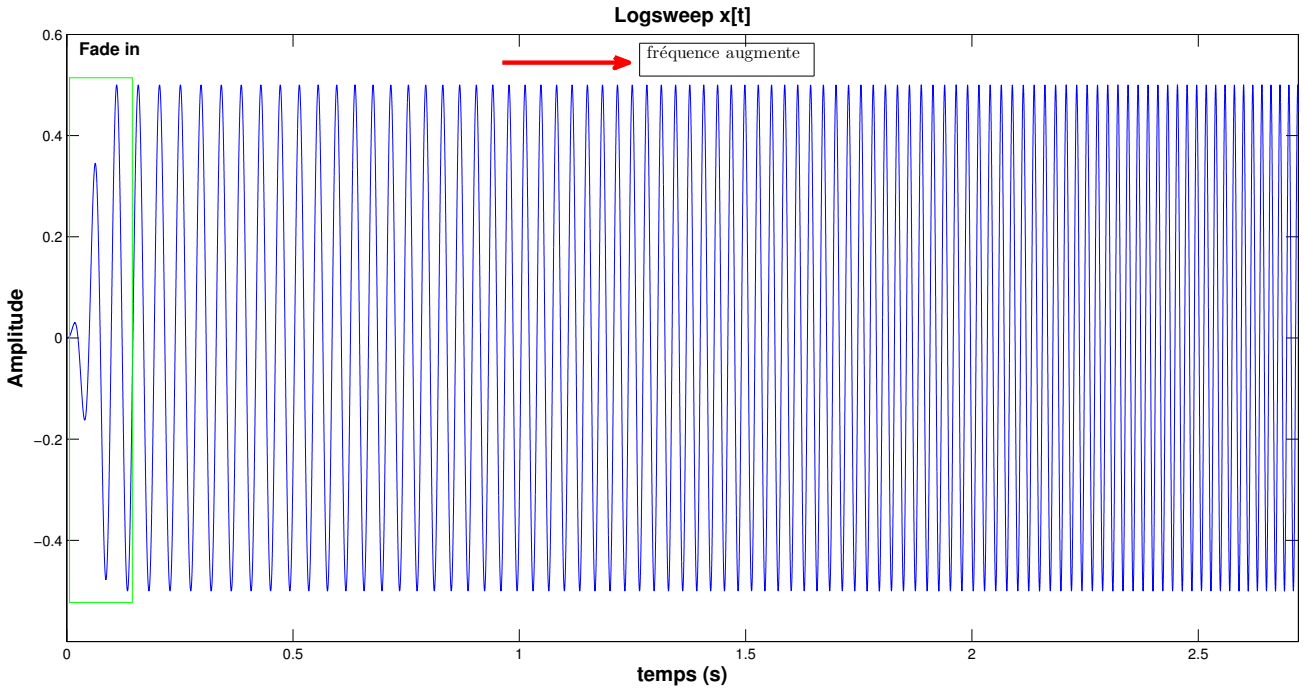


Figure 4.14 – Sine sweep logarithmique et *fade in*

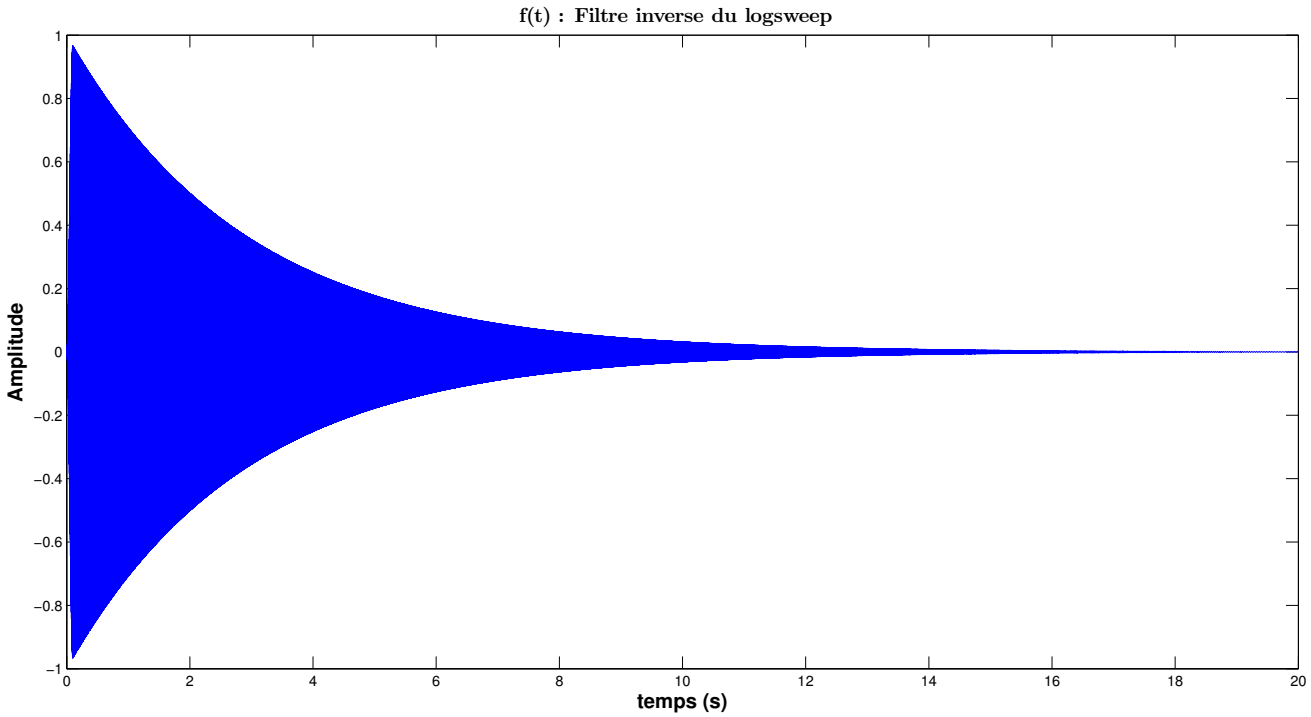


Figure 4.15 – Filtre inverse du sine sweep logarithmique

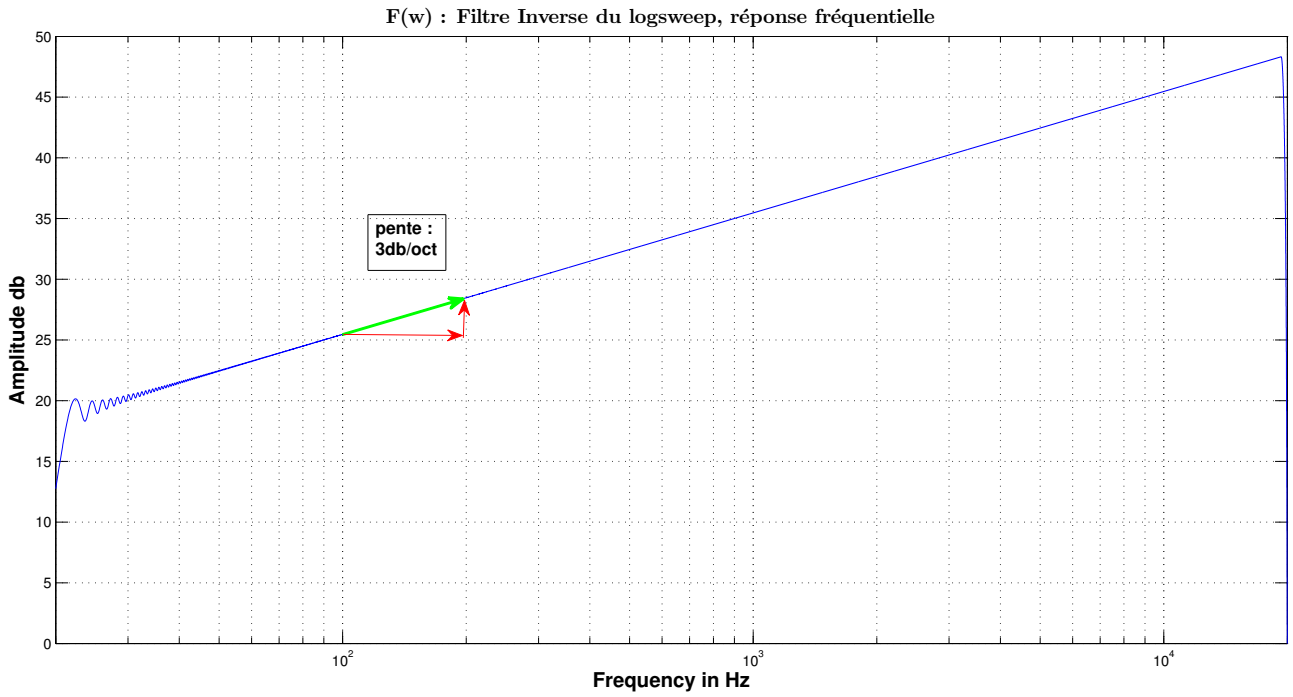


Figure 4.16 – Transformée de Fourier du filtre inverse d'un sweep logarithmique.

Remarque par rapport à la figure 4.13

– Nous appelons h'_i le noyau mesuré d'ordre i (note : ce n'est pas le noyau de Volterra).

– La théorie prédit un délai Δ_t entre h'1 le noyau dit *linéaire* et les noyaux d'ordre N [15]. La fonction matlab *timeBetweenharmo.m* [16] nous donne (de gauche à droite) le délai entre le noyau d'ordre *n* (*n* variant de 1 à N) et le noyau d'ordre 1.

Exemple pour N=5 :

```
1  >> timeBetweenHarmo(5)
2 ans =
3            0      2.0069      3.1808      4.0137      4.6598
```

De manière triviale, le délai entre le noyau d'ordre 1 et lui même vaut zéro ; les autres délais correspondent bien à ceux mesurés sur la figure 4.13.

– La réponse impulsionnelle est normalisée, l'amplitude maximale du noyau d'ordre 1 vaut donc 1.

15. Voir équation (2.20).
16. Voir la description de la fonction matlab section 8.1 page 94

– Les noyaux d'ordre 4 et 5 représentent en amplitude, moins d'un % de la réponse d'ordre 1. Pour la reconstruction du système de Volterra, nous nous limiterons donc aux 3 premiers noyaux.

– En zoomant sur la figure 4.13, nous observons les répliques γ_{22} (voir figure 4.17) et γ_{20} (figure 4.18) dont nous avons parlé à la section 3.2 page 27.

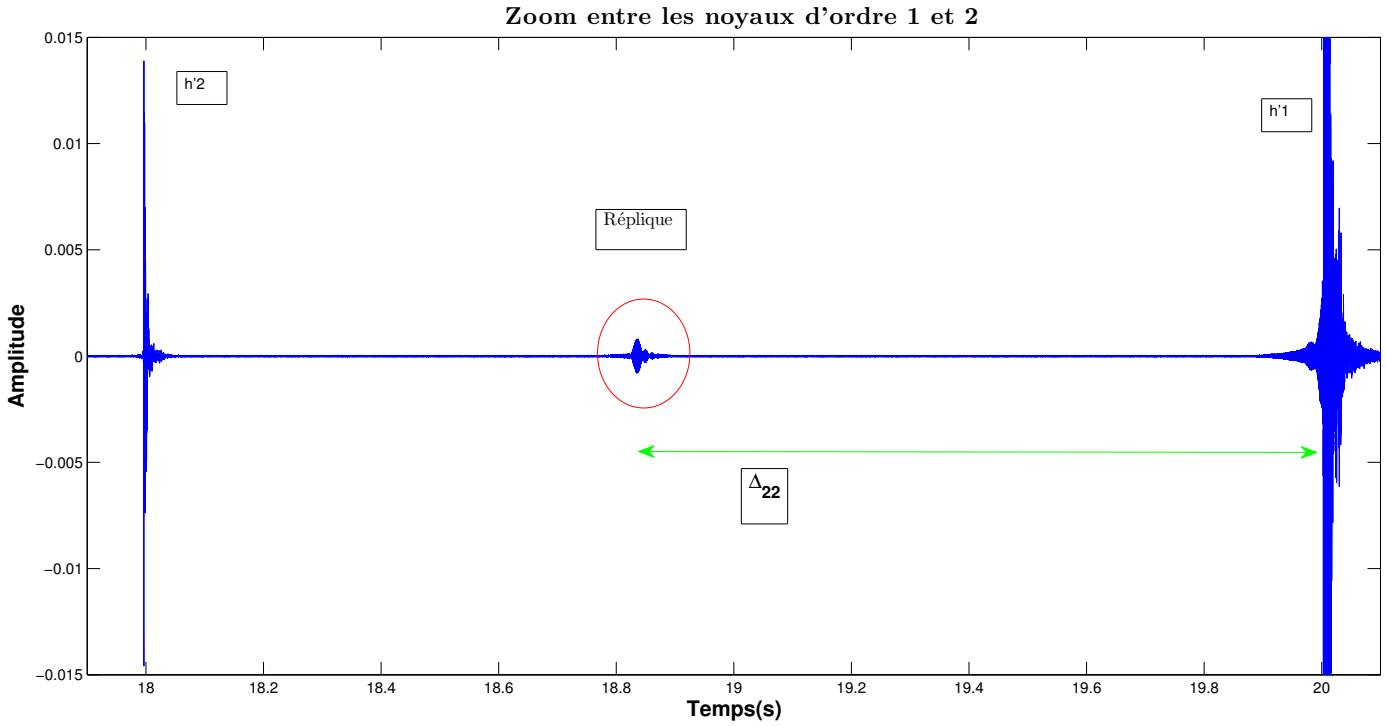


Figure 4.17 – Réplique artéfact décalée de Δ_{22} (théorie de la section 3.2).

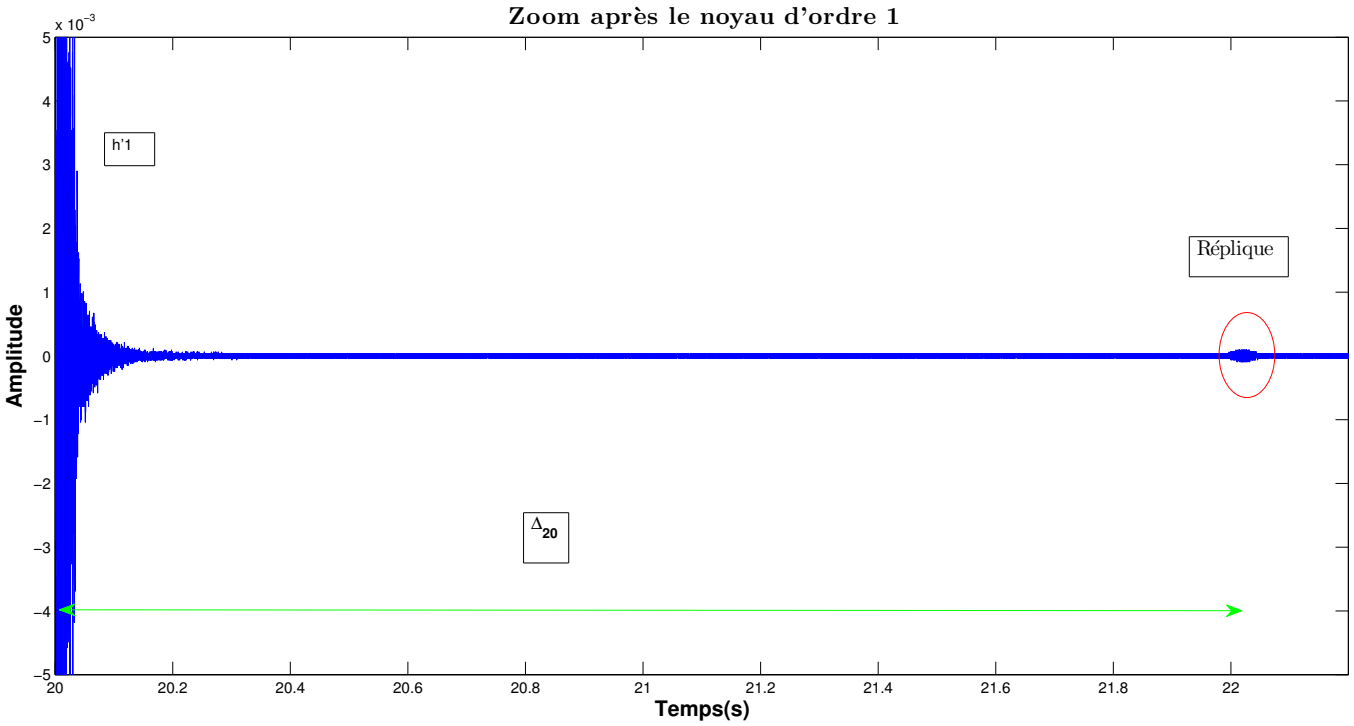


Figure 4.18 – Réplique artéfact décalée de Δ_{20} (théorie de la section 3.2).

4.3.3.1 Isoler les noyaux mesurés h'_i

Les noyaux *mesurés* h'_i ne sont que des portions de la réponse impulsionnelle h(t) du haut-parleur, obtenue en déconvoluant la réponse du système y(t) mesurée lorsqu'on applique un sweep logarithmique x(t) en entrée. Ces noyaux *mesurés* sont nécessaires à la reconstruction des noyaux de Volterra (voir équation (3.13)). Ils doivent être séparés [17] et mis en forme [18] afin que leur transformée de Fourier aie la même résolution fréquentiel (voir equation (3.13)). La fonction matlab *extractionKernels.m* [19] s'occupe de la séparation et la mise en forme de ces noyaux.

4.3.3.2 Reconstruire les différents noyaux de Volterra

En passant dans le domaine fréquentiel et en se servant de l'équation (3.13), nous pouvons reconstituer les différents noyaux de Volterra à condition de pouvoir estimer les paramètres A [20] et B [21]. Pour un Logsweep de durée T=20 secondes , de pulsation initiale et finale $w_1 = 2\pi.20$ et $w_2 = 2\pi.20000$ respectivement, le paramètre B vaut :

17. Chaque noyau est écrit dans un vecteur.
18. Chaque vecteur doit être complété de zéro pour que les dimensions des vecteurs soient identiques.
19. voir section 8.1 page 92
20. A est un paramètre liant l'amplitude du sweep à l'amplitude des noyaux h'_i (voir éq. (3.12)).
21. Voir équation (10.6).

$$B = \frac{Tw_1}{ln(\frac{w_2}{w_1})} = \frac{20 * 20 * 2\pi}{ln(\frac{20000}{20})} = 363.83$$

Estimation du paramètre d'amplitude A

L'idée pour approximer le facteur A est la suivante (voir figure 4.20) :

1. Envoyons au haut-parleur un sinus d'amplitude 31 Vrms et de fréquence 100 Hz et enregistrons le signal z(t) émis par le haut-parleur.
2. Envoyons un sinus $x(t) = 31 * (\sqrt{2})sin(2\pi 100t)$ à notre simulation [22] et enregistrons le signal y(t) qui en sort.
3. Comparons la transformée de Fourier des signaux z(t) et y(t) (voir figure 4.19)
4. Si le facteur A est bien choisi, alors les deux systèmes sont équivalents et l'amplitude relative entre les harmoniques et la fondamentale est la même pour les 2 systèmes.
5. Si les amplitudes ne sont pas les mêmes alors il faut recommencer la reconstruction des noyaux de Volterra avec un autre facteur A.
6. Le facteur A qui mène à une même différence d'amplitude entre la fondamentale et l'harmonique 2 (flèche verte [23].) ainsi qu'à une différence d'amplitude entre la fondamentale et l'harmonique 3 (flèche rouge), identique pour les 2 systèmes, a été calculée de manière itérative et vaut A=31.7 .

22. Initialement, les noyaux de Volterra sont reconstruits avec une facteur A=1.
23. Voir figure 4.19

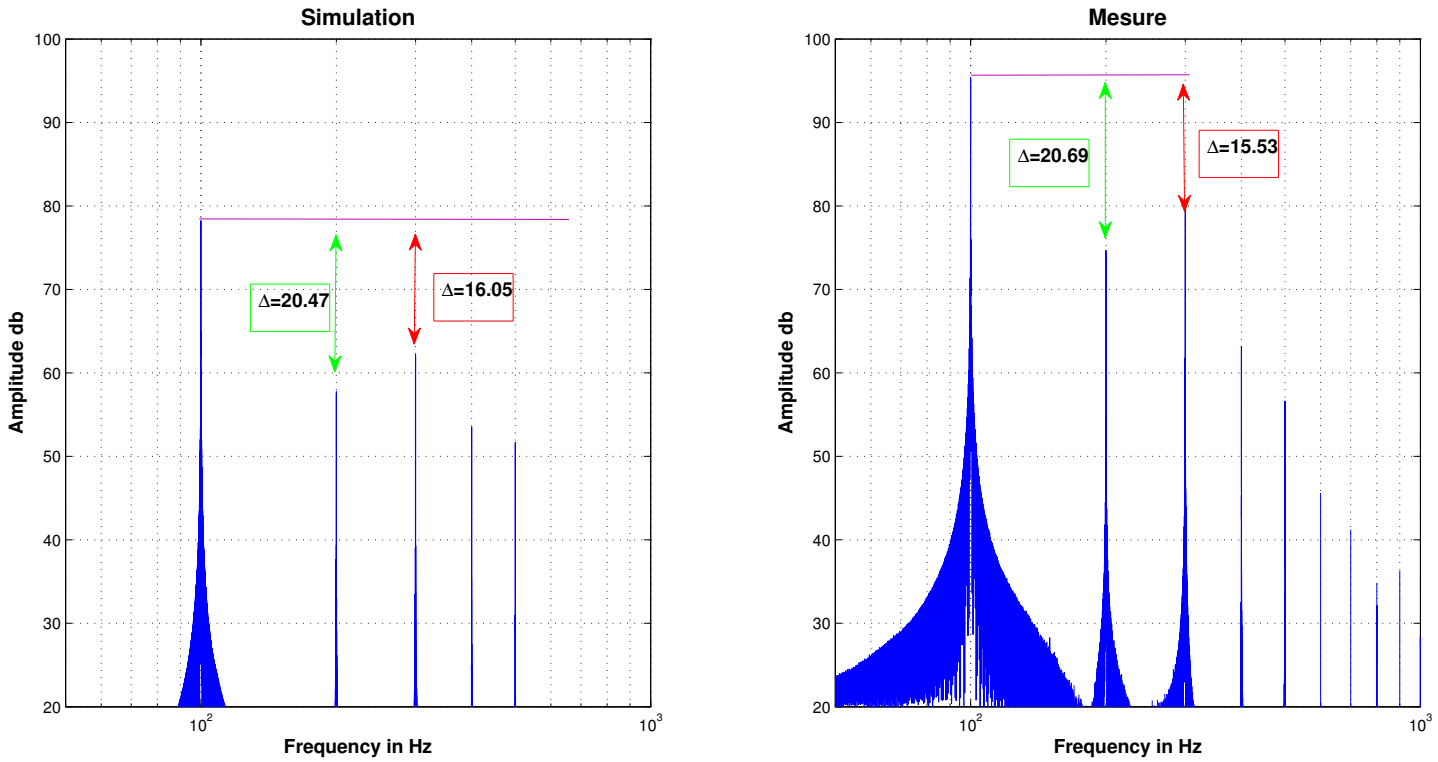


Figure 4.19 – Comparaison des transformées de Fourier des signaux y(t) (simulation) et z(t) (mesure)

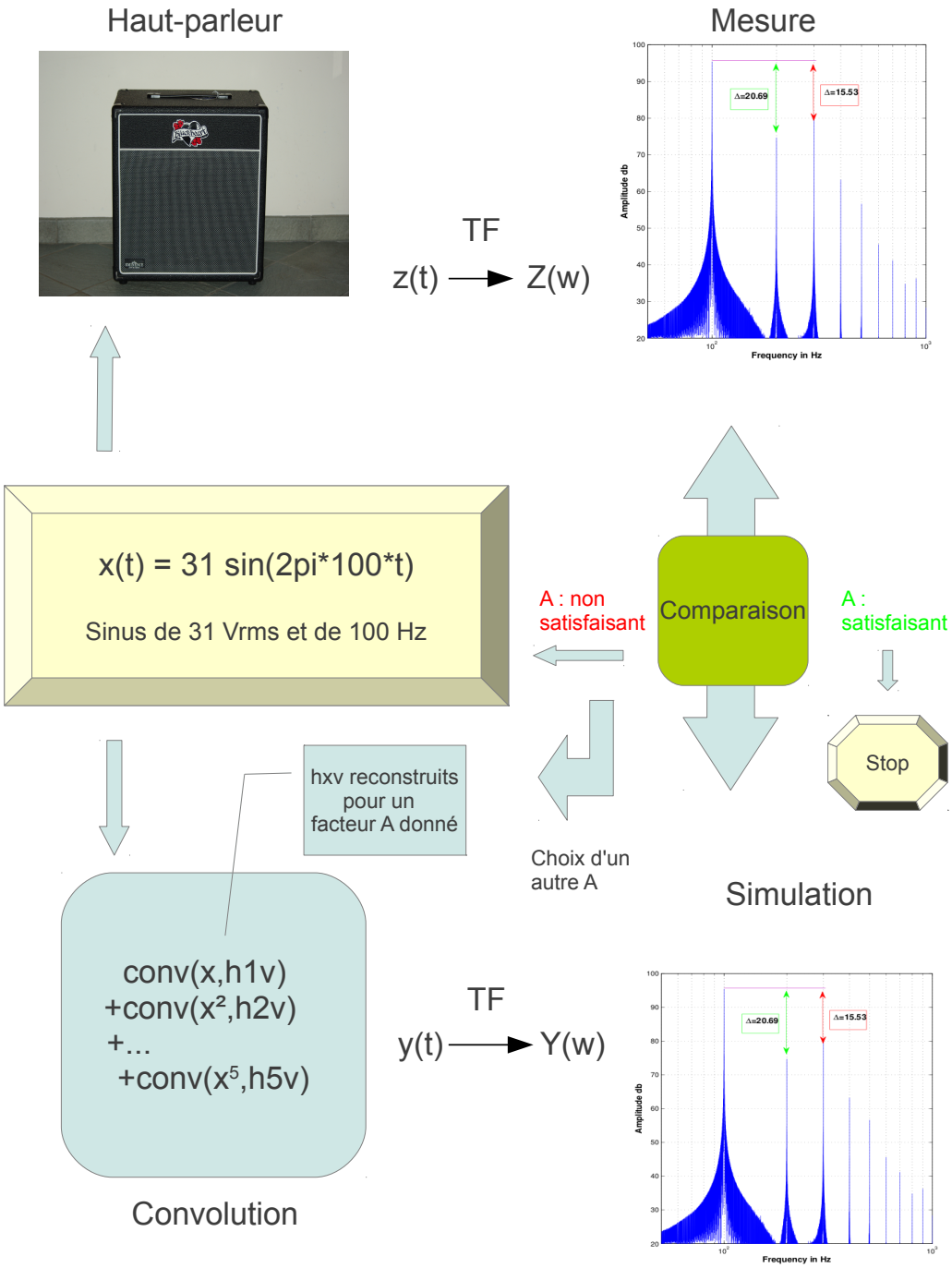


Figure 4.20 – Algorithme pour la détermination du facteur A

4.3.3.3 Comparaison de la réponse linéaire reconstruite H1v et de la réponse bas niveau Hlin2V mesurée via Lin sweep

Rappels sur les notations

- Hxv représente la transformée de Fourier du $x^{ième}$ noyau de Volterra.
- HlinxV représente la transformée de Fourier de la réponse impulsionnelle du haut-parleur obtenue en déconvoluant le signal de sortie du système lorsqu'un sweep linéaire de x Volts est appliqué en entrée.
- Cas particuliers : H1v est la réponse linéaire de Volterra, Hlin2V et Hlin31V sont les réponses fréquentielles à bas et haut niveau [24] respectivement.

La figure 4.21 compare H1v la réponse linéaire reconstruite à l'aide des noyaux de Volterra et Hlin2V . Une légère différence est perceptible en basse fréquence ainsi qu'aux alentours de 1700 hertz. Hormis ces détails, la reconstruction est satisfaisante.

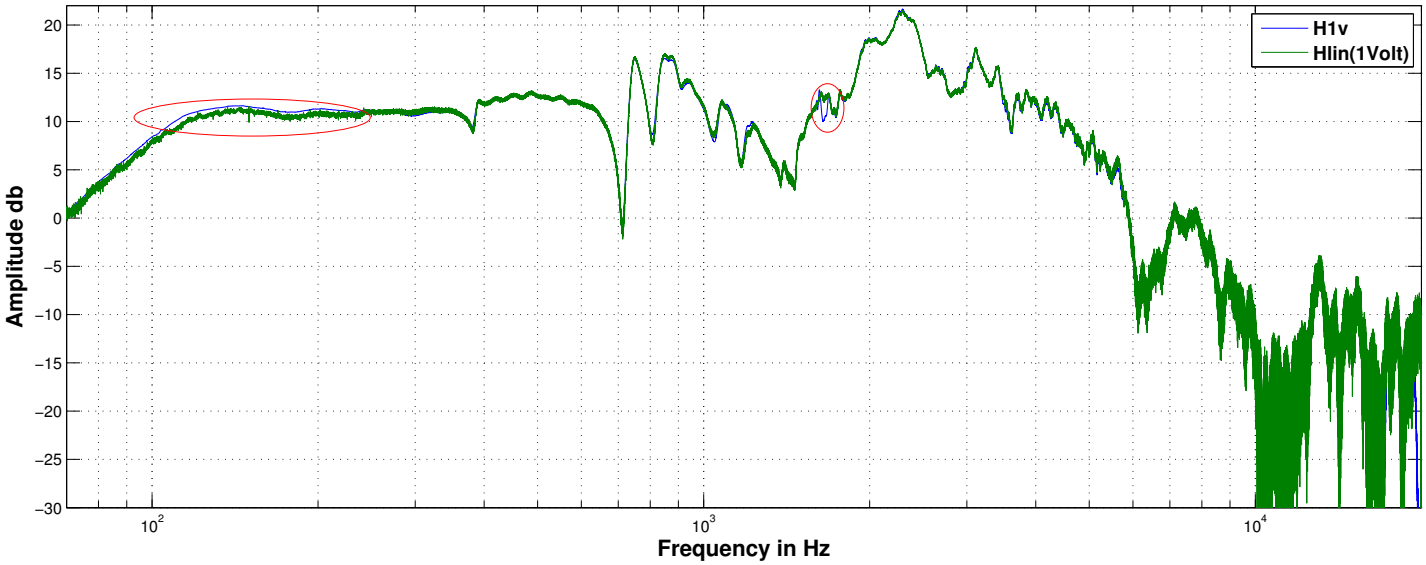


Figure 4.21 – Comparaison de : (1) La réponse fréquentielle reconstruite à bas niveau H1v via les noyaux de Volterra. (2) La transformée de Fourier de la réponse impulsionnelle du haut-parleur obtenue en déconvoluant la sortie du système lorsqu'on applique un sweep linéaire de bas niveau à son entrée.

4.3.3.4 Comparaison de la réponse à haut niveau reconstruite avec la réponse haut niveau Hlin31V mesurée via Lin sweep

La figure 4.22 compare la réponse totale du système [25] reconstruite à l'aide des noyaux de Volterra, la réponse H1v (réponse linéaire reconstruite) et la réponse Hlin30V obtenue par un Linsweep à haut niveau. Entre 100 et 200 hertz la courbe totale Hvtot est un peu

24. section 4.3.2 page 46
25. (c'est a dire Hvtot= H1v+H2v+...+H5v)

plus élevée que la réponse linéaire de Volterra H1v. La différence est peu flagrante et il est difficile de dire qu'il y a là une amélioration. Par contre, la réponse mesurée par log sweep est nettement moins bruitée dans la bande passante.

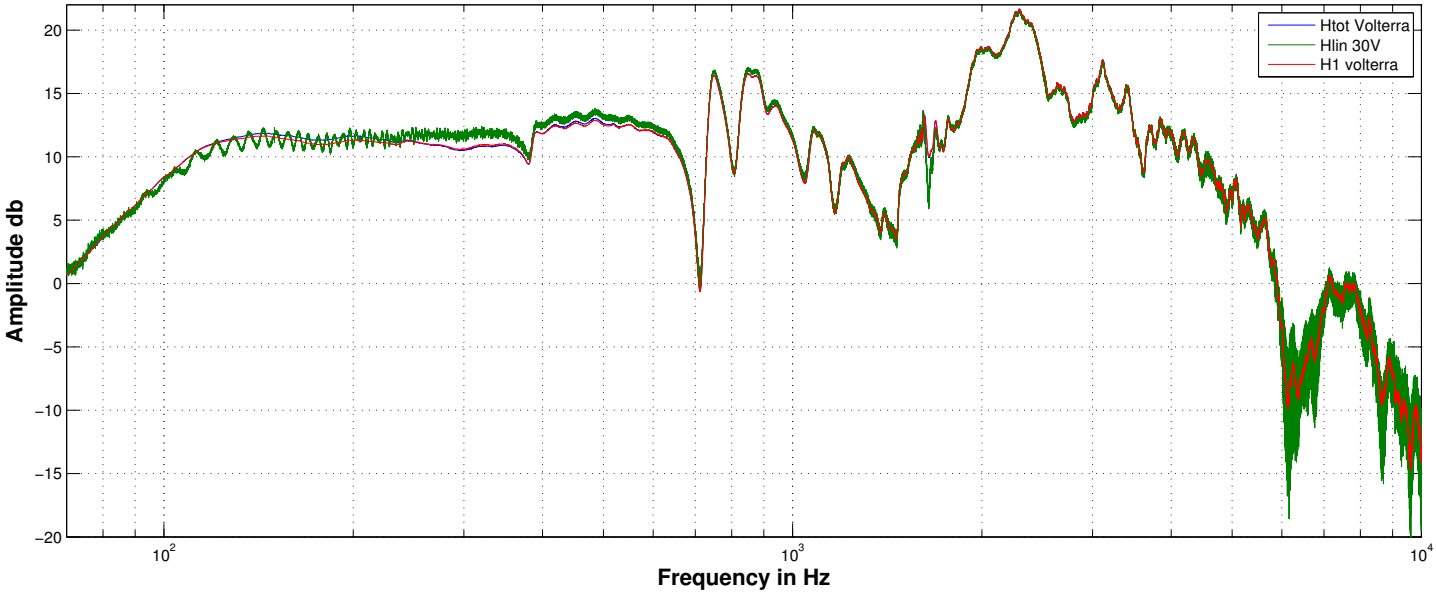


Figure 4.22 – Comparaison de : (1) la réponse fréquentielle reconstruite à haut (Hvtot) et bas niveau (H1v) via les noyaux de Volterra. (2) La transformée de Fourier de la réponse impulsionnelle du haut-parleur obtenue en déconvoluant la sortie du système lorsqu'on applique un sweep linéaire de haut niveau à son entrée.

La reconstruction de Volterra améliore-t-elle la fidélité du système simulé ? il est très difficile d'apporter une réponse en comparant les courbes car les variations entre H1v (réponse linéaire de Volterra) et Htotv sont très faibles. Par conséquent, on ne peut affirmer avec certitude que la courbe Htot est plus proche de Hlin31V que H1v ne l'est.

De plus, la variabilité de la mesure (prise de réponse impulsionnelle) face à la température de la bobine du haut-parleur rend encore plus délicate la distinction de si petites variations.

En effet, à ces niveaux de puissance, l'échauffement de la bobine joue un rôle prépondérant sur la réponse du haut-parleur, la figure 4.23 nous montre l'amplitude du signal enregistré par le micro lorsque nous envoyons au haut-parleur une sinusoïde à 1000 hertz de 30 Vrms pendant 25 secondes.

Cette variation d'amplitude est fortement dépendante de la fréquence (l'effet est assez marqué à 1000 hertz mais il est pratiquement inexistant à 100 hertz). Une façon de se défaire de ce problème serait de générer 2 sweeps consécutifs et de ne prendre en compte que le second lorsque le régime peut être considéré comme établi vis a vis de la température.

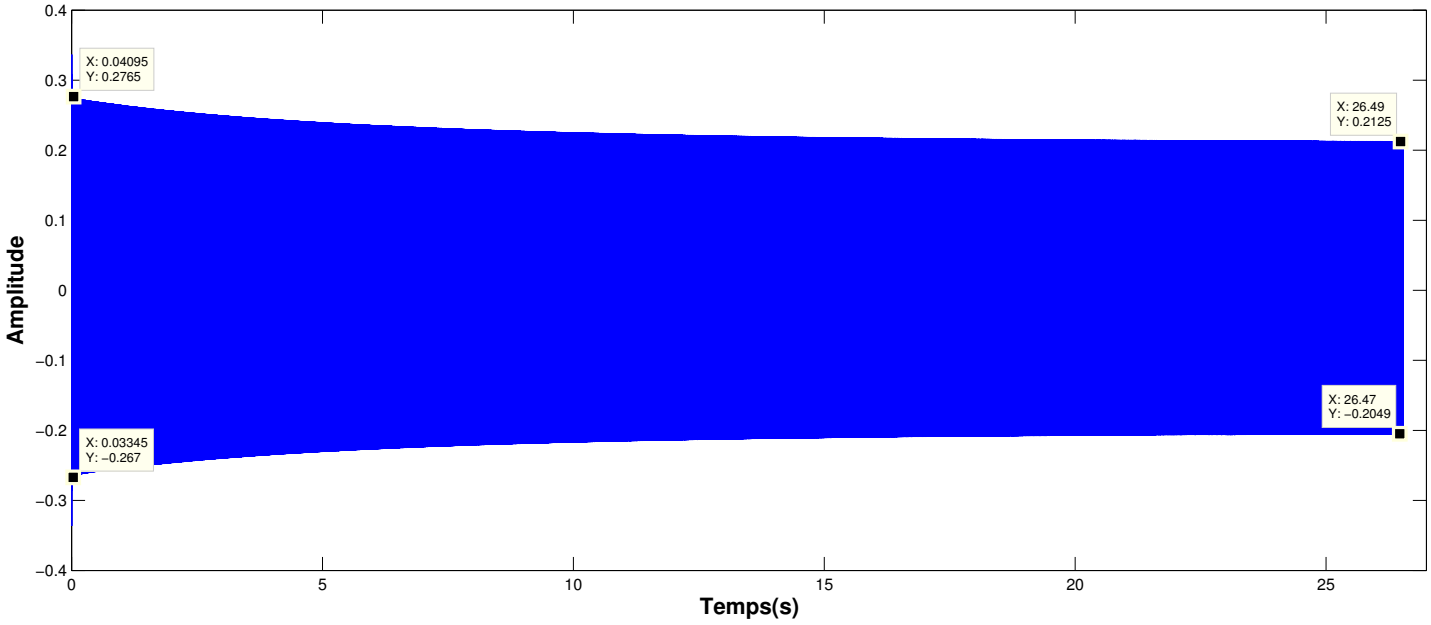


Figure 4.23 – Diminution de l'amplitude du signal récupéré par le microphone dans le temps (probablement du à l'échauffement de la bobine) pour un sinus à 1000hertz et d'amplitude 31Vrms

Nous comprenons aisément pourquoi il peut y avoir des différences entre nos différentes mesures de réponses impulsionnelles successives :

1. La température initiale de la bobine dépend du temps entre 2 mesures successives
   (a) t_0 = température ambiante
   (b) mesure 1 $\Rightarrow$ échauffement bobine $\Rightarrow$ température $t_1 > t_0$
   (c) préparation de la deuxième mesure $\Rightarrow$ refroidissement de la bobine $\Rightarrow$ température $t_2 < t_1$
   (d) mesure 2 avec condition initiale de température $t_2 \neq t_0$
2. Les mesures via Log et Linsweep sont difficiles à comparer. En effet, la température de la bobine est une fonction du temps T(t) et la fréquence instantanée des sweeps logarithmique et linéaire sont deux fonctions du temps différentes. Par conséquent l'effet de la température de la bobine ne se manifeste pas pour les mêmes fréquences [26] dans le cas d'un sweep linéaire ou dans le cas d'un sweep logarithmique.

26. voir section 4.2.1.2 page 42

4.3.3.5 Note :

Un test réalisé sur un système moins linéaire que notre haut-parleur nous permettrait de nous assurer que le système de Volterra apporte bien un avantage à la modélisation, les courbes seraient alors plus faciles à distinguer.

4.3.4 Haut-parleur Eminence Wizard

Recommençons les tests pour un second haut-parleur. Comme nous le verrons, la reconstruction ne se passe pas aussi bien que pour le haut-parleur Blackheart. L'ampli utilisé était le Pioneer A-307, le niveau *haut* est plafonné à 25V pour éviter les problèmes de saturation de l'ampli observés à l'oscilloscope lors de sweep à 31 V_{rms}.

4.3.4.1 Réponse impulsionnelle h(t) via sweep logarithmique

Comme le montre la figure 4.24, la réponse impulsionnelle du haut-parleur Wizard16 possède des noyaux mesurés h'_i d'amplitude plus importante que la réponse impulsionnelle du cab blackheart (figure 4.13). Et ce, malgré le fait que le cab blackheart est soumis à une puissance électrique supérieur au cabinet contenant le haut-parleur Wizard16 (31V contre 25V, soit 60wrms pour le Blackheart contre 40 Wrms pour le Wizard16).

Des répliques apparaissent le long de la réponse impulsionnelle (entourées en rouge). En accord avec la théorie de la section 3.2, les 2 premières répliques situées de part et d'autre de la réponse dite linéaire h'1 correspondent probablement aux contributions γ_{20} et γ_{22} des équations (3.15) et (3.17) respectivement.

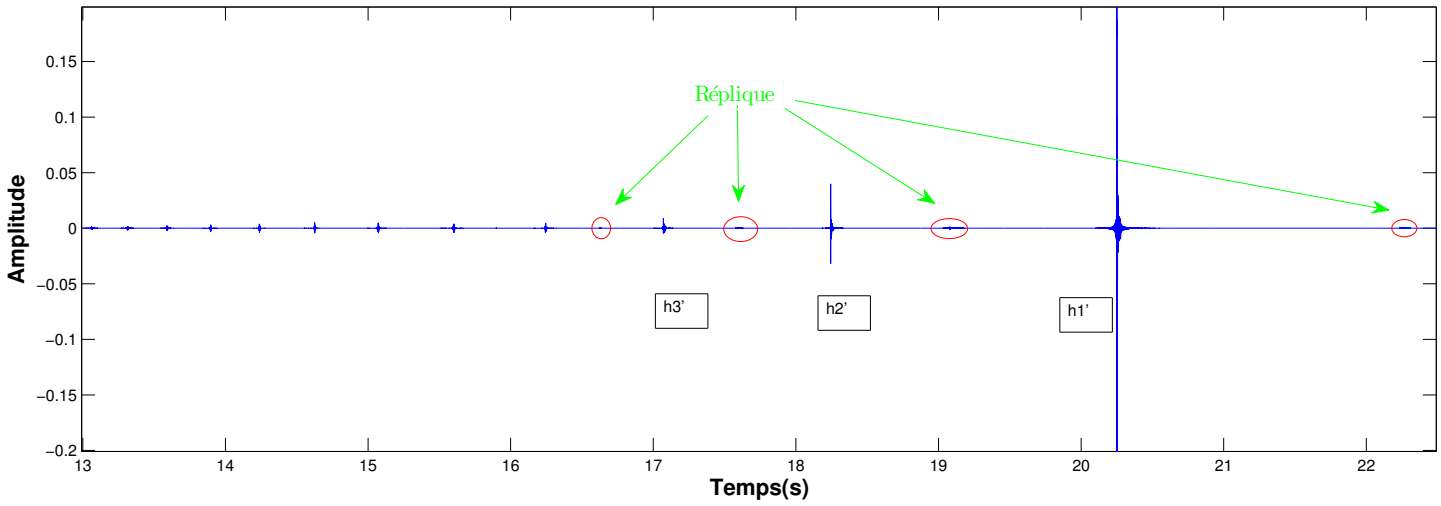


Figure 4.24 – Réponse impulsionnelle du haut-parleur *Eminence Wizard* obtenue en déconvoluant un sweep logarithmique de 25 *Vrms* appliqué à son entrée.

4.3.4.2 Comparaison des réponses à bas et haut niveau (Hlin2V et Hlin25V)

La figure 4.25 montre des différences allant de 1 à 5db entre les réponses fréquentielles à 2V et à 25V prises à l'aide d'un sweep linéaire. Nous observons aussi que l'amplitude du 50 hertz (parasites) et de ses harmoniques sont plus importantes à bas niveau qu'à haut niveau (si le parasite est indépendant du niveau généré, alors son amplitude relative est plus importante dans le cas d'un sweep à bas niveau).

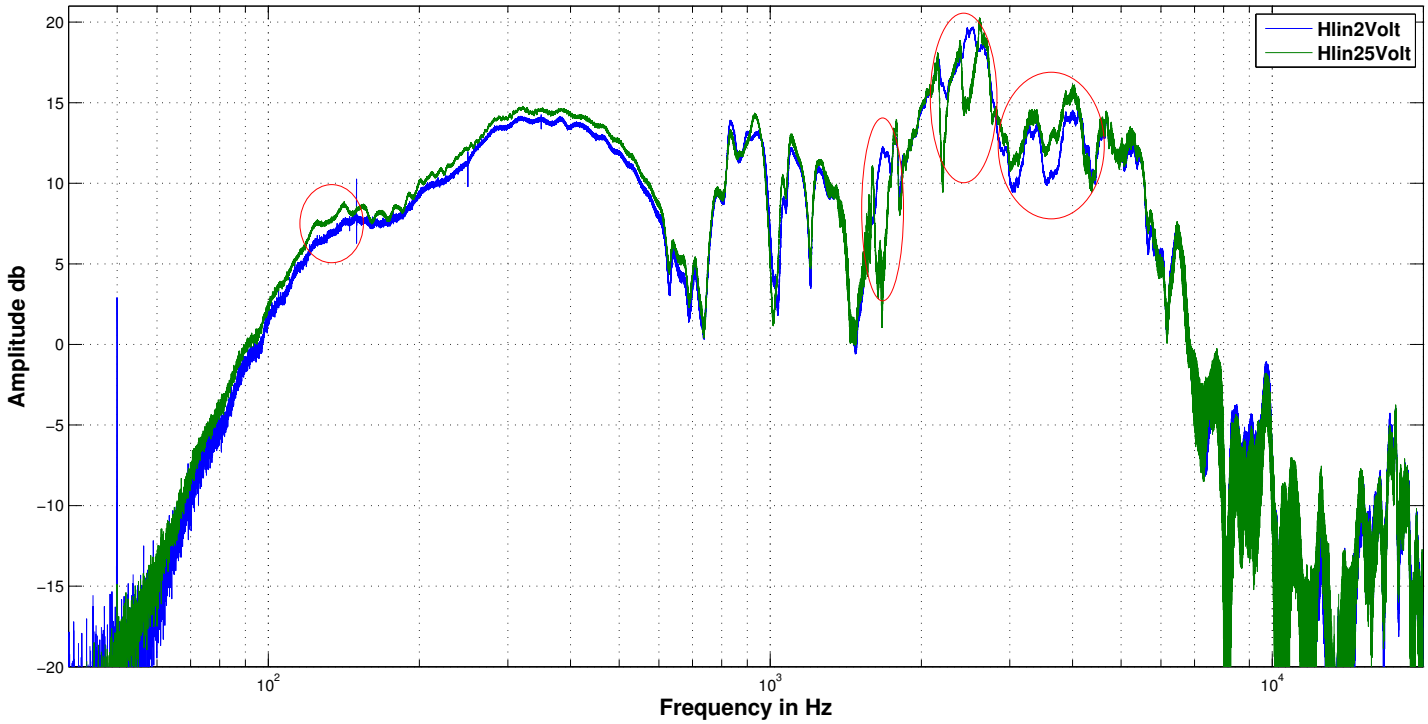


Figure 4.25 – Transformées de Fourier des réponses impulsionnelles du haut-parleur *Eminence Wizard* obtenues en déconvoluant les sweeps linéaires de 2 et 25 V appliqués à son entrée.

4.3.4.3 Comparaison des réponses linéaires H1v et Hlin2V

La figure 4.26 compare la réponse linéaire du système reconstruite à l'aide des noyaux de Volterra avec la réponse fréquentielle Hlin2V obtenue par transformée de Fourier de la réponse impulsionnelle du haut-parleur lorsqu'un sweep linéaire à bas niveau (2V) est appliqué en entrée.

Nous remarquons que :

- Les courbes sont assez éloignées l'une de l'autre entre 150 et 300 hertz et entre 2000 et 4000 Hz.

- La différence en basse fréquence (flèche rouge) est un problème lié à la différence d'effet que produit l'opération de fenêtrage sur un sweep logarithmique et sur un sweep linéaire. [27]

En conclusion, la reconstruction n'est pas aussi bonne que pour le cab Blackheart, ce qui prouve qu'il reste des points obscurs dans la reconstruction du système linéaire.

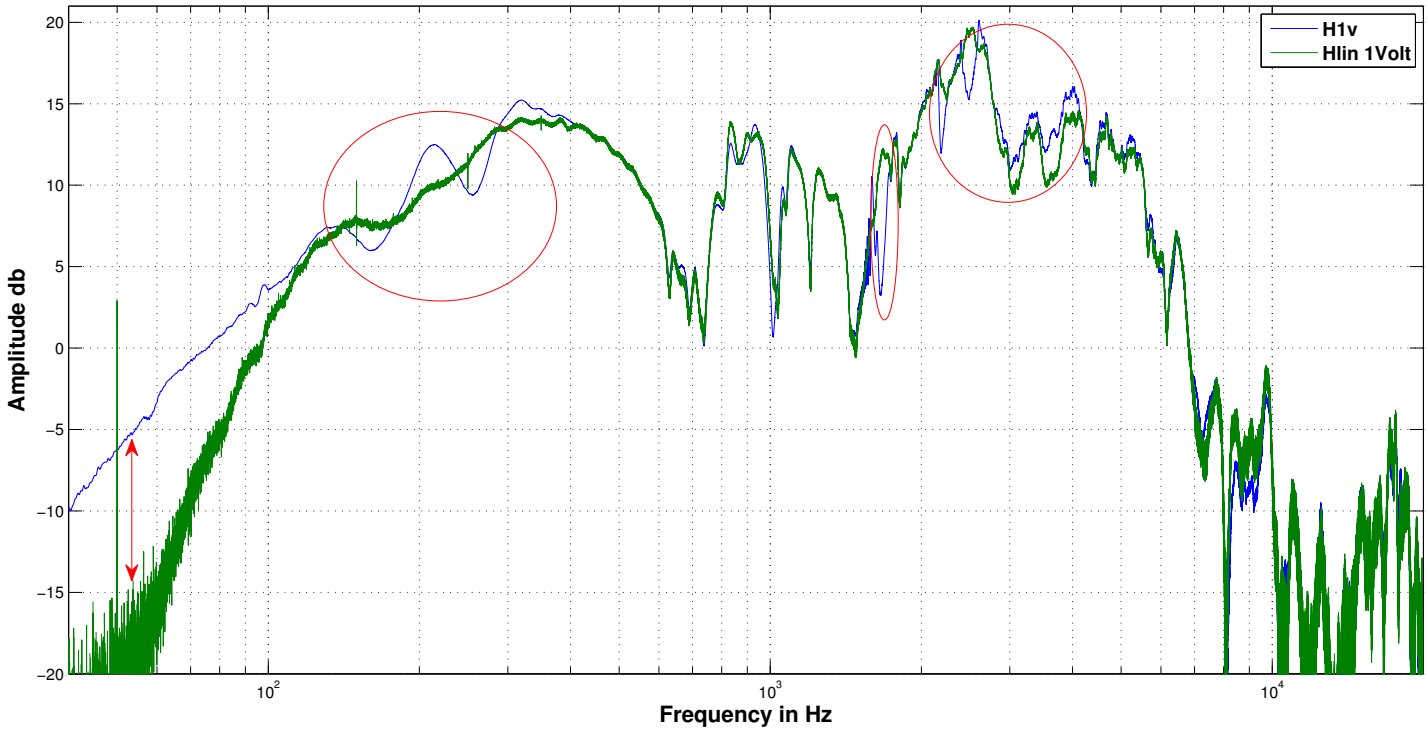


Figure 4.26 – Comparaison des réponses fréquentielles du haut-parleur *Eminence Wizard* : H1v et Hlin2V

4.3.4.4 Comparaison des réponses à hauts niveaux Htotv et Hlin25V

Comparons la réponse Htotv (réponse reconstruite pour un haut niveau à l'aide des noyaux de Volterra) avec la réponse Hlin25V (transformée de Fourier de la réponse impulsionnelle du haut-parleur lorsqu'un sweep linéaire de 25Vrms est appliqué à son entrée).

Comme pour le haut-parleur Blackheart, il faut trouver le facteur d'amplitude A [28].

Plutôt que de d'appliquer la méthode développée à la section 4.3.3.2, nous chercherons le facteur A qui minimise la "distance" entre la courbe Htotv et Hlin25V (approximation au sens des moindres carrés). Le facteur A ainsi obtenu vaut 1.

La figure 4.27 nous présente la comparaison de :

- H1v la réponse fréquentielle linéaire de Volterra
- Htotv la réponse de Volterra obtenue en additionnant les réponses en fréquences de tous les noyaux hxv

27. Voir section 4.2.1.1 page 39
28. voir section 4.3.3.2 page 54

– Hlin25V la transformée de Fourier de la réponse impulsionnelle du haut-parleur lorsqu'un sweep linéaire de 25Vrms est appliqué à son entrée.

Comme nous le constatons, la réponse Htotv s'approche mieux de Hlin25V dans la bande de fréquence [150 300] hertz, au détriment des fréquences situées aux alentours de 100 hertz.

Nous comprenons qu'un des inconvénients de procéder à une recherche du facteur d'amplitude A en minimisant la distance entre les courbes au sens des moindres carrés est que le facteur A optimal trouvé sert plus à corriger l'erreur faite sur la reconstruction de la réponse linéaire qu'à compenser les différences qui existe entre les réponses fréquentielles à bas et haut niveaux.

La figure 4.28 montre les transformées de Fourier des différents noyaux de Volterra reconstruits avec un facteur A=1. Pour certaines fréquences, les réponses fréquentielles des noyaux d'ordre 2 et plus sont supérieures à la réponse fréquentielle linéaire.

Finalement, remarquons que les pics de distorsions se situent :

– pour H2v $\Rightarrow$ dans la plage comprise entre 100 et 200 hertz, soit une fondamentale comprise entre 50 et 100 Hz.
– pour H3v $\Rightarrow$ dans la plage comprise entre 200 300 hertz, soit une fondamentale comprise entre 70 et 100 Hz.

Pour les instrumentistes, cela veut dire que les fréquences de 80 à 100 hertz donnent lieu à beaucoup de distorsion au niveau du haut parleur. (80 hertz est la fréquence la plus basse de la guitare).

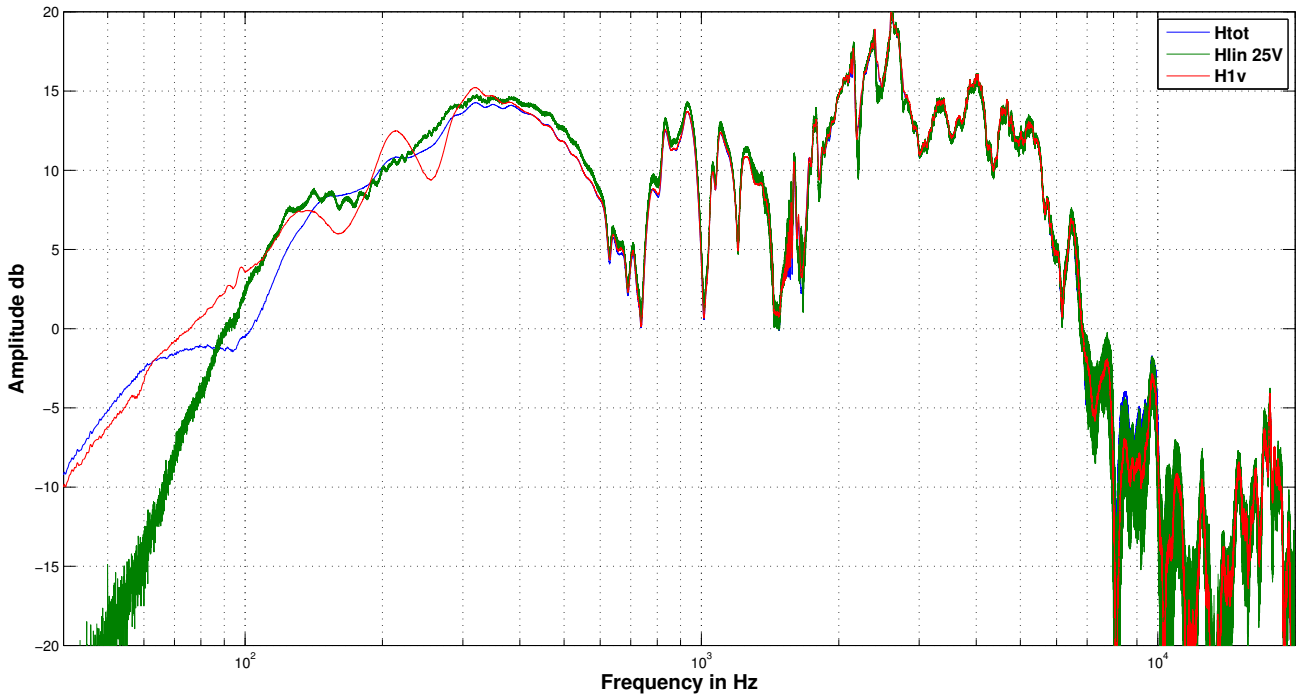


Figure 4.27 – Comparaison des réponses fréquentielles : H1v,Htotv et Hlin25V

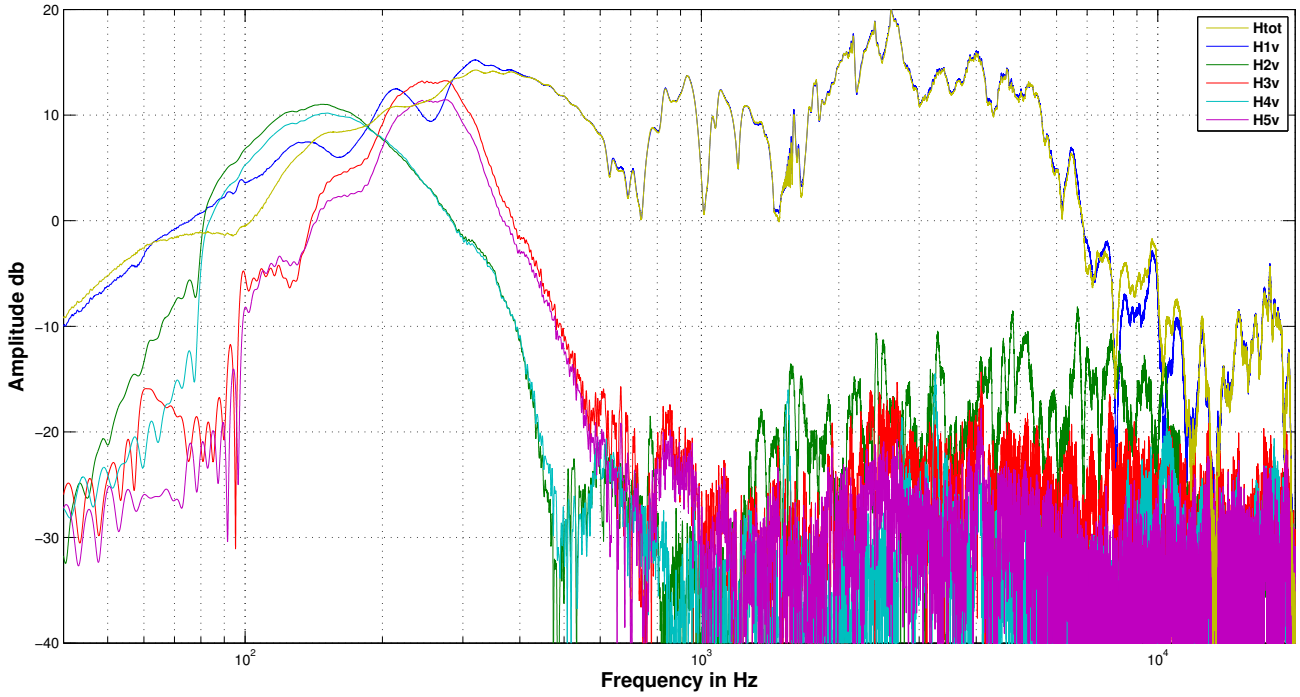


Figure 4.28 – Réponses fréquentielles des différents noyaux de Volterra

5 Résultat audio

5.1 Système d'écoute

5.1.1 Loadbox

La loadbox est un appareil conçu pour adapter le signal de grande amplitude sortant de l'ampli (60 Wrms = 31 Vrms sur une charge de 16 ohm) vers un signal de faible amplitude (niveau ligne 1 à 2 Vrms). Grâce à cet appareil nous récupérerons le signal électrique en sortie de l'ampli et le ramènerons à la carte son pour le traiter via un plugin *simulateur de haut-parleur*. Le schéma électrique de la loadbox est présenté à la figure 5.1.

Quelques mots sur le schéma :

- J1 est l'entrée à connecter en parallèle du haut-parleur.
- La réduction du signal se fait via le potentiomètre P1 de 10k, la plage d'atténuation disponible s'étend de -9db à -31db, soit un gain compris entre $\frac{1}{3}$ et $\frac{1}{30}$ environ .
- Les diodes connectées aux bornes plus et moins de l'alimentation protègent la carte son d'une éventuelle surtension en limitant la tension de sortie du montage. La tension maximale vaut : (6-0.7)/$\sqrt{2}$=3.74 Vrms (où 0.7V représente la tension de seuil de la diode).
- U2 :B et U2 :C sont des inverseurs de phase (si le switch *tp1* est passant, sinon seul U2 :C est un inverseur de phase). Cela permet d'inverser la phase du signal de sortie de 180° suivant que le switch *tp1* est On ou Off.
- J2 et J3 sont 2 sorties symétriques envoyant un signal de niveau ligne (seul une des sorties sera utilisée dans notre cas).
- U2 :D est un redresseur de signal si bien que la tension aux bornes du condensateur C14 est fonction du niveau de crête du signal entrant.
- Le *Lm3915* est une puce *bar-graph* qui permet d'allumer une led supplémentaire à chaque augmentation de 3db du niveau du signal d'entrée. La led verte (D7) s'allume pour signaler un niveau de sortie de 0dbV.
- Le système est alimenté en 220 V et possède sa propre alimentation +-6V régulée via lm317 et lm337.

– Les connectique face avant et face arrière sont décrites à la figure 5.2 et 5.3 respectivement.

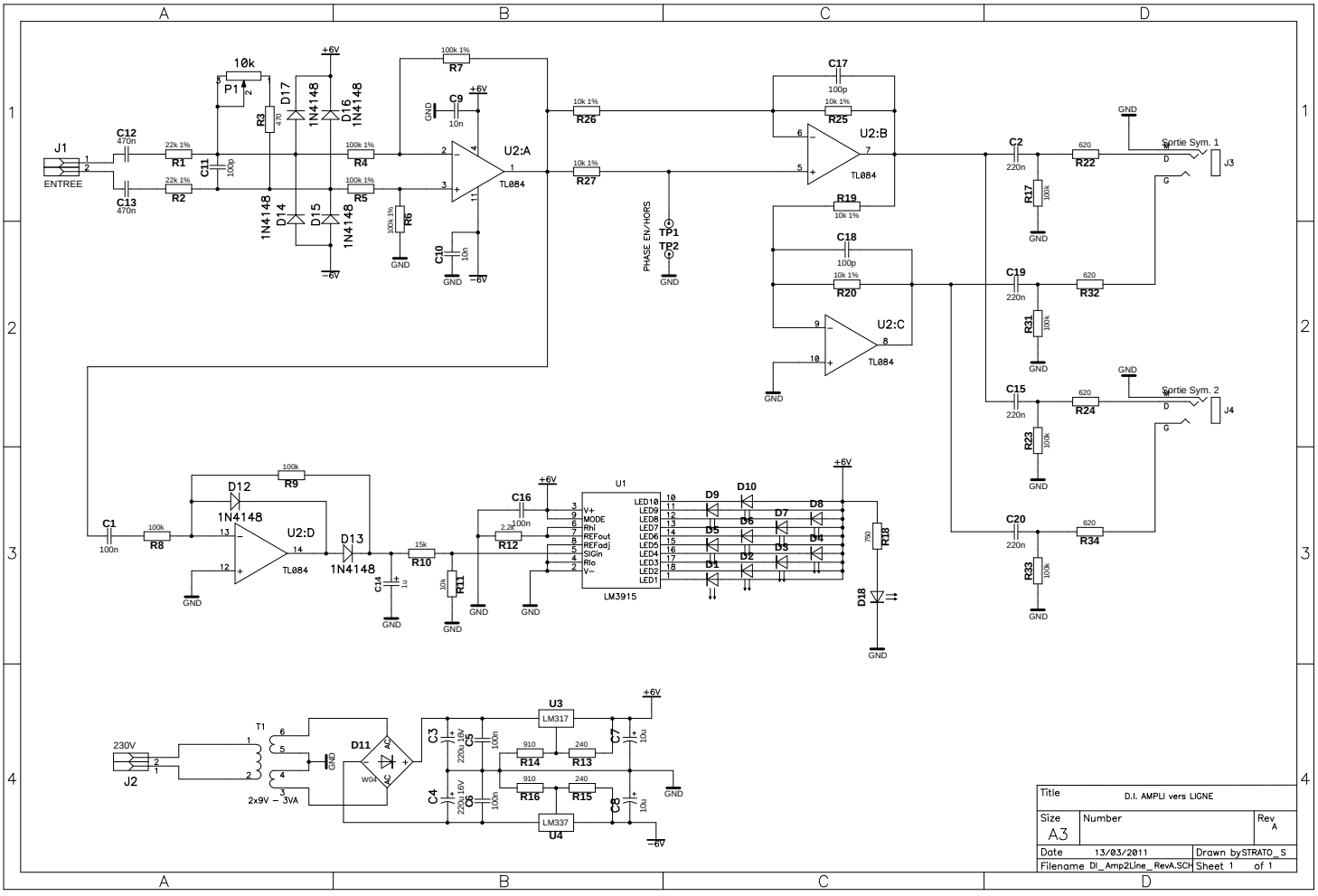


Figure 5.1 – Loadbox, schéma original de Claude Cavallo

Figure 5.2 – Loadbox face avant : volume, switch inverseur de phase et les deux sorties jacks en symétriques

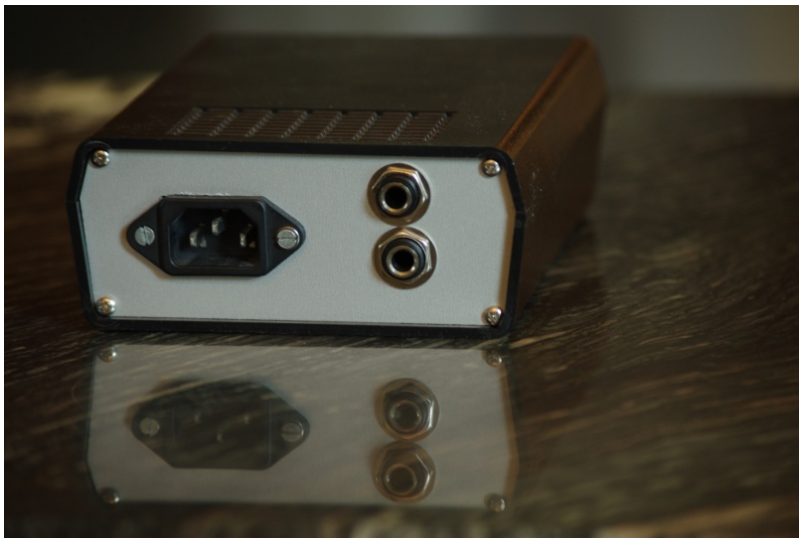


Figure 5.3 – Loadbox face arrière : connecteur 220VAc, les 2 prises jacks sont montées en parallèles et sont à raccorder à la sortie de l'ampli et au haut-parleur (ou à une charge équivalente).

5.1.2 Réponse en fréquence de la Loadbox

La figure 5.4 présente la réponse fréquentielle de la loadbox, la fréquence de coupure basse est de 47hz. Nous remarquons une atténuation en hautes fréquences due à la bande passante non idéale de l'ampli utilisé (voir la figure 11.3). Nous sommes aussi confronté à un problème de parasites à 50 hz et ses multiples.

Pour s'assurer que la bande passante de la loadbox est indépendante de la charge connectée en entrée, nous avons remplacé le haut-parleur par une charge résistive de 16 ohms. La figure 5.5 nous présente la réponse fréquentielle de la loadbox chargée par

cette résistance, nous n'observons pas de changement dans la bande passante.

Remarquons cependant que les parasites à 50 Hz et ses multiples sont amplifiés dans le cas où le haut parleur est substitué au profit de la résistance de 16 ohms . Ces parasites proviennent vraisemblablement d'un couplage capacitif ou inductif puisque :

1. La tension d'alimentation est bien régulée (ondulations résiduelles $<$ mV), de plus l'ampli op utilisé (TL072) est alimenté en symétrique et le montage possède une entrée différentielle, son PSRR (power supply rejection ratio) et son CMRR (common mode rejection ratio) sont tous deux supérieurs à 70 db. [28]
2. L'amplitude des parasites dépend du cable utilisé pour les 2 mesures.
   - cable coaxial pour le haut-parleur situé dans la chambre anéchoïque,
   - fiches bananes et cable hp pour la résistance de 16 ohms située en dehors de la chambre anéchoïque.

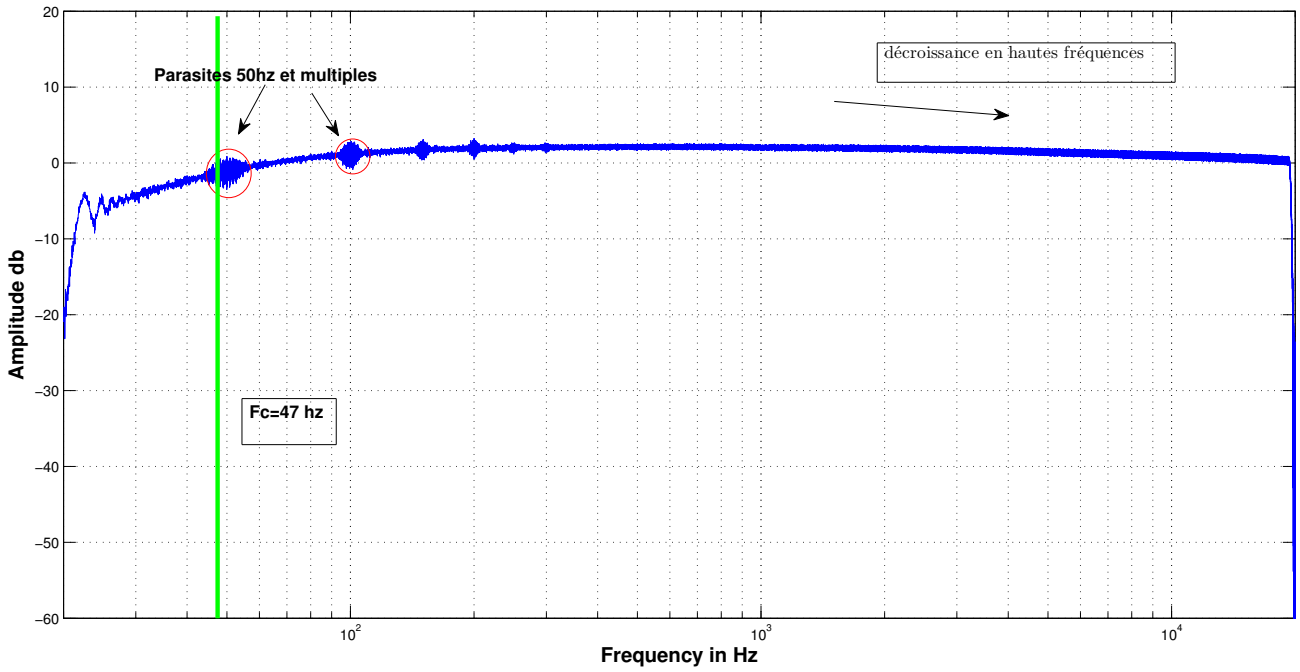


Figure 5.4 – Réponse en fréquence de la loadbox connectée au haut-parleur (sweep amplifié avec ampli Kenwood).

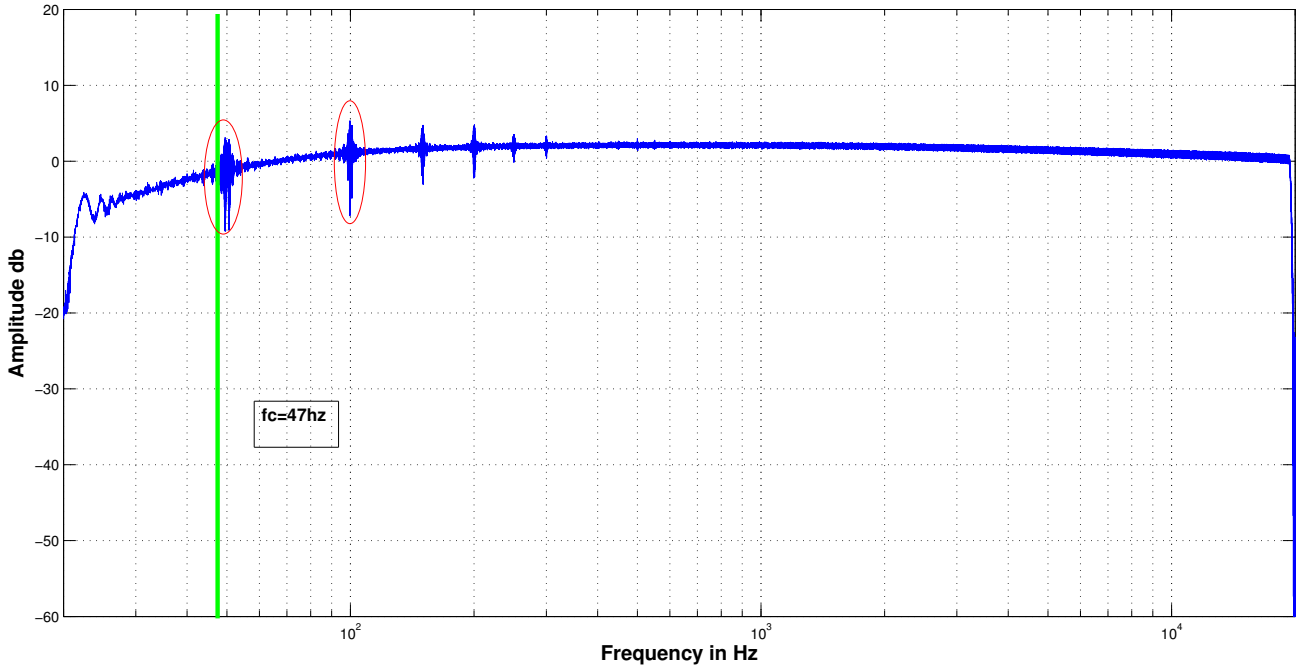


Figure 5.5 – Réponse en fréquence de la loadbox connectée à une charge résistive de 16 ohms (sweep amplifié avec ampli Kenwood).

5.1.3 Carte son 2 [27]

Nous utilisons une deuxième carte son USB (figure 5.6 et 5.7) pour les tests audio. Cette carte son est basée sur *l'application note* d'une puce de chez Texas Instruments , le pcm2912 (voir figure 5.8). La bande passante est présentée à la figure 5.9, la fréquence de coupure basse est de 146 hz.

Ce problème de fréquence de coupure trop haute vient très probablement d'une sous-estimation de ma part du condensateur C5 [1].

Quelques spécifications :

- convertisseur A/D 16 bits
- fréquence d'échantillonnage maximale (fs_{max}) = 48khz
- +-0.1db de ripple dans la bande passante
- SNR 92 db
- fréquence de coupure haute 0.445*fs_{max}= 19600 Hz dans notre cas.

1. Ce problème devrait être résolu pour le jour de la démonstration.

Figure 5.6 – Carte son 2 à base de pcm2912

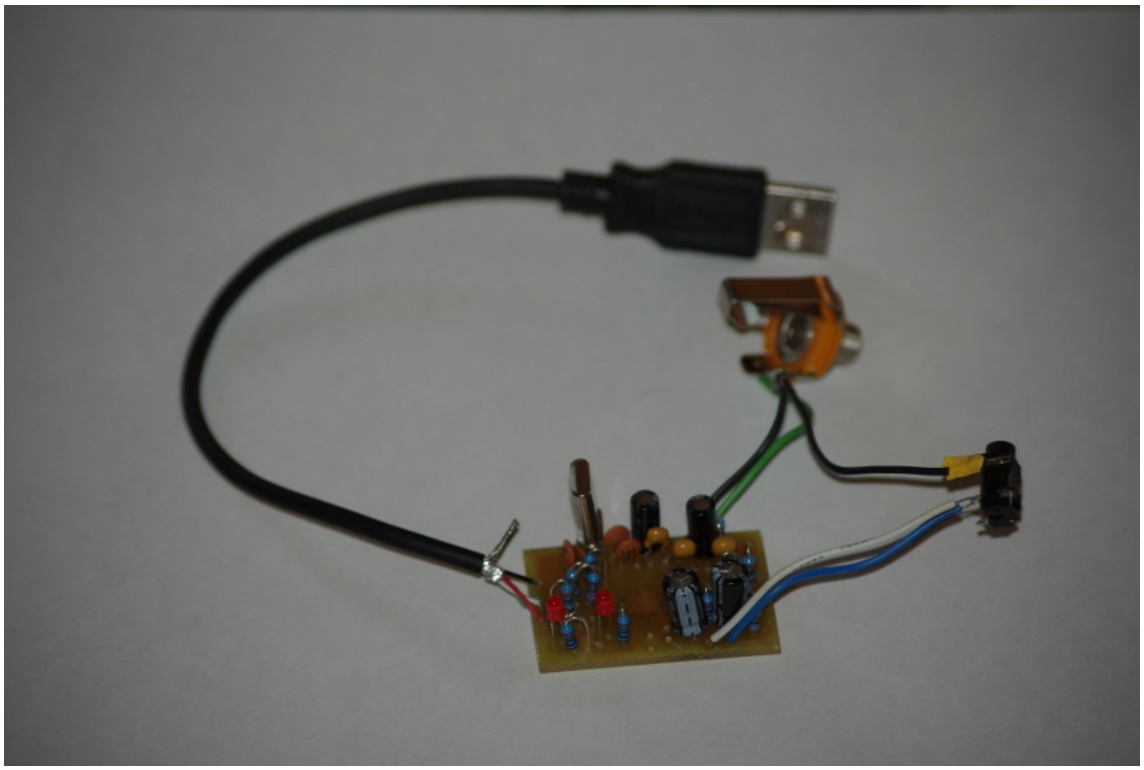


Figure 5.7 – Circuit de la carte son 2

TYPICAL CIRCUIT CONNECTION

A bus-powered (Hi-power), +20-dB microphone amplifier application example follows.

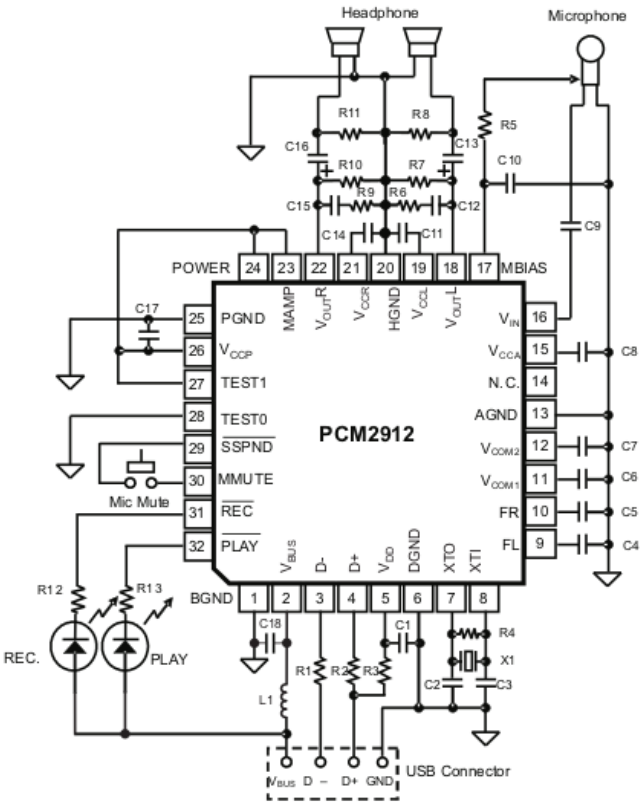


Figure 5.8 – Schéma électrique de la carte son 2

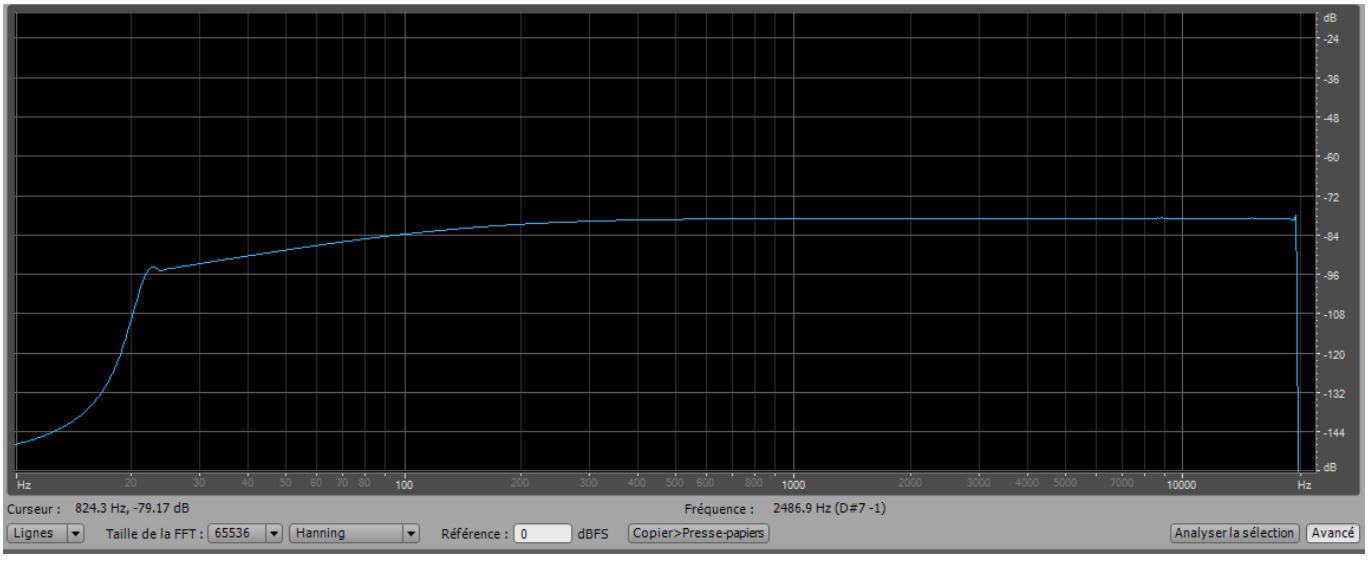


Figure 5.9 – Réponse en fréquence de la carte son 2

5.1.4 Amplificateur guitare

L'amplificateur guitare utilisé est le SIYH V2 (fig 5.10) , c'est un amplificateur à transistor d'une puissance de 50Wrms, l'entrée guitare se situe en façade avant.

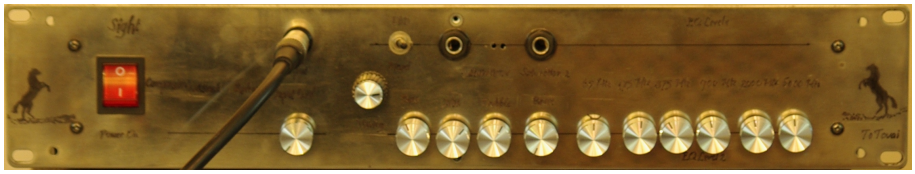


Figure 5.10 – Amplificateur guitare 50W

5.1.5 Casque Audio

L'écoute des différents tests audio se fait à l'aide d'un casque Sennheiser HD25-1 [49]

- Bande passante : 16 Hz - 22 kHz
- THD (Total Harmonic Distortion) <0.3%

5.2 Schéma du système d'écoute

L'objectif final est de comparer le son provenant de notre simulation au son produit par le haut-parleur, le montage de la figure 5.11 permet de basculer d'un système d'écoute à l'autre en switchant d'un canal A à un canal B sur l'ampli Kenwood.

Quelques explications :

1. Le signal provient de la guitare en haut à gauche
2. Le signal part sur l'ampli de puissance du guitariste (ampli + préampli) comme il le ferait en temps normal.
3. A la sortie "haut-parleur" de l'ampli sont connectés 2 modules en parallèle :
   (a) Le haut-parleur
      i. Le son provenant du haut-parleur est repris par le micro *metravib* .
      ii. Le signal passe ensuite par le conditionneur de signaux *Opus*.
      iii. Il est finalement envoyé sur l'entrée auxiliaire A de l'amplificateur Hifi *Kenwood*.
   (b) La Loadbox.
      i. Après atténuation par la loadBox, le signal est envoyé à la carte son 2
      ii. Le son est alors traité numériquement par le plugin.
      iii. Il est ensuite redirigé vers l'entrée auxiliaire numéro B de l'amplificateur Kenwood.

4. Le signal est envoyé à une enceinte hifi (ou un casque) afin de pouvoir comparer le son direct (guitare => ampli => baffle => micro) au son simulé (guitare => ampli => Loadbox => plugin)

Ce système permet une bonne comparaison du son direct et du son simulé car le passage d'un système à l'autre s'effectue en moins d'une seconde.

Gardons cependant à l'esprit qu'aux non-idéalités de mesures et de reconstruction des noyaux de Volterra viennent s'ajouter les non-idéalités d'écoutes suivantes :

- Bande passante de la carte son numéro 2 réduite dans les basses fréquences.
- Bande passante de la loadbox (+ parasites).
- Une éventuelle différence de réponse en fréquence des canaux A et B de l'ampli Kenwood (non vérifié).

Cette liste n'est évidemment pas exhaustive, une liste des non-idéalités plus détaillée sera faite au chapitre 6.

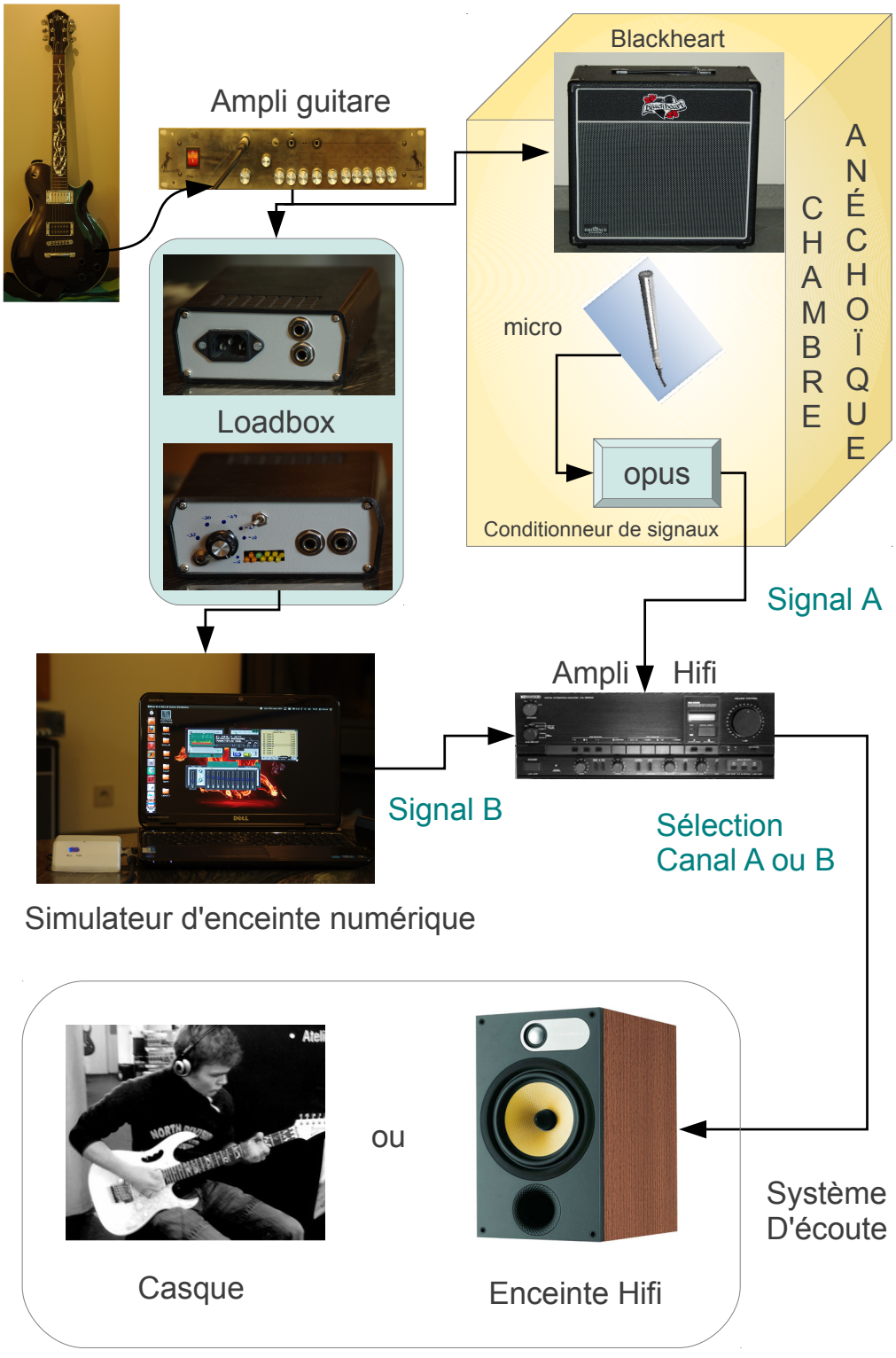


Figure 5.11 – Diagramme de fonctionnement des tests acoustiques.

5.3 Le plugin

5.3.1 Reaper

Reaper est un séquenceur multi-pistes gratuit et performant. Il permet d'enregistrer plusieurs pistes audio, de rejouer une piste lors de l'enregistrement d'une autre piste ou encore d'écouter une piste **lors** de son enregistrement. Reaper permet aussi d'appliquer des effets (sous forme de plugin) en temps réel [2] sur une piste audio. Nous avons développé un plugin pour Reaper convoluant le signal d'entrée avec les différents noyaux de Volterra.

5.3.2 Virtual Studio Technology

En 1996 un protocole audio à vu le jour, le VST pour Virtual Studio Technology, développé par la société Steinberg [51]. Les plugins VST sont des modules de conversion de donnée audio, les plugins sont sous la forme de fichier dll sous Windows et tourne sur un grand nombre de séquenceurs connus tel que :

- Reaper
- Ardour2
- Audacity
- Cubase
- Adobe audition
- ...

La dll est codée en C++ et devra être ajoutée dans le répertoire *fx* du répertoire *plugin* de Reaper, une seconde dll devra être placée dans le répertoire *Plugin* de Reaper. Cette seconde dll (libboost_thread-mgw46-mt-1_49.dll) est en fait une librairie de *Boost* [26] qui permet de créer un plugin multi-Thread.

Le détail de l'implémentation du plugin sera traité dans **l'Appendix B** [3], la figure 5.12 nous montre l'interface graphique du Plugin. Sur le coté gauche de la fenêtre se situe la chaine de plugin utilisé. Il est possible de placer plusieurs plugins en séries (exemple émulation de haut-parleur suivie d'une émulation d'espace acoustique). Nous pouvons également activer et désactiver les plugins en temps réel ce qui nous permet de nous rendre compte de l'utilité de celui ci. En bas à gauche, nous pouvons lire la charge cpu associée au plugin pour les 2 processeurs.

2. Pseudo temps réel, il y a tout de même une latence contrôlée par la taille du buffer utilisé et qui dépend des performances de l'ordinateur et de sa carte son.

3. section 9 page 100

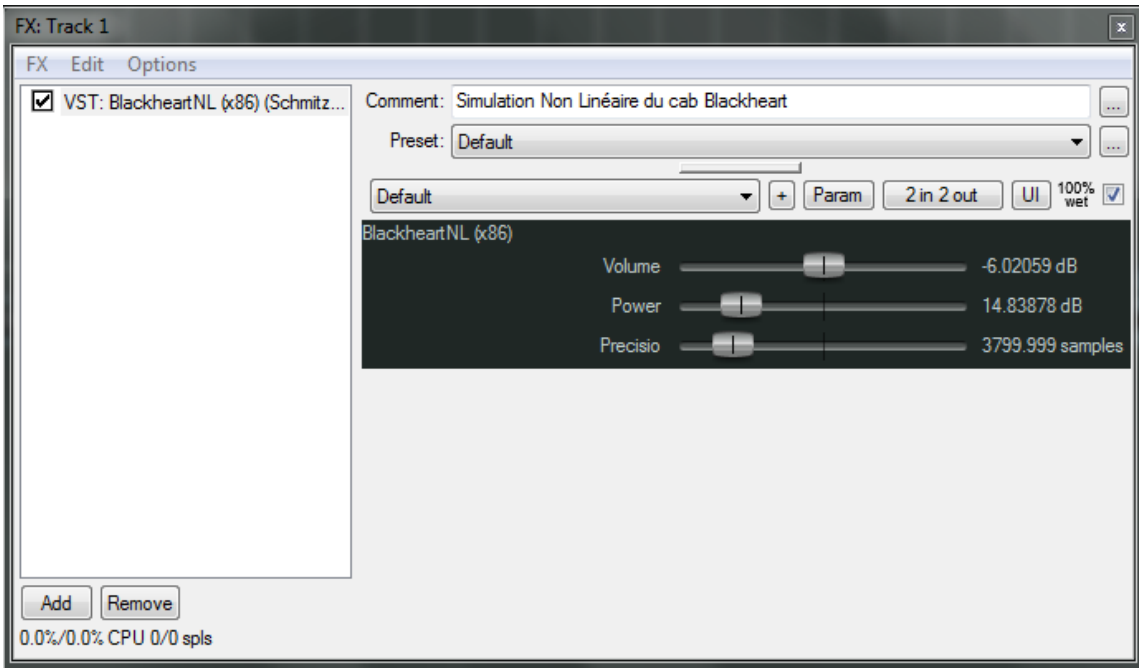


Figure 5.12 – Plugin de simulation non linéaire du cab Blackheart (interface graphique)

5.3.2.1 Les paramètres associés au plugin

1. **Volume :** gère le niveau de sortie du plugin.
2. **Power :** compense la perte de niveau provoquée par la loadbox, rappelons que le signal récupéré par la loadbox aux bornes du haut-parleur possède une grande amplitude. Afin de pouvoir traiter ce signal dans un ordinateur, la loadbox atténue celui ci jusqu'à un niveau ligne. Le plugin tient compte de cette atténuation et la compense.

   La loadbox est calibrée et l'atténuation qu'elle provoque peut être lue sur sa façade avant. Ce facteur *Power* est représentatif de la puissance envoyée au haut-parleur. En effet, si la puissance envoyée au haut-parleur est élevée, nous avons successivement que :

   (a) la tension aux bornes du haut-parleur est élevée,

   (b) l'atténuation de la loadbox doit être forte (pour atteindre un niveau ligne),

   (c) le gain de compensation doit être élevé $\Rightarrow$ facteur *Power* élevé.

3. **Précision :** permet de choisir, en temps réel, la longueur de la réponse impulsionnelle linéaire à convoluer avec le signal d'entrée. Nous pouvons ainsi adapter la précision du plugin à la puissance de la machine utilisée (ou aux ressources que nous souhaitons lui attribuer).

   D'après les blind-tests effectués [4], une longueur de 3000 échantillons est nécessaire

4. Voir section 5.4.4 page 80

pour tromper l'oreille. Les longueurs des réponses impulsionnelles d'ordre 2 et 3 sont liées à la longueur de la réponse linéaire par des rapports 2,5 :1 et 5 :1 [5] .

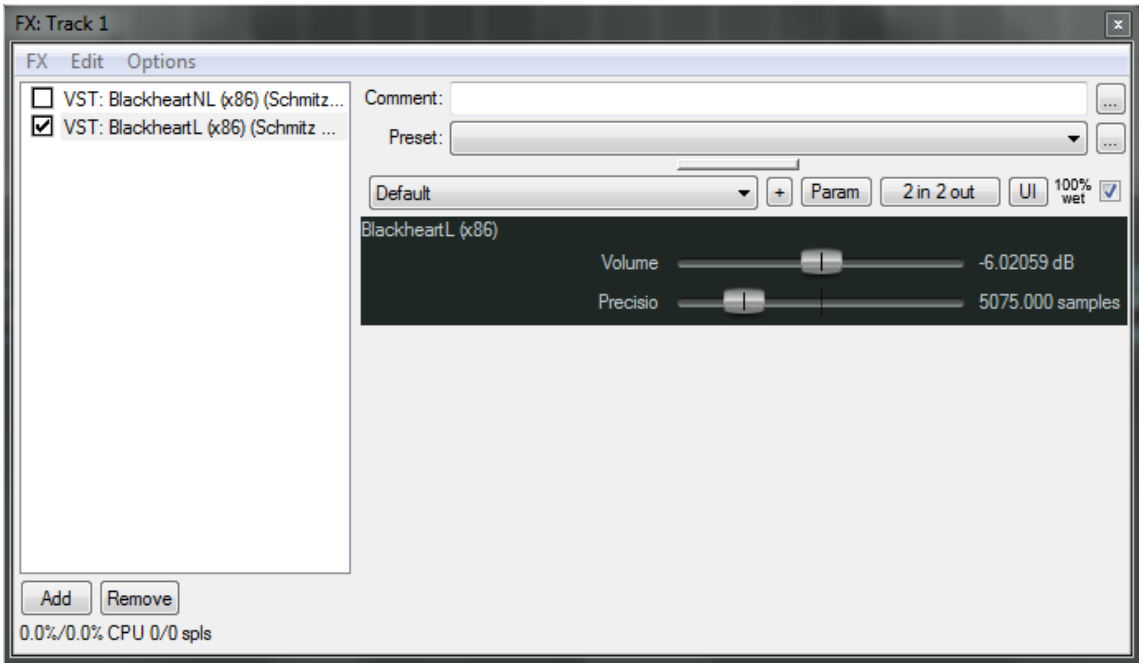


Figure 5.13 – Plugin de simulation linéaire du cab blackheart (interface graphique)

5.4 Tests Audio

5.4.1 Test 1 : comparaison en direct de la simulation linéaire et non-linéaire du haut-parleur

L'hôte Reaper permet d'activer et de désactiver le plugin en temps réel, ce qui permet de créer une liste de 2 plugins (un plugin de simulation linéaire et un plugin de simulation non-linéaire) et de choisir le plugin à utiliser en le sélectionnant d'un simple clic. Le basculement d'un modèle vers l'autre peut alors se faire en moins d'une seconde, rendant la comparaison des deux modèles aisée.

5.4.2 Test 2 : rendu de la simulation

Un autre test intéressant serait de simuler 5 haut-parleurs différents, un morceau pré-enregistré via la loadbox serait ré-enregistré en passant soit par le plugin soit par le haut-parleur.

Les différents samples audio serait alors mélangé ensemble et il serait demandé au testeur de rassembler les différents samples en catégories.

5. Pour plus d'explication voir section 9.5 page 108

Si le testeur associe les haut-parleurs à leurs simulations correspondantes, alors le test prouve que la simulation est capable de rendre la signature sonore de chaque haut-parleur avec suffisamment de précision pour permettre une distinction entre les haut-parleurs .

Note :

Il se pourrait que plusieurs haut-parleurs se ressemblent en terme de signature sonore ; si tel est le cas, le testeur qui associe 2 haut-parleurs dans une catégorie devra nécessairement leurs associer les 2 simulations correspondantes pour que le test soit concluant.

5.4.3 Test 3 : comparaison d'enregistrements effectués à l'aide du plugin linéaire et du plugin non-linéaire

La simulation du haut-parleur à l'aide des séries de Volterra est bien plus gourmande en cpu que la simulation linéaire. Il est donc légitime de se demander si la simulation non-linéaire est supérieure à la simulation linéaire. Ce test est similaire au test 1 mais ne se fait plus en direct [6], il est composé de 3 parties :

1. Envoyons un morceau pré-enregistré au haut-parleur et au plugin, nous obtenons :
   - 1 sample **A** qui est l'enregistrement du morceau passé à travers le haut-parleur.
   - 1 sample **B** qui est l'enregistrement du morceau passé à travers le plugin non-linéaire.
   - 1 sample **C** qui est l'enregistrement du morceau passé à travers le plugin linéaire.

2. Donnons au testeur 3 samples A identiques et un sample B, le testeur écoute les 4 samples et détermine celui qui est différent des autres (s'il le peut).

3. Finalement, le point précédent est répété mais avec 3 samples A identiques et 1 sample C

Ce test met 2 éléments en évidence :

1. Les testeurs peuvent-ils discerner le haut-parleur de leur simulation ?
2. Les testeurs discernent-ils mieux le haut-parleur de sa simulation linéaire ou le haut-parleur de sa simulation non-linéaire ?

Les tests 2 et 3 n'ont pas été implémenté jusqu'à ce jour.(mai 2012)

6. Au test 1, le guitariste écoute en direct le son qu'il joue, passé par le plugin. Au test 3, il écoute le morceau qu'il a enregistré et qui passe par le plugin linéaire ou non-linéaire.

5.4.4 Longueur de la réponse impulsionnelle

Comme nous l'avons vu au chapitre 2[7], la convolution d'un signal en temps réel avec une réponse impulsionnelle de 1 seconde demande 3.88 Giga opérations par seconde. En nous limitant à la série de Volterra constituée de 3 noyaux, nous aurions 11.64 Gops à effectuer rien que pour les opérations de convolution. Il est judicieux de déterminer la réponse impulsionnelle la plus courte possible sans perte de précision audible.

En envoyant au plugin de simulation linéaire 1 morceau audio et en récupérant le résultat de la convolution pour 3 longueurs différentes de la réponses impulsionnelle, nous obtenons 3 samples :

1. Le sample 1 a été convolué avec une réponse impulsionnelle d'une longueur de 16000 samples.
2. Le sample 2 a été convolué avec une réponse impulsionnelle d'une longueur de 246 samples.
3. Le sample 3 a été convolué avec une réponse impulsionnelle d'une longueur de 3100 samples.

Le sample 1 est le sample de référence (le plus précis des 3). Il est demandé au musicien de comparer le sample 2 et 3 au sample 1 , afin de déterminer s'il perçoit une différence entre les samples.

Nous avons ajouté un $4^{\text{ème}}$ sample "piège" qui n'est autre qu'une copie du premier. Ce $4^{\text{ème}}$ sample permet "d'éliminer" les testeurs qui entendent une différence là où il n'y en a pas (victimes de leur subconscient).

Le premier diagramme de la figure 5.14 montre les résultats du test. Nous remarquons que :

- Une part importante des guitaristes ont perçu une différence entre 2 sons identiques (sample 1 et 4).
- Quelques guitaristes ne perçoivent pas de différence entre les samples même pour une réponse impulsionnelle très courte.
- Le sample 2 a quand même été reconnu comme étant le moins proche du sample 1.

Puisqu'un même testeur peut voter pour plusieurs samples il est important de savoir pour quels samples il a voté, en effet nous avons vu qu'une personne ayant voté pour une différence entre le sample 1 et 4 (identiques) doit être "éliminée", il existe d'autres types de combinaison qui montre que le vote n'est pas objectif, si bien que les testeurs sont rassemblés en 3 catégories :

1. Cette catégorie rassemble les personnes n'ayant voté que pour le sample 2, ils ne se sont pas laissés influencer par leur subconscient et entendent une différence entre

7. section 2.5.2 page 19

le sample le plus précis (16000 échantillons) et le moins précis (246 échantillons).

2. Cette catégorie rassemble les personnes ayant une très bonne oreille, en effet ils ont distingué une différence entre le sample 2 (246 échantillons) et le sample 1 (16000 échantillons) **et** une différence entre le sample 3 (3000 échantillons) et le sample 1 , mais ne sont pas tombés dans le piège du sample 4 (= sample 1).

3. Cette catégorie rassemble tous les autres testeurs c'est-à-dire :
   (a) Les personnes qui n'entendent pas de différences entre les différents samples.
   (b) Les personnes qui entendent une différence entre 2 samples identiques (ils cherchent l'erreur et finissent par en trouver là où il n'y en a pas).
   (c) Les personnes ayant détecté une différence entre le sample 3 (3100 échantillons) et le sample 1 (16000 échantillons) mais n'ayant pas décelé de différence entre le sample 1 et 2 (246 échantillons) (bonne réponse hasardeuse).

Ces différentes catégories sont représentées sur le deuxième graphique de la figure 5.14. En rassemblant la partie rouge et bleue, Nous pouvons avancer l'hypothèse que 50% des musiciens testés peuvent reconnaître le sample 2 (246 échantillons) du sample 1 (16000 échantillons).

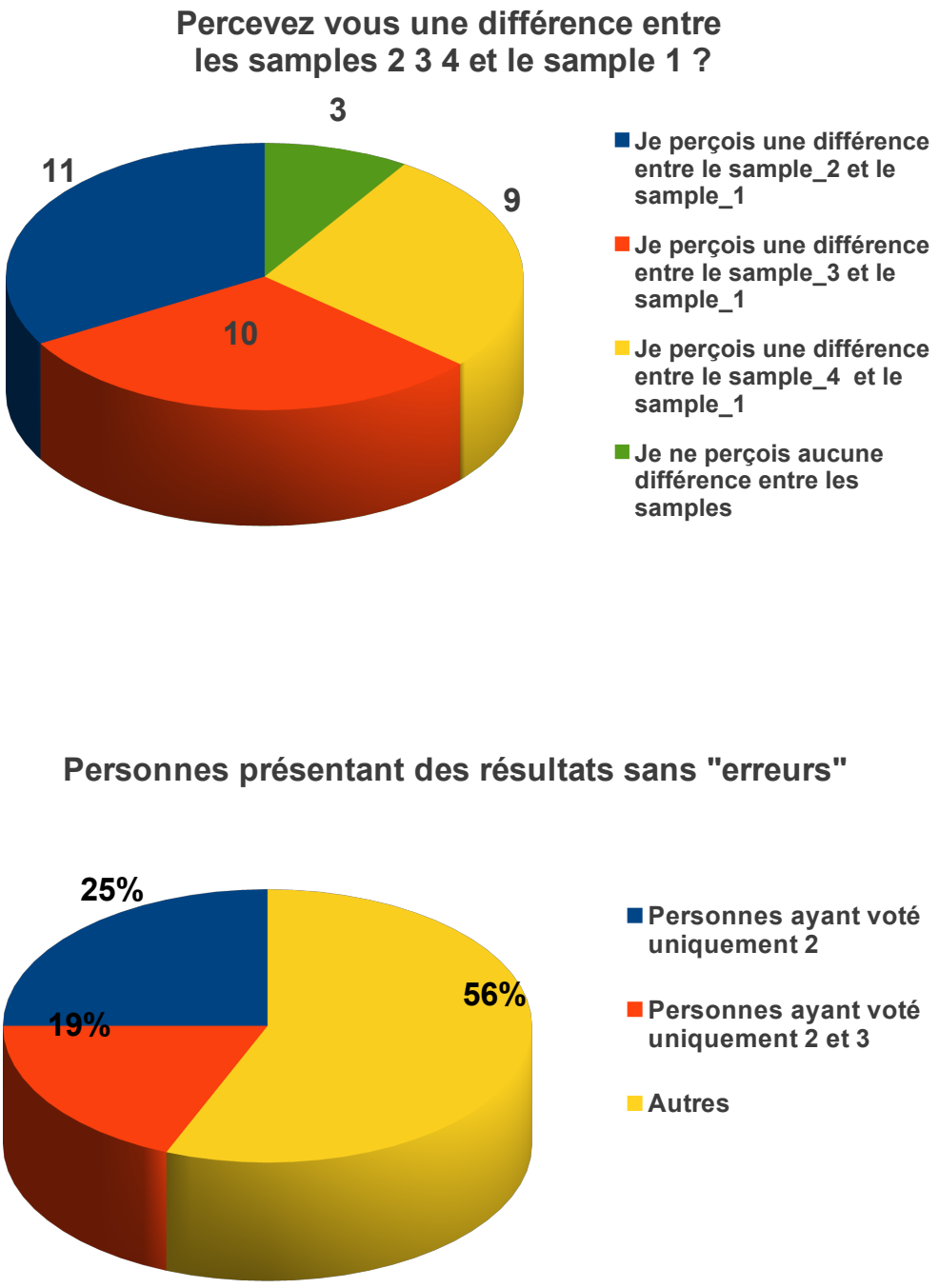


Figure 5.14 – Différences perçues par des guitaristes entre le sample 1 et les autres samples

5.4.4.1 Second test sur le choix de longueur de la réponse impulsionnelle

Le premier test permet de dégager 2 valeurs clés quand au choix de la longueur de la réponse impulsionnelle nécessaire, il semblerait que :

- Pour 50% des guitaristes,la réponse de 246 échantillons soit trop courte
- Pour 20% des guitaristes, la réponse de 3100 échantillons soit trop courte, ce qui veut aussi dire qu'elle est suffisante pour 80% des guitaristes.

Pour nous assurer de ce constat, nous proposons un second test :

- De façon similaire au test 3, il faut "déceler l'erreur" qui s'est glissée parmi les 4 samples.
- 3 samples seront identiquement constitués de la convolution d'un morceau de musique avec la réponse impulsionnelle linéaire (16000 échantillons) du haut-parleur.
- 1 des samples est constitué de la convolution du même morceau de musique avec la réponse impulsionnelle linéaire (3100 échantillons) du haut-parleur.
- Il est demandé au testeur de déceler le sample différent des autres.

Influence de la longueur de la réponse impulsionnelle ?

Tronquer la réponse impulsionnelle, c'est la fenêtrer en temporel. L'effet de ce fenêtrage sur la transformée de Fourier de la réponse impulsionnelle du haut-parleur est présenté aux figures 5.15, 5.16 et 5.17 .

Nous observons que les courbes deviennent moins détaillées avec la diminution de la longueur de la réponse impulsionnelle. Cet effet peut s'expliquer par le lien qui existe entre la largeur de la fenêtre rectangle et la largeur du lobe principal de sa transformée de Fourier. [8]

8. Voir section 4.2.1.2 page 41 et figure 4.8

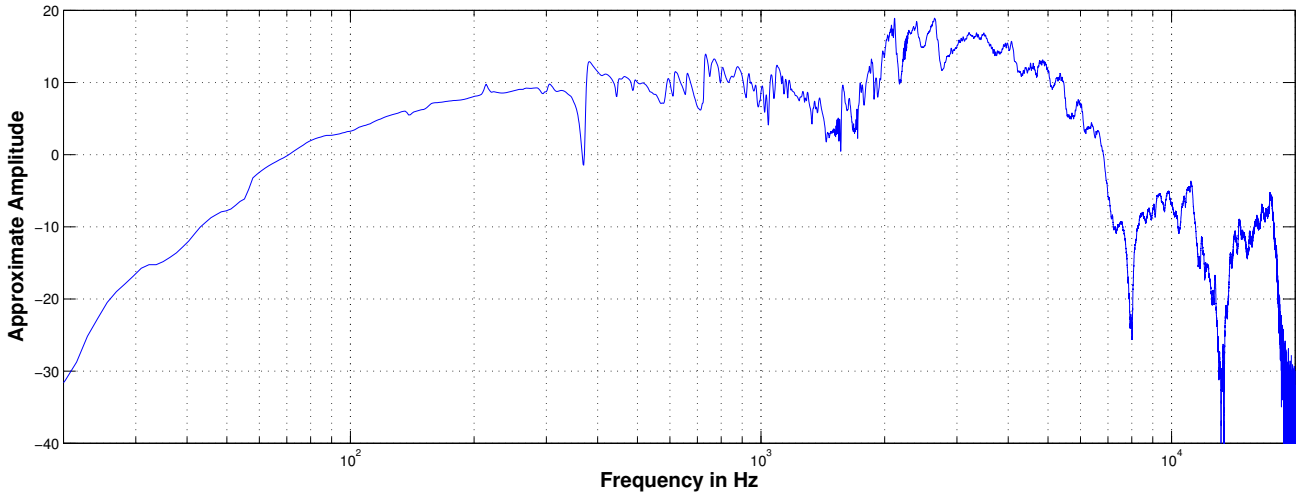


Figure 5.15 – Transformée de Fourier de la réponse impulsionnelle du haut-parleur tronquée après 16000 échantillons

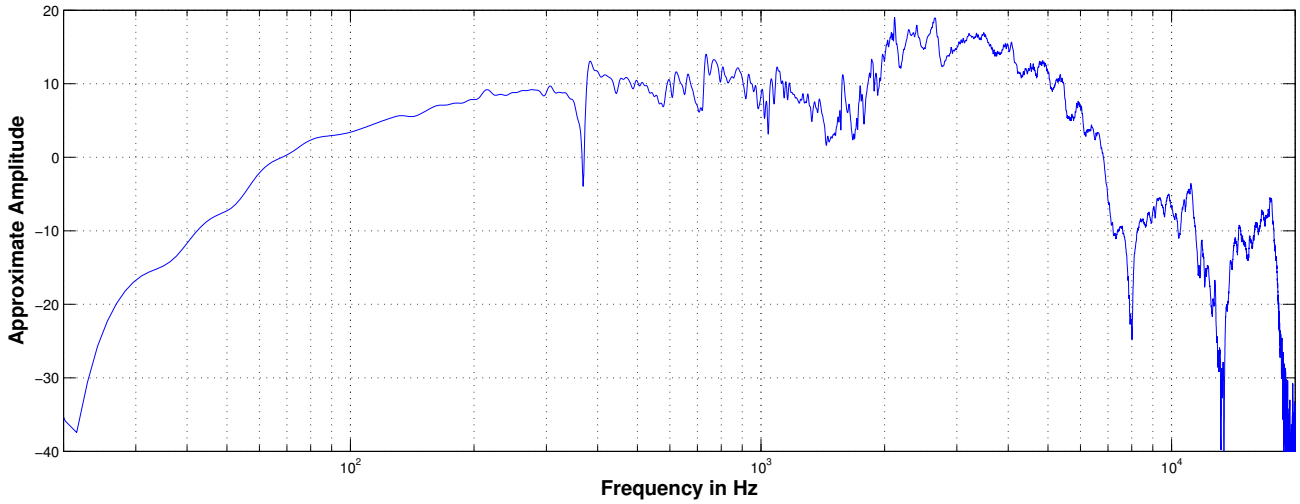


Figure 5.16 – Transformée de Fourier de la réponse impulsionnelle du haut-parleur tronquée après 3100 échantillons

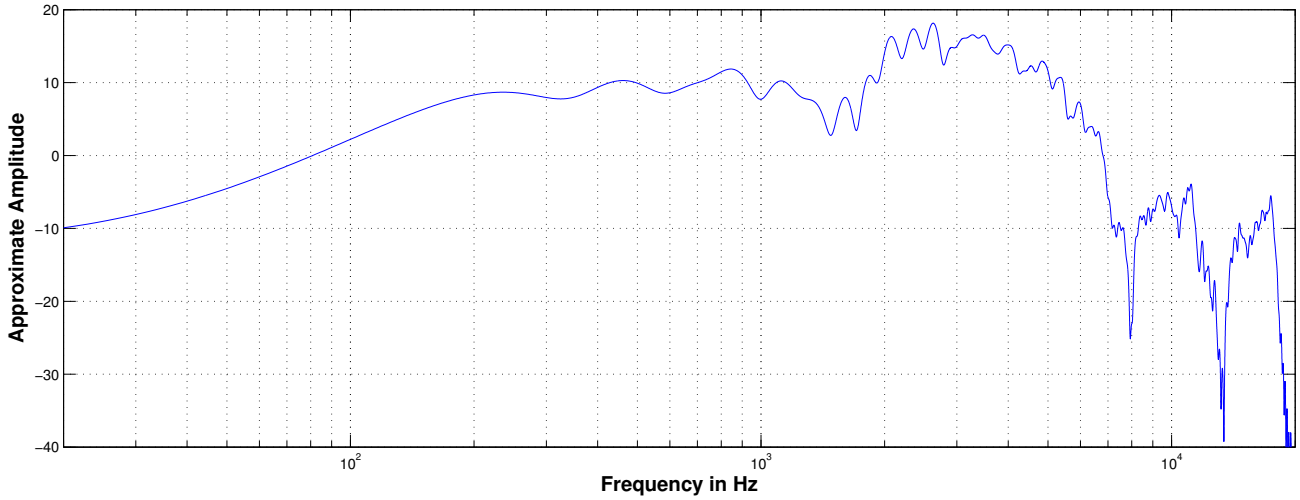


Figure 5.17 – Transformée de Fourier de la réponse impulsionnelle du haut-parleur tronquée après 246 échantillons

6 Futurs travaux et perspectives

6.1 Liste des non-idéalités

Lors des différents tests et mesures, nous avons observé une série de non-idéalités pouvant entraîner des différences de sonorité entre le haut-parleur et sa simulation.

L'importance de ces non-idéalités est à mettre en relation avec le matériel utilisé et/ou la direction d'éventuels futurs travaux. Les non-linéarités écrites en gras sont celles que nous avons jugées les plus à même d'entraîner des différences entre le haut-parleur et sa simulation.

6.1.1 Non idéalités du modèle de Volterra

1. **Choix du modèle non linéaire n°2, ne tenant pas compte des artéfacts** (voir figure 3.2 page 23).
2. La série de Volterra est infinie en temps et en ordre mais nous n'avons considéré que les noyaux jusqu'à l'ordre 5 et de temps max = 20 secondes.
3. Le modèle n'est valable que pour les systèmes faiblement non linéaires (voir section 3.3 page 33).

6.1.2 Non idéalités des mesures

1. Bande passante de la carte son 1 .
2. **Bande passante de l'ampli Kenwood amplifiant le sweep** (voir section 11.1.2 page 117).
3. Variation de l'amplitude du signal électrique aux bornes du haut-parleur (variation de l'impédance de sortie de l'ampli avec la fréquence ?).
4. Échauffement de la bobine et variation du niveau du signal acoustique émis en fonction du temps (voir figure 4.23 page 59).
5. Bande passante du microphone [38].

6. Sweep borné dans le temps et choix du filtre utilisé pour le fenêtrage (voir section 4.2.1.2 page 41).
7. Signaux parasites à 50 Hz et ses multiples.

6.1.3 Non idéalités d'écoutes

1. **Bande passante de la carte son 2** (voir section 5.1.3 page 70)
2. **Bande passante et non-linéarités de la loadbox.** (voir section 5.1.1 page 66)
3. Restitution des signaux provenant de la simulation, sur un haut parleur et une chaîne Hifi considérés comme "transparent".
4. ...

6.2 Etude d'optimisation

6.2.1 SNR

Nous pourrions nous intéresser aux paramètres du sweep (durée, plage de fréquence, fenêtre) nous amenant à la réponse impulsionnelle la plus optimale. En effet, nous avons vu que le type et la longueur du filtre fenêtrant le sweep avait une incidence sur la précision, la bande passante et l'oscillation de la transformée de Fourier de la réponse impulsionnelle.

Un sweep logarithmique plus long implique une énergie plus importante en basse fréquence, menant à une précision accrue des fréquences basses, ce qui est souhaitable pour un haut-parleur de guitare puisque fondamentalement les notes de la guitare sont comprises entre 80 et 1500 Hz.

En contre partie, un sweep plus long accumule les parasites constants. En effet si un parasite à 50 hertz est présent sur toute la durée du sweep , alors son amplitude relative aux autres fréquences sera d'autant plus importante que la durée du sweep est grande.

6.2.2 Longueur des réponses impulsionnelles pour le plugin

Comme nous l'avons vu à la section 5.4.4, les longueurs des différents noyaux de Volterra à utiliser dans le plugin restent à optimiser en fonction de l'importance à accorder à chacun deux.

6.3 Optimisation de la charge cpu

La convolution brute est très gourmande en opérations, il serait intéressant d'accélérer le processus de convolution moyennant certain compromis. nous pourrions, par exemple, nous baser sur un des algorithmes décris dans les articles [8], [39],[25].

6.4 Un modèle tenant compte des artéfacts

Il pourrait être intéressant de chercher à compléter la modélisation sur base des différents artéfacts décrits à la section 3.2 page 27.

Mesure de ces artéfacts

La section 3.2 essaie de déterminer le lieu des répliques additionnelles sous l'hypothèse qu'il existe un écart constant entre τ_1 et τ_2 tel que le double produit de convolution $h_2[\tau_1, \tau_2]$ mène à une contribution significative non diluée dans le temps. Certaines de ces contributions se superposent rendant difficile leurs identifications. Une manière de les mettre en évidence serait de réaliser d'autres sweeps avec des fréquences w_1, w_2 et des temps T différents, afin que le terme $\frac{w_2}{w_1}^{\frac{-\Delta}{N}}$ des équations conduisent à des Δ_{xx} différents pour un même Δ .

6.5 D'autres façons de mesurer les noyaux de Volterra

Dans ce travail nous avons cherché à obtenir les noyaux de Volterra sur base d'une excitation en sweep logarithmique. D'autres travaux[1] tel que ceux de M.J.Reed et M.O.J.Hawksford propose de déterminer les coefficients du modèle de Wiener en appliquant un signal MLS à différents niveaux et d'en déduire les noyaux de Volterra en appliquant un changement de Base [43], [44],[42].

6.6 Comparaison Mécanique et noyaux de Volterra

Parallèlement à cette étude sur l'obtention des noyaux de Volterra, il serait intéressant de comprendre quels phénomènes physiques et mécaniques sont responsables des différentes répliques mesurées. [9], [10],[37],[6],[24],[36], [34],[29],[35],[36].

6.7 Amélioration du plugin

Outre l'aspect technique du plugin, celui ci pourrait être développé de façon à mieux prendre en compte les besoins du studio, entre autres :

- prise de son stéréophonique,
- choix de l'enceinte et du micro qui la reprend,
- placement du micro (position par rapport à l'axe du haut-parleur et inclinaison).

1. Voir aussi [3],[12],[13],[29]

6.8 La charge

Émulation de l'impédance du haut-parleur pour respecter la réponse en fréquence de l'amplificateur guitare [2]

2. La question de l'émulation de la charge est discutée dans l'appendix D section 11.1.4 page 119

7

Conclusion

Dans ce travail, nous avons modélisé le comportement du haut-parleur électrodynamique en tenant compte de ses non-linéarités. La modélisation est basée sur la théorie des systèmes non-linéaires invariants et de la série de Volterra [1].

L'identification des noyaux de Volterra est une part importante de ce mémoire, nous nous sommes basés sur les propriétés [2] du sweep logarithmique pour séparer les distorsions harmoniques de la fondamentale sur la réponse impulsionnelle du haut-parleur [3].

La position des différentes répliques était décrite dans la littérature ([21],[7]) et nous avons dérivé un modèle mathématique pour reconstruire la diagonale principale des noyaux de Volterra via ces répliques. L'hypothèse selon laquelle la diagonale des noyaux de Volterra est suffisante pour modéliser un haut-parleur est basée sur la supposition que le haut-parleur peut être modélisé par un système non-linéaire sans mémoire suivi d'une réponse impulsionnelle "classique" d'un haut-parleur(voir modèle non-linéaire numéro 2 , figure 3.2).

La présence de répliques "artéfacts" observées lors de nos mesures de réponses impulsionnelles [4] nous a permis de montrer que cette hypothèse est probablement trop forte. Nous avons alors proposé quelques formules afin de relier la position des répliques "artéfacts" [5] à un des maxima dans la fonction d'auto-corrélation.

Nous avons ensuite développé un plugin permettant de réaliser des tests audio afin de comparer la simulation du haut-parleur au son provenant directement du haut-parleur. Il est difficile de tirer une conclusion de ce test audio vu les non-idéalités d'écoute. Nous avons aussi comparé le plugin dit "linéaire" au plugin "non-linéaire" : une différence est audible entre les deux simulations mais aucun test permettant de choisir la simulation la plus fidèle n'a été implémenté jusqu'à ce jour (mai 2012).

1. section 2.4 page 16
2. Voir section 10.1.3
3. figure 2.3
4. voir figure 3.3
5. section 3.2.2 page 27

Nous avons aussi montré que la technique de reconstruction du modèle à l'aide de la série de Volterra n'était pas effective dans 100% des cas [6] (modélisation incomplète ou erreur de mesure ?).

Finalement, remarquons que les haut-parleurs modélisés ne sont pas fortement non-linéaire et qu'il est donc difficile de juger graphiquement d'une éventuelle amélioration de la fidélité du modèle. Une modélisation d'un ampli de puissance à lampe ou un effet overdrive pour guitare [7] serait sans doute plus à même de mettre en évidence cette amélioration(plus de non-linéarité).

Rappelons quelques avantages de l'utilisation d'une simulation de haut-parleur.

1. Possibilité d'obtenir des dizaines d'enceintes différentes pour un faible coût et de les avoirs toujours avec soit (en chargeant les réponses impulsionnelles sur une clé usb par exemple).
2. Plus besoin de se déplacer avec des enceintes lourdes et volumineuses, un ordinateur contenant le plugin suffit.
3. Indépendance de l'enregistrement face aux bruits ambiants.

6. section 4.3.4
7. voir [5]

8

Appendix A : code Matlab

Ce chapitre contient tout ce qui concerne le code Matlab. Une brève description des fonctions utilisées sera présentée suivie d'un exemple d'utilisation afin de passer de la réponse impulsionnelle mesurée aux différents noyaux de Volterra directement appelables par le plugin.

8.1 Description des fonctions Matlab

Fonction computeKernels.m

Inputs :

- Une matrice constituée des noyaux mesurés h'_i, la colonne i contenant le noyau mesuré i.
- Le facteur d'amplitude A.

Outputs :
- une matrice contenant les noyaux de Volterra Hv_i reconstitués selon les formules (3.13)
- la valeur de B (voir (10.6))

La fonction est paramétrée pour un sweep de 20 secondes balayant les fréquences comprises entre 20 et 20000 Hz

Fonction extractionKernels.m

Inputs :
- impulse est un fichier .wav contenant la réponse du système excité par un sweep logarithmique.
- tFirstKernel est le moment (en seconde) où débute la réponse dite linéaire, h'1.
- longueurImpulse est un vecteur pour lequel la case i contient la longueur (en échantillons) de l'impulse h'i souhaité.

– nbKernels est le nombre de noyaux de Volterra souhaités pour la reconstruction.

Outputs : un tableau dont la colonne i correspond au noyau h'i

Bien qu'il soit possible de déterminer les longueurs des différents noyaux h'i séparément, la fonction renvoie un tableau où chaque impulse a été complété par des zéros afin de posséder les mêmes dimensions (car les différents noyaux devront être sommés entre eux).

Fonction FindA.m

Inputs :

– A le facteur d'amplitude **estimé**.
– Hf la matrice contenant la transformée de Fourier des noyaux de Volterra.
– fs la fréquence d'échantillonage.
– Un fichier wav contenant l'enregistrement d'un sinus à la fréquence de 100 Hz et d'amplitude 31 V_{rms} passé sur le haut parleur.

Outputs : Un graphique comparant l'amplitude des harmoniques du sinus provenant du haut parleur et de la simulation .

Note : Le facteur A passé en argument de la fonction est correct : si la **différence** d'amplitude, entre la fondamentale et ses harmoniques est la même pour les deux systèmes.

Fonction impulsePlugin.m

Inputs : un vecteur de 3 cases contenant les longueurs (en nombre d'échantillons) des trois premiers noyaux de Volterra désirés dans le plugin.

Outputs : un fichier texte contenant les noyaux de Volterra que l'on peut placer dans le plugin par un simple copier coller (il faut tout de même recompiler le plugin).

Fonction rmsWave.m

Inputs : un fichier wav dont nous voudrions connaître la tension V_{rms}.

Outputs : la valeur rms du signal.
Cette fonction sert surtout pour les tests comparatifs. En effet, pour comparer 2 signaux, il faut qu'ils aient le même niveau sonore.

Fonction showFft.m

Inputs : un vecteur contenant notre signal et sa fréquence d'échantillonnage.

Outputs :
- X= 20 log de la valeur absolue de la transformée de Fourrier de notre signal.
- w l'axe des fréquences correspondantes à X
- si aucun argument de sortie n'est spécifié, showFft trace X en fonction de w sur un axe logarithmique.

Fonction timeBetweenHarmo.m

Inputs : le nombre de kernels désirés.

Outputs : un vecteur contenant à la case i, le temps qui sépare le noyau h'1 du noyau h'i .

Note : la première case vaut évidemment 0.

Fonction ViewWave.m

Inputs : un fichier wav.

Outputs : un graphique du fichier wav en fonction du temps.

Fonction ViewFftWave.m

Inputs :

- Un fichier .wav dont on voudrait visualiser la réponse fréquentielle
- Optionnel : un plage de fréquence à visualiser (exemple [20 20000] pour visualiser la bande audio).
- optionnel : une plage d'amplitude à visualiser (exemple [-20 60] pour visualiser le signal compris entre -20 et 60 db).

Outputs : Trace le graphique de 20 log de la valeur absolue de la transformée de Fourrier sur un axe logarithmique compris dans la plage de fréquence spécifiée en entrée.

8.2 Exemple d'extraction des noyaux de Volterra pour une utilisation dans le plugin

Comme nous avons eu beaucoup de mesures à traiter, nous avons écrit une fonction main capable de générer les différents noyaux de Volterra et de comparer les différentes réponses presque automatiquement. Elle se présente sous la forme d'une suite de cellules.

Fonction extractionDesNoyauxDeVolterra_exemple.m

La première étape est de récupérer la réponse impulsionnelle h(t) du système obtenue par déconvolution du signal de sortie du système (sortie micro) lorsqu'un sweep logarithmique est appliqué à son entrée.

Nous obtenons un fichier wav que nous pouvons visualiser avec la fonction *viewWave()*, afin d'obtenir un graphique similaire à la figure 4.13. Dans cet exemple, nous l'avons nommé *14logsweep31V* (14ème laboratoire , excitation logarithmique d'amplitude 31 Volts).

Si nous ne désirons que les noyaux de Volterra, alors nous pouvons insérer ce wav à la ligne 5 et 6. Si nous désirons aussi comparer la reconstruction non-linéaire de la série de Volterra à la réponse impulsionnelle obtenue en déconvoluant le signal de sortie quand le système est excité par un sweep linéaire, alors nous aurons besoin de 2 autres wav :

1. hlin2V[1] (nommé ici 14linsweep2V)
2. hlin31V (nommé ici 14linsweep31V)

Finalement, nous aurons besoins de la réponse du haut parleur excité par un sinus de fréquence 100 Hz et d'amplitude 31 Vrms nommé *31sin(100).wav*

Première cellule

```
1 %% Chargement des variables
2 clear
3
4 impulse='14logsweep31V.wav';
5 impulse_lin_bas_level='14linsweep2V.wav';
6 impulse_lin_haut_level='14linsweep31V.wav';
7 Asinus='31sin(100).wav';
8
9 nbKernels=5;
10 tFirstKernel=20;
11 longueurImpulse=[83000 25000 15000 10000 10000];
12 longueur_noyaux_plugin=[25000 10000 5000];
13 tFirstLin=20;
14 A=22.5;
15
16 fs=44100;
17 nbits=16;
```

A la ligne 9, nous indiquons le nombre de noyaux de Volterra désirés, même si le plugin ne se sert que de 3 noyaux, il vaut mieux laisser 5 noyaux puisque le noyau hv3 par

1. voir section 4.3.3.3 page 57

exemple, dépend des noyaux mesurés h'3 et h'5 (note : 5 noyaux est le maximum admis).

A la ligne 14, nous indiquons une première estimation du facteur d'amplitude A.

- tFirstKernels est le moment où débute le noyaux h'1 (en seconde)
- longueurImpulse donne la longueur des noyaux h' dont nous nous servirons pour la reconstruction
- longueur_noyaux_plugin donne la longueur des noyaux de Volterra qui seront utilisés dans le plugin.

Deuxième cellule

```
1
2 tableauKernels= extractionKernel(impulse,tFirstKernel,
    longueurImpulse,nbKernels);
3
4 for i= 1:nbKernels
5 nomFichier=['h' int2str(i) 'p.wav'];
6 wavwrite(tableauKernels(:,i),fs,nbits,nomFichier)
7 end;
8
9 newTaille=size(tableauKernels);
```

Il s'agit d'extraire les différents noyaux mesurés h'i et de les écrire dans des fichiers hpi.wav (p comme prim ' et i est l'ordre du noyau). Ces fichiers seront disponibles dans le répertoire courant de matlab. Nous pouvons vérifier grâce à la fonction *viewWave* que les fichiers sont correctement coupés (h'1 doit commencer le plus tôt possible, si l'impulse est mal coupé, alors la variable tFirstKernel est mal choisie à la cellule précédente.)

Troisième cellule

Cette troisième cellule calcule la transformée de Fourier des noyaux mesurés hp_i

```
1 %% Fft des signaux:
2
3 Hp=zeros(newTaille(1,1),nbKernels);
4 for i=1:nbKernels
5     Hp(:,i)=fft(tableauKernels(:,i));
6 end;
```

Quatrième cellule

Sur base de la transformée des noyaux mesurés Hp_i, La fonction *computeKernels* calcule la transformée de Fourier des noyaux de Volterra.

```
1
2 [Hf,B]=computeKernels(Hp,A);
```

Cinquième cellule

La 5ème cellule permet de convertir les différents noyaux de Volterra fréquentiels Hp en réponse impulsionnelle écrite dans des fichiers hvi.wav où l'indice i donne l'ordre du noyau de Volterra.

```
%% conversion des Noyaux de Volterra en .wav

for i = 1:nbKernels
nomFichier=['h' int2str(i) 'v.wav'];
wavwrite(real(ifft(Hf(:,i))),fs,nbits,nomFichier);
end;
```

Sixième cellule

Cette cellule permet de vérifier que le facteur A choisi à la cellule 1 est le bon (voir section 4.3.3.2). Si le facteur A est correct, pressez la touche *Enter* pour passer à la cellule suivante, sinon il faut recommencer à la cellule 1 avec un autre facteur A.

```
%% Calcul du Facteur A

findA(A,Hf,fs,Asinus);
pause
```

Septième cellule

Cette sixième cellule compare graphiquement H1v à Hlin2V. voir section 4.3.4.3 page 62

```

figure,
handle0(1)=viewFftWave('h1v.wav',[20 20000]);
hold all,
handle0(2)=viewFftWave(impulse_lin,[20 20000]);
title('H1 volterra et reponse lineaire a bas niveau (1V)')
legend(handle0,'H1v','lin(1V)');

hold off;
```

Huitième cellule

Nous comparons cette fois Hvtot,Hlin31V et H1v voir section 4.3.3.4 page 57.

```
%% Comparaison Htot,H1v,Hlin30V

Htot=zeros(newTaille(1,1),1);
for i=1:nbKernels
Htot=Htot+Hf(:,i);
end;
```

```
wavwrite(real(ifft(Htot)),fs,nbits,'htot.wav');

figure,
handle1(1)=viewFftWave('htot.wav',[20 20000]);
hold all
handle1(2)=viewFftWave(impulse_lin_haut_level,[20 20000]);
handle1(3)=viewFftWave('h1v.wav',[20 20000]);
title('Comparaison r ponse totale reconstruite,H1v et Hlin
    30V')
legend(handle1,'Htot','Hlin 30V','H1v')
hold off
```

Neuvième cellule

Comparaison des réponses fréquentielles des différents noyaux de Volterra voir figure 4.28 page 65

```
%% Comparaison des Hv

figure,
for i=1:nbKernels

nomFichier=['h' int2str(i) 'v.wav'];
p(i+1)=viewFftWave(nomFichier,[20 20000]);
hold all
end;
p(1)=viewFftWave('htot.wav',[20 20000]);
title('Comparaison Hv et Htot')
legend(p,'Htot','H1v','H2v','H3v','H4v','H5v')
hold off
```

Dixième cellule

Elle permet de visualiser les différences qui existent entre les réponses fréquentielles obtenues à bas et haut niveaux lorsque nous appliquons un sweep linéaire en entrée.

```

figure,
handle2(1)=viewFftWave(impulse_lin_bas_level,[20 20000]);
hold all
handle2(2)=viewFftWave(impulse_lin_haut_level,[20 20000]);
title('Comparaison Hlin')
legend(handle2,'Hlin1.5V','Hlin31')
hold off
```

Onzième cellule

Cette dixième cellule rempli un fichier texte (*impulsePlugin.txt*) des noyaux de Volterra, le texte est formaté en c++ et il ne reste qu'a l'insérer dans le code du plugin.

```
%% Adaptation des noyaux de Volterra pour le plugin

implulse_plugin(longueur_noyaux_plugin);
```

9 Appendix B : code Plugin

Cette section contient quelques explications quant au fonctionnement du plugin

9.1 Installation de la sdk

La première chose à faire pour réaliser un plugin est de télécharger la librairie *VST 2.4 Audio Plug-Ins SDK* sur le site `http://www.steinberg.net` (inscription obligatoire mais gratuite).

Ensuite il faut créer un répertoire *Vst* dans lequel placer la bibliothèque *VST SDK2.4*. Finallement, pour créer un nouveau plugin il suffit de :

1. Ouvrir un ide (code block dans mon cas).
2. Il faut ensuite cliquer sur *new empty project*, appeler le projet *monPlugin* et l'enregistrer dans le répertoire *Vst*.
3. L'archive *Plugin* du CD-Rom[48] contient un dossier appelé *VST_type*, il contient tous les fichiers à importer, nécessaires à la création d'un plugin.

   Le fichier AGain.cpp est le fichier d'un exemple simple de plugin, un bouton de volume, il est plus simple de se baser sur ce fichier à chaque départ d'un nouveau plugin que de recréer tout l'environnement à chaque fois.
4. Clic droit sur le projet,propriété, build target => DLL.
5. Clic droit sur le projet => Build Option => linker tab=> Other linker option, puis copier ce code :

```
1 -def E:\VST\vstsdk2.4\public.sdk\samples\vst2.x\win\
      vstplug.def
```

   La partie E :\VST est le chemin d'accès au répertoire VST et dépend du lieu où se trouve la librairie SDK2.4.
6. La dll devrait pouvoir compiler. Elle est créée dans le répertoire bin du projet *Monplugin*. Il suffit alors de la transférer dans le répertoire *plugin* du séquenceur que nous souhaitons utiliser.

9.2 Boost et multi thread

Boost est une bibliothèque qui entre autres, permet de gérer le multi-threading. Comme nous l'avons vu, il y a trois convolutions à appliquer en temps réel au signal d'entrée, ces convolutions peuvent être réalisées en parallèle afin de mieux utiliser toutes les ressources des processeurs.

En effet, par défaut, le plugin n'utilise qu'un thread sur les 4 généralement disponibles dans un ordinateur dual-core. L'utilisation des 4 threads nous permet d'utiliser une plus grande part des ressources disponibles et donc d'améliorer la précision de notre modèle.

Nous avons 3 convolutions à appliquer au signal d'entrée mais la convolution de la réponse linéaire est la plus longue [1] . Nous profitons du fait que l'ordinateur possède 4 threads pour spliter la tache de convolution de la réponse linéaire en 2 convolutions de moitié de réponse impulsionnelle linéaire. Ainsi nous obtenons 4 taches à effectuer :

1. Convolution de la première moitié de h1v avec le signal d'entrée.
2. Convolution de la deuxième moitié de h1v avec le signal d'entrée.
3. Convolution de h2v avec le signal d'entrée.
4. Convolution de h3v avec le signal d'entrée.

Les performances du plugin sont accrues [2] mais ne sont pas proportionnelles au nombre de Threads utilisés car nous devons instancier un nouveau thread à chaque *frame* [3] et cette opération est coûteuse en opération.

Finalement pour que le plugin fonctionne avec la librairie boost, il faut inclure le fichier *libboost_thread-mgw46-mt-1_49.dll* dans le répertoire Plugin de l'hôte utilisé. Nous essayé sans succès d'inclure la dll de boost dans la dll du plugin pour une utilisation facilitée. Il semblerait que codeblock en soit incapable alors que Visual-Studio le fait . (Par ailleurs il semblerait que l'installation de Boost soit grandement facilité avec Visual-Studio).

9.3 Principe de l'algorithme

Il faut savoir que le son n'est pas traité en temps réel dans Reaper. En effet, Reaper traite le son entrant par buffer de taille sampleFrames ce qui permet d'éviter des phénomènes de glitchs [4] lorsque le processeur est sollicité fortement pendant un cours instant. En contre partie le logiciel crée une latence inévitable (mais réglable en fonction

1. voir section 5.3.2.1 page 77
2. (convolution de 3000 échantillons max avec un seul thread contre une convolution de 5000 échantillons avec 2 threads)
3. voir plus loin
4. craquements

des performances de l'ordinateur).

Nous voudrions donc que le plugin effectue la tâche suivante pour tout y[n] devant se retrouver dans le buffer courant de sortie :

$$y[n] = h1[n] \circledast x[n] + h2[n] \circledast x^2[n] + h3[n] \circledast x^3[n] \tag{9.1}$$

Où :

- x[n] correspond au signal d'entrée (associé au tableau inputs dans le plugin)
- y[n] correspond au signal de sortie (associé au tableau outputs dans le plugin)
- $h_i[n]$ correspondent aux trois premiers noyaux de Volterra

L'opération de convolution est définie comme :

$$h[n] \circledast x[n] = \sum_{\tau=0}^{\infty} h[\tau]x[n-\tau] \tag{9.2}$$

Comme h[n] est de longueur finie M, le produit est nul pour $\tau > M$; par conséquent le produit de convolution peut s'écrire :

$$h[n] \circledast x[n] = \sum_{\tau=0}^{M} h[\tau]x[n-\tau] \tag{9.3}$$

Nous faisons ainsi apparaître la mémoire sur x nécessaire au produit de convolution, nous devons retenir les M derniers échantillons du signal d'entrée.

9.3.1 Buffer circulaire

Nous allons donc nous servir d'un buffer circulaire dont la première case contiendra la valeur la plus récente de x et la dernière case contiendra x[n-M]. Ces buffers sont au nombre de 4 :

1. cb_1a (voir Table 9.1) est le buffer retenant les M derniers échantillons de x[n] dans le but d'être convolué avec la première partie de la réponse impulsionnelle linéaire .
2. cb_1b vaut cb_1a mais sert à la convolution de la deuxième partie de la réponse impulsionnelle linéaire[5].
3. cb_2 (voir Table 9.2) contient les M_2 derniers échantillons de $x^2[n]$, où M_2 est la longueur du noyaux de Volterra d'ordre 2.
4. cb_3 (voir Table 9.3) contient les M_3 derniers échantillons de $x^3[n]$, où M_3 est la longueur du noyaux de Volterra d'ordre 3.

$x[n]$	$x[n-1]$	...	$x[n-M]$

Table 9.1 – cb_1a et cb_1b

$x^2[n]$	$x^2[n-1]$	...	$x^2[n-M_2]$

Table 9.2 – cb_2

$x^3[n]$	$x^3[n-1]$	...	$x^3[n-M_3]$

Table 9.3 – cb_3

La classe CirculaireBuffer implémente les méthodes suivantes :

- cbWrite qui permet de mettre un élément de type double dans le buffer circulaire,
- convTab_x qui permettent de convoluer les différents noyaux de Volterra avec les différents Buffers qui leurs sont associés.

9.3.2 Schéma de l'algorithme

Le schéma général de l'algorithme est présenté à la figure 9.1 et le principe de la fonction convTab_2 est présenté à la figure 9.2 et fonctionne ainsi :

ConvTab_2 doit réaliser l'opération suivante :

$$y_2[n] = h_2[n] \circledast x^2[n] = \sum_{\tau=0}^{M_2} h_2[\tau] x^2[n-\tau] \ \forall n \in [1, sampleFrames] \tag{9.4}$$

La fonction prend en entrée :

1. Le Buffer Frame envoyé par l'hôte.
2. Le buffer circulaire cb_2 contenant les M_2 derniers échantillons du signal d'entrée
3. Le noyau d'ordre 2 de Volterra

La fonction va calculer chaque $y_2[n] \ \forall \ n \in [1, sampleFrames]$ et renvoyer un tableau contenant tous ces $y_2[n]$ Remarquons que le thread est instancié une seule fois pour tout le Buffer Frame envoyé par l'hôte. En effet l'instanciation d'un thread est une opération trop coûteuse pour se servir d'un thread pour calculer un $y_2[n]$, attendre tous les threads, calculer la sortie $y[n] = y_1[n] + y_2[n] + y_3[n]$ puis recommencer avec y[n+1].

Pour chaque y[i] elle procède ainsi :

1. La fonction calcule $x^2[i]$ et place sa valeur sur le buffer circulaire cb_2.

5. Les buffers cb_1a et cb_1b sont équivalent mais sont envoyés à des Threads différents, cette façon de procéder permet de s'affranchir d'éventuel problème de mémoire partagée.

2. Elle multiplie ensuite terme à terme le noyau de Volterra d'ordre 2 avec le buffer.
3. Elle somme tous les produits dans une variable *Result*.
4. Finalement elle place la valeur de ce résultat dans un tableau ouptuts_2 constitués des y[i]
5. Une fois que tous les échantillons du Buffer Frame sont traités, la fonction renvoie le tableau outputs_2 et attend que les autres fonctions convTab aient fini leur convolution. (voir figure 9.1).

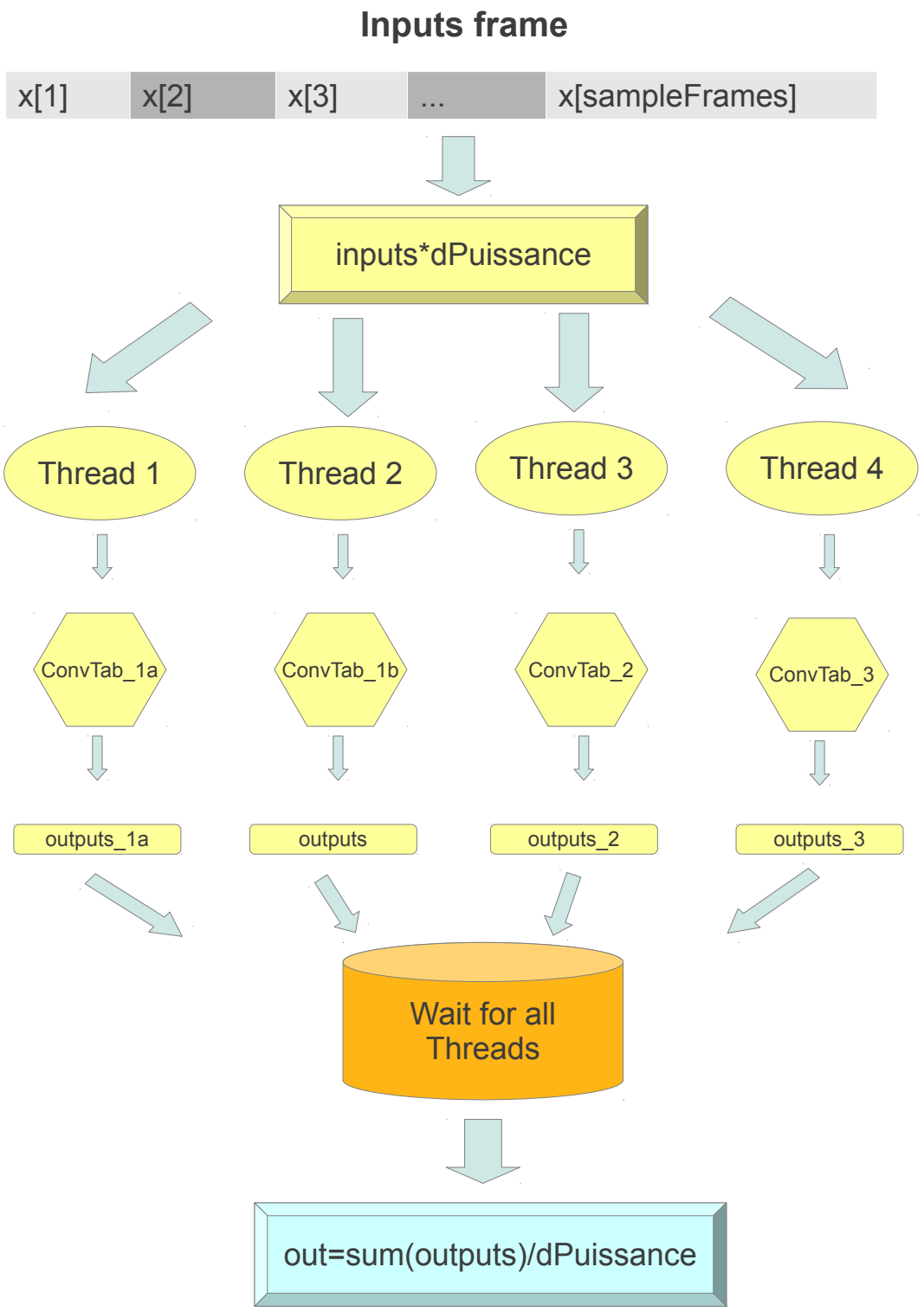


Figure 9.1 – Principe général de l'algorithme

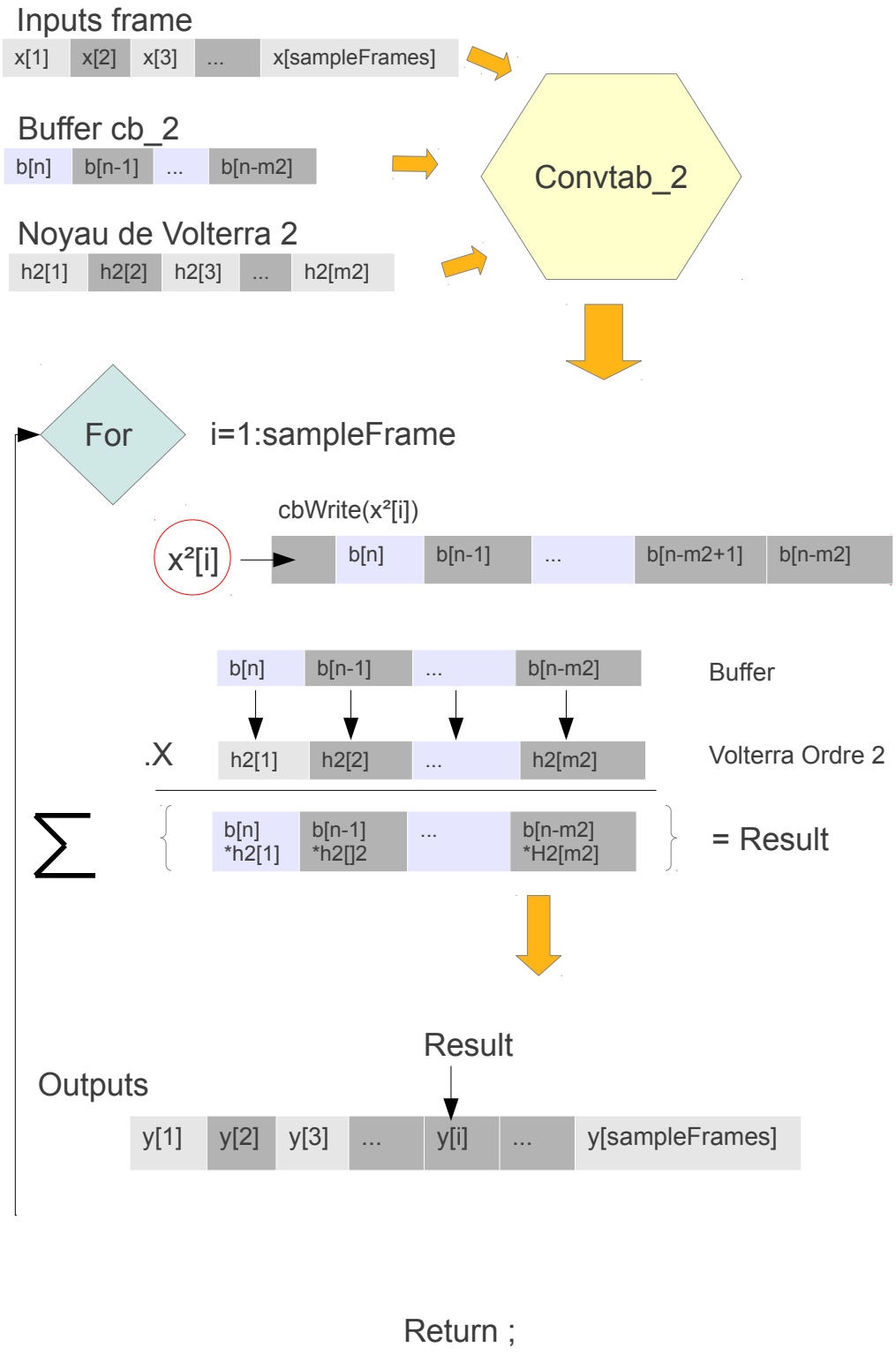


Figure 9.2 – Principe de la fonction ConvTab

9.4 Se servir du fichier ImpulsePlugin.txt pour créer un nouveau plugin

Nous avons vu dans l'Appendix A que Matlab générait un fichier texte contenant les noyaux de Volterra à insérer dans le code du plugin. Il suffit de copier le contenu du fichier texte (voir figure 9.3) et de les coller dans le fichier Convolution.cpp (voir figure 9.4) à la place du texte en rose.

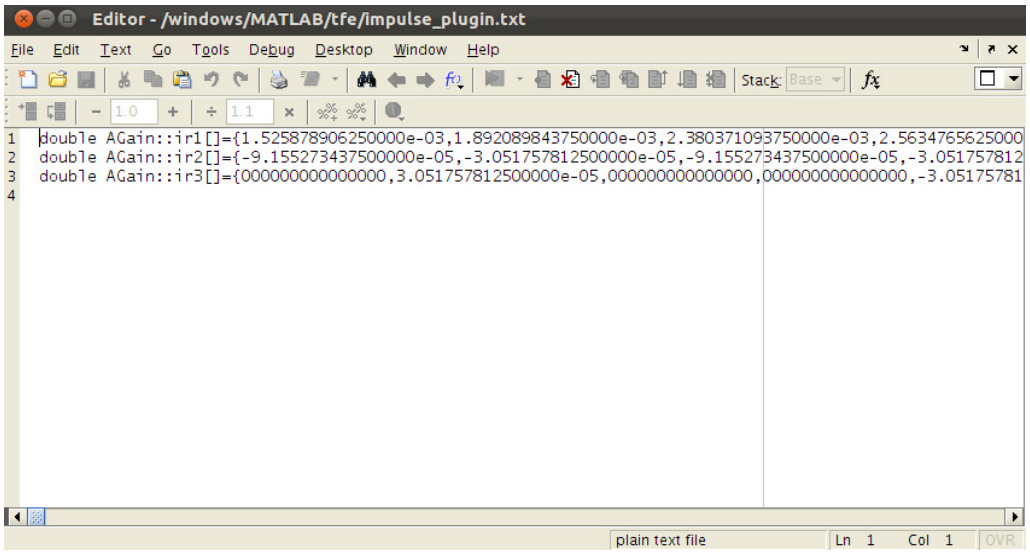


Figure 9.3 – Fichier ImpulsePlugin.txt

Figure 9.4 – Fichier Convolution.cpp

9.5 Longueurs des différents impulses

Comme nous l'avons vu au chapitre 5, la longueur de la réponse linéaire h1v nécessaire pour tromper l'oreille est de 3000 samples. Cette longueur n'est valable que pour des mesures effectuées en chambre anéchoïque. Cela assure une décroissance rapide de la réponse impulsionnelle.

Le temps Δ_M séparant l'ordre 1 de la réponse d'ordre M croit de façon logarithmique, c'est à dire que les temps séparant Δ_M et $\Delta_M + 1$ sont de plus en plus courts. Il est donc inutile de prendre un longueur fixe de 25000 samples pour tous les ordres de réponses impulsionnelles car :

1. Nous risquons de prendre une partie de la réponse impulsionnelle d'ordre M-1.
2. Nous risquons de prendre une partie des artéfacts.
3. Nous consommons un grande partie des ressources cpu pour des détails ; en effet, il vaut mieux assurer une forte précision pour la réponse impulsionnelle linéaire puisque la distorsion est censée être relativement faible par rapport aux fréquences fondamentales. Il vaut donc mieux attribuer 3000 ou 4000 samples au premier noyau de Volterra et répartir le reste des ressources pour les convolutions d'ordre secondaire.

   Finalement, nous avons choisi un rapport de 2.5 entre la longueur de la réponse impulsionnelle d'ordre 1 et celle d'ordre 2, ainsi qu'un rapport 5 entre la longueur de la réponse impulsionnelle d'ordre 1 et celle d'ordre 3. Ces rapports restent à optimiser.

10

Appendix C : développements mathématiques

Ce chapitre contiendra une partie des développements mathématiques allourdissant l'idée principale du texte mais qui peuvent faciliter la compréhension entre certaines étapes.

10.1 Développements mathématiques du chapitre 3

10.1.1 Obtention de la sortie du système (3.1) en page 22

La sortie du premier système linéaire vaut :

$$\begin{cases} w[n] = x[n] \circledast h_a[n] = \sum_{\tau_1=0}^{\infty} h_a[\tau_1]x[n-\tau_1] \end{cases} \tag{10.1}$$

La sortie du système non-linéaire :

$$\begin{cases} z[n] = a_1 w[n] + a_2 w^2[n] + a_3 w^3[n] + ... \\ \quad = a_1 \sum_{\tau_1=0}^{\infty} h_a[\tau_1]x[n-\tau_1] + a_2 \sum_{\tau_1=0}^{\infty} h_a[\tau_1]x[n-\tau_1] \sum_{\tau_2=0}^{\infty} h_a[\tau_2]x[n-\tau_2] \\ \quad + a_3 \sum_{\tau_1=0}^{\infty}\sum_{\tau_2=0}^{\infty}\sum_{\tau_3=0}^{\infty} h_a[\tau_1]x[n-\tau_1]h_a[\tau_2]x[n-\tau_2]h_a[\tau_3]x[n-\tau_3] \end{cases} \tag{10.2}$$

10.1.2 Cas particulier du système (3.2) page 22

Le système (3.2) est fort complexe, il fait intervenir des noyaux bi et tridimensionnels, le professeur Farina propose de poser $\tau_2 = \tau_1$ et de négliger les autres contributions. Une

autre possibilité,[1] est de considérer le cas où une autre valeur de $\tau_2 = \tau_1 + \Delta$ mène à une contribution significative du double produit $h_a[\tau_1]h_a[\tau_2]$

alors pour $\Delta > 0$, le système (3.2) devient :

$$\begin{cases} h_1[n] = a_1 h_a[n] \circledast h_b[n] \\ h_2[\tau_1', \tau_1'] = a_2 \sum_{k=0}^{\infty} h_b[k] h_a^2[\tau_1' - k] = a_2 h_a^2[n] \circledast h_b[n] \\ h_2[\tau_1', \tau_1' + \Delta] = a_2 \sum_{k=0}^{\infty} h_b[k] h_a[\tau_1' - k] h_a[\tau_1' + \Delta - k] \\ h_3[\tau_1', \tau_1', \tau_1'] = a_3 \sum_{k=0}^{\infty} h_b[k] h_a^3[\tau_1' - k] = a_3 h_a^3[n] \circledast h_b[n] \\ h_3[\tau_1', \tau_1', \tau_1' + \Delta] = a_3 \sum_{k=0}^{\infty} h_b[k] h_a^2[\tau_1' - k] h_a[\tau_1' + \Delta - k] \end{cases} \tag{10.3}$$

10.1.3 Obtention du système (10.12) de la page 113

En multipliant x[n] par lui même, nous trouvons :

$$\begin{cases} x^2[n] = A^2 \ \sin^2(\phi[n]) = \frac{A^2}{2}[1 - \cos(2\phi[n])] \\ x^3[n] = \frac{A^3}{2}[\sin(\phi[n]) - \cos(2\phi[n])\sin(\phi[n])] = \frac{A^3}{4}[3 \ sin(\phi[n]) - sin(3\phi[n])] \\ x^4[n] = \frac{A^4}{4}[3\sin^2(\phi[n]) - \sin(3\phi[n]\sin(\phi[n])] = \frac{A^4}{8}[3 - 4\cos(2\phi[n]) + \cos(4\phi[n])] \\ x^5[n] = \frac{A^5}{16}[10 sin(\phi[n]) - 5 sin(3\phi[n]) + sin(5\phi[n])] \end{cases} \tag{10.4}$$

Remarquons que :

$$M\phi[n] = \phi[n + \Delta_M] - (M-1)B \tag{10.5}$$

Où M est un entier et :

$$B = \frac{T w_1}{ln(\frac{w_2}{w_1})} \tag{10.6}$$

$$\Delta M = N \frac{ln(M)}{ln(\frac{w_2}{w_1})} \tag{10.7}$$

1. cette possibilité n'est pas prise au hasard et sera discutée à la section 3.2

En effet :

$$\begin{aligned} M\phi[n] &= B\left[M\left(\frac{w_2}{w_1}\right)^{\frac{n}{N}} - M\right] \\ &= B\left[\left(\frac{w_2}{w_1}\right)^{\frac{ln(M)}{ln(\frac{w_2}{w_1})}}\left(\frac{w_2}{w_1}\right)^{\frac{n}{N}} - M\right] \end{aligned} \tag{10.8}$$

Puisque :

$$M = \left(\frac{w_2}{w_1}\right)^{\frac{ln(M)}{ln\left(\frac{w_2}{w_1}\right)}}$$

Finalement :

$$\begin{aligned} M\phi[n] =& B\left[\left(\frac{w_2}{w_1}\right)^{\frac{n+\Delta_M}{N}} - 1\right] - B(M-1) \\ =& \phi[n+\Delta_M] - B(M-1) \end{aligned} \tag{10.9}$$

En appliquant dans le système (10.4) le résultat de l'équation (10.9), nous obtenons :

$$\begin{cases} x^2[n] = \dfrac{A^2}{2}[1 - \cos(\phi(n+\Delta_2) - B)] = \dfrac{A^2}{2}[1 - \cos(B)\cos(\phi[n+\Delta_2]) - \sin(B)\sin(\phi[n+\Delta_2])] \\ x^3[n] = \dfrac{A^3}{4}[3\ sin(\phi[n]) - sin(\phi[n+\Delta_3] - 2B)] \\ \qquad = \dfrac{A^3}{4}[3sin(\phi[n]) - \cos(2B)\sin(\phi[n+\Delta 3]) + sin(2B)\cos(\phi[n+\Delta_3])] \\ x^4[n] = \dfrac{A^4}{8}[3 - 4\cos(\phi[n+\Delta_2] - B) + \cos(\phi[n+\Delta_4] - 3B)] \\ \qquad = \dfrac{A^4}{8}[3 - 4cos(B)cos(\phi[n+\Delta_2]) - 4sin(B)sin(\phi[n+\Delta_2]) \\ \qquad\quad + cos(3B)cos(\phi[n+\Delta_4]) + sin(3B)sin(\phi[n+\Delta_4])] \\ x^5[n] = \dfrac{A^5}{16}[10sin(\phi[n]) - 5sin(\phi[n+\Delta_3] - 2B) + sin(\phi[n+\Delta_5] - 4B)] \\ \qquad = \dfrac{A^5}{16}[10sin(\phi[n]) - 5cos(2B)sin(\phi[n+\Delta_3]) + 5sin(2B)cos(\phi[n+\Delta_3]) \\ \qquad\quad + cos(4B)sin(\phi[n+\Delta_5]) - sin(4B)cos(\phi[n+\Delta_5])] \end{cases} \tag{10.10}$$

Note : Relation avec l'article [53] de Torras-Rosell et Jacobsen

Nous retrouvons l'équation (4) de l'article en remarquant que :

Si $\tau_1 = \tau_2$, alors :

$$\gamma_{22} = N\frac{ln\left[2\left(\frac{w_2}{w_1}\right)^{\frac{-\tau_1}{N}}\right]}{ln\left(\frac{w_2}{w_1}\right)} = N\frac{ln(2)}{ln\left(\frac{w_2}{w_1}\right)} - \tau_1 = \Delta_2 - \tau_1$$

et

$$\gamma_{20} = N\frac{ln(0)}{ln\left(\frac{w_2}{w_1}\right)} = -\infty \;\Rightarrow\; \phi[n+\gamma_{20}] \rightarrow\; -B$$

Impliquant que :

$$cos\left[\frac{Tw_1}{ln\left(\frac{w_2}{w_1}\right)}\right] cos(\phi[n+\gamma_{20}]) - sin\left[\frac{Tw_1}{ln\left(\frac{w_2}{w_1}\right)}\right] sin(\phi[n+\gamma_{20}]) \rightarrow\; cos^2(B)+sin^2(B) = 1$$

Fin de note

Dans l'équation (10.10), les termes en $\sin(\phi[n+\Delta_M])$ peuvent être transformés en $x[n+\Delta_M]$ mais pas les termes en cosinus. Les auteurs de l'article [53] proposent alors de transformer les cosinus en sinus en passant par la transformée de Hilbert :

La transformée de Fourier de $\sin(\phi(t))$:

$$\int_{-\infty}^{+\infty} sin(\phi(t))e^{-jwt}dt = \frac{1}{2j}\left[\int_{-\infty}^{+\infty} e^{j\phi(t)}e^{-jwt}dt - \int_{-\infty}^{+\infty} e^{-j\phi(t)}e^{-jwt}dt\right] = \frac{1}{2j}(X_A(w)-X_B(w))$$

La transformée de Fourier de $\cos(\phi(t))$:

$$\int_{-\infty}^{+\infty} cos(\phi(t))e^{-jwt}dt = \frac{1}{2}\left[\int_{-\infty}^{+\infty} e^{j\phi(t)}e^{-jwt}dt + \int_{-\infty}^{+\infty} e^{-j\phi(t)}e^{-jwt}dt\right] = \frac{1}{2}(X_A(w)+X_B(w))$$

Où :

- $X_A(w)$ est la transformée de Fourier de $e^{j\phi(t)}$
- $X_B(w)$ est la transformée de Fourier de $e^{-j\phi(t)}$

Comme $e^{j\phi(t)}$ ne contient que des fréquences positives (Sweep), alors $X_A(w) = 0 \;\forall\; w < 0$ de même $X_B(w) = 0 \;\forall\; w > 0$

Par conséquent, on peut écrire que :

$$Fourier(cos[\phi(t)]) = \frac{1}{2}(X_A(w) + X_B(w)) = Fourier(sin[\phi(t)]) * (-H_H(w))$$

Avec : $H_H(w) = -jsign(w)$ la fonction de transfert d'un filtre LTI appliquant la transformée de Hilbert et dont la réponse impulsionnelle est notée $h_H(t)$.[41]

Et donc :

$$cos(\phi[n]) = -sin(\phi[n]) \circledast h_H[n] \tag{10.11}$$

En appliquant cette relation aux système (10.10), nous obtenons :

$$\begin{cases} x^2[n] = \frac{A^2}{2}\left[1 + \cos(B)x[n+\Delta_2] \circledast h_H[n] - \sin(B)x[n+\Delta_2]\right] \\ x^3[n] = \frac{A^3}{4}[3sin(\phi[n]) - \cos(2B)x[n+\Delta_3] - sin(2B)x[n+\Delta_3] \circledast h_H[n]] \\ x^4[n] = \frac{A^4}{8}[3 + 4cos(B)x[n+\Delta_2] \circledast h_H[n] - 4sin(B)x[n+\Delta_2] \\ \qquad - cos(3B)x[n+\Delta_4] \circledast h_H[n] + sin(3B)x[n+\Delta_4]] \\ x^5[n] = \frac{A^5}{16}[10x[n] - 5cos(2B)x[n+\Delta_3] - 5sin(2B)x[n+\Delta_3] \circledast h_H[n] \\ \qquad + cos(4B)x[n+\Delta_5] + sin(4B)x[n+\Delta_5] \circledast h_H[n]] \end{cases} \tag{10.12}$$

Nous avons ainsi obtenu les différentes composantes x^2, x^3, x^4, x^5 exprimées en terme du signal envoyé x[n].

10.1.4 Développements des 2 premiers termes de l'équation (3.10) page 25

Le premier terme du second membre vaut :

$$h_1[n] \circledast x[n] \circledast f[n] = h_1[n] \circledast \delta[n] = h_1[n] \tag{10.13}$$

Le deuxième terme du second membre vaut :

$$\begin{cases} h_2[n] \circledast x^2[n] \circledast f[n] \\ = \frac{A}{2} h_2[n] \circledast [A \circledast f[n] + cos(B) A sin(\phi[n+\Delta_2]) \circledast f[n] \circledast h_H[n] - sin(B) A sin(\phi[n+\Delta_2]) \circledast f[n]] \\ = \frac{A}{2} h_2[n] \circledast [A \circledast f[n] + cos(B) x[n+\Delta_2] \circledast h_H[n] - sin(B) x[n+\Delta_2] \circledast f[n]] \\ = \frac{A}{2} h_2[n] \circledast [A \circledast f[n] + cos(B) h_H[n+\Delta_2] - sin(B) \delta[n+\Delta_2]] \end{cases}$$

10.1.5 Obtention de la transformée de Fourier des noyaux de Volterra en partant du système (3.12) page 26

La transformée de Fourier du système (3.12) vaut :

$$\begin{cases} H_1'(w) = H_1(w) + \frac{3A^2}{4} H3(w) + \frac{5A^4}{8} H_5(w) \\ H_2'(w) = \frac{A}{2} H_2(w)[cos(B) H_H(w) - sin(B)] \\ \qquad + \frac{A^3}{2} H_4(w)[cos(B) H_H(w) - sin(B)] \\ H_3'(w) = \frac{-A^2}{4} H_3(w)[cos(2B) + sin(2B) H_H(w)] \\ \qquad - \frac{5A^4}{16} H_5(w)[cos(2B) + sin(2B) H_H(w)] \\ H_4'(w) = \frac{A^3}{8}[-cos(3B) H_H(w) - sin(3B)] \\ H_5'(w) = \frac{A^4}{16} H_5(w)[cos(4B) + sin(4B) H_H(w)] \end{cases} \tag{10.14}$$

En réorganisant le système et en tenant compte que la transformée de Hilbert pour w>0 vaut $H_H(w) = -j$, nous obtenons les différents noyaux de Volterra :

$$\begin{cases} H_1(w) = H_1'(w) + 3e^{2jB} H'3(w) + 5e^{4jB} H_5'(w) \\ H_2(w) = \dfrac{2}{A}\, je^{jB}[H_2'(w) + 4e^{2jB} H'4(w)] \\ H_3(w) = \dfrac{-4}{A^2}\, e^{2jB}[H_3'(w) + 5e^{2jB} H_5'(w)] \\ H_4(w) = \dfrac{8}{A^3}\left(-je^{3jB}\right) H_4'(w) \\ H_5(w) = \dfrac{16}{A^4} H_5'(w) e^{4jB} \end{cases} \tag{10.15}$$

11

Appendix D : Appareils de mesures et datasheets

11.1 Description des appareils de mesure.

11.1.1 Audio-précision

L'audio-précision est un appareil de mesure permettant entre autres de mesurer la réponse fréquentielle d'un système à une excitation donnée. Dans ce travail,il sera notre référence et nous permettra de mesurer la réponse linéaire à un niveau donné (amplitude du signal d'excitation).

Quelques données :

- Analog sine sweep entre 10 Hz et 240 kHz
- précision de +- 0.06db à 1khz sur l'amplitude du signal
- THD+N résiduel dans la bande 20Hz-20kHz 0.0004%

L'audio-précision procède en *step-sine*, c'est-à-dire qu'il envoie un sinus à une certaine fréquence f_0, analyse le niveau en sortie du système puis envoie le sinus suivant à la fréquence $f_1 = f_0 - \Delta f$ (balayage de la fréquence la plus élevée vers la plus basse), Comme le montre la figure 11.1, le Δf (flèche rouge) diminue de manière logarithmique afin d'assurer un nombre de points constant dans chaque octave. Nous avons donc une diminution de la fréquence envoyée à chaque pas, plus importante en haute fréquence qu'en basse fréquence.

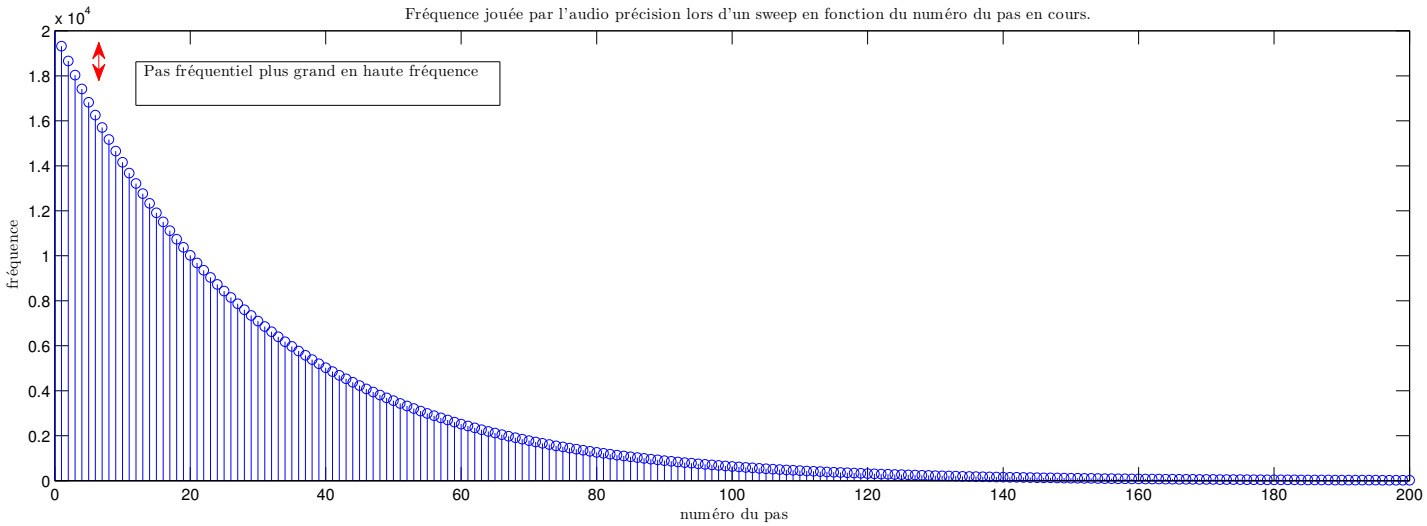


Figure 11.1 – Fréquence envoyée par l'audio précision en fonction du pas en cours

Figure 11.2 – Appareil de mesure "*Audio-précision*"

11.1.2 Ampli Kenwood [33]

L'amplificateur est un élément important dans la chaîne de mesure. En effet, il faut qu'ils soit "transparent" (réponse fréquentielle plate) dans la bande de fréquence que l'on cherche à caractériser pour ne pas influencer la réponse impulsionnelle du haut-parleur. Comme nous cherchons à caractériser les distorsions dues au haut-parleur à haute puissance, il faut aussi que l'ampli génère très peu de distorsion à haute puissance.

Nous avons testé plusieurs amplis pendant les laboratoires et le Kenwood était le seul capable de délivrer une puissance de 60W sur une charge de 16 ohms sans trop de distorsion. La figure 11.3 présente la courbe de réponse de l'ampli à basse puissance connecté sur le Hp éminence, une différence de 2 db existe entre les basses et hautes fréquences.

Par acquis de conscience, nous avons effectué la même mesure mais sur une charge

résistive de 16 ohms pour vérifier que la variation d'impédance du haut-parleur n'était pas en cause, le résultat est similaire (voir figure 11.4). Nous remarquerons un parasite non identifié à 60 hz et ses multiples.

Le signal réinjecté dans la carte son est pris aux bornes du haut-parleur comme le montre la figure 11.5.

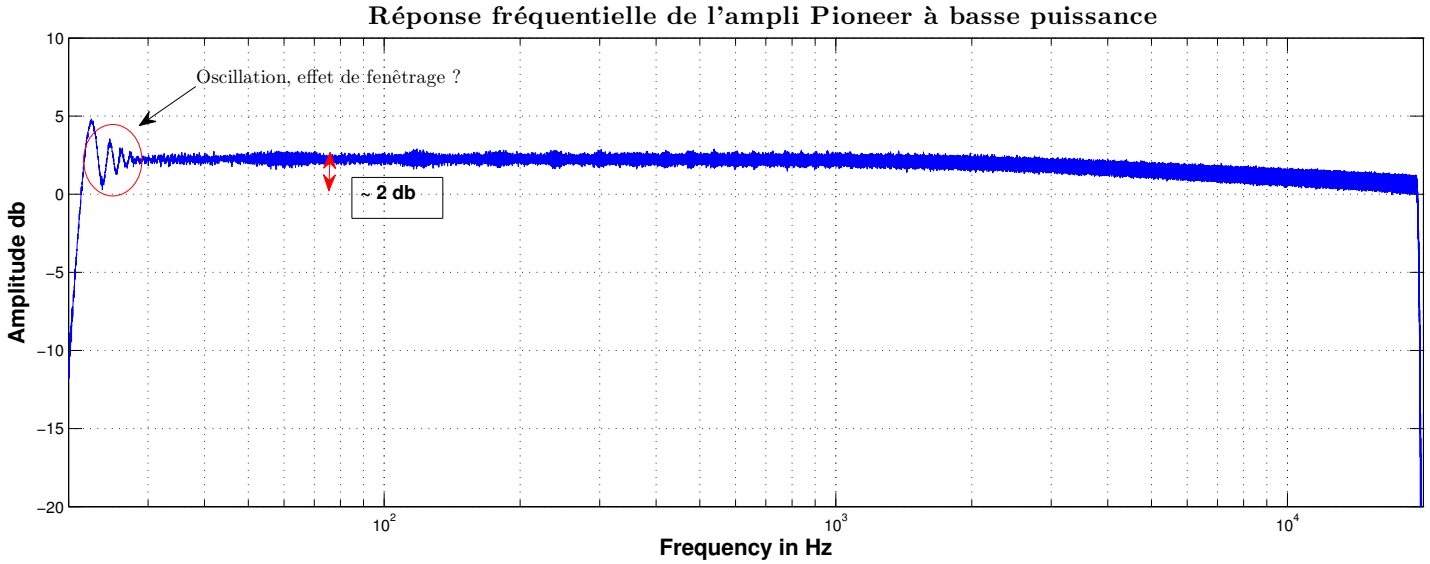


Figure 11.3 – Réponse fréquentielle de l'ampli Kenwood branché sur le haut-parleur Eminence Wizard

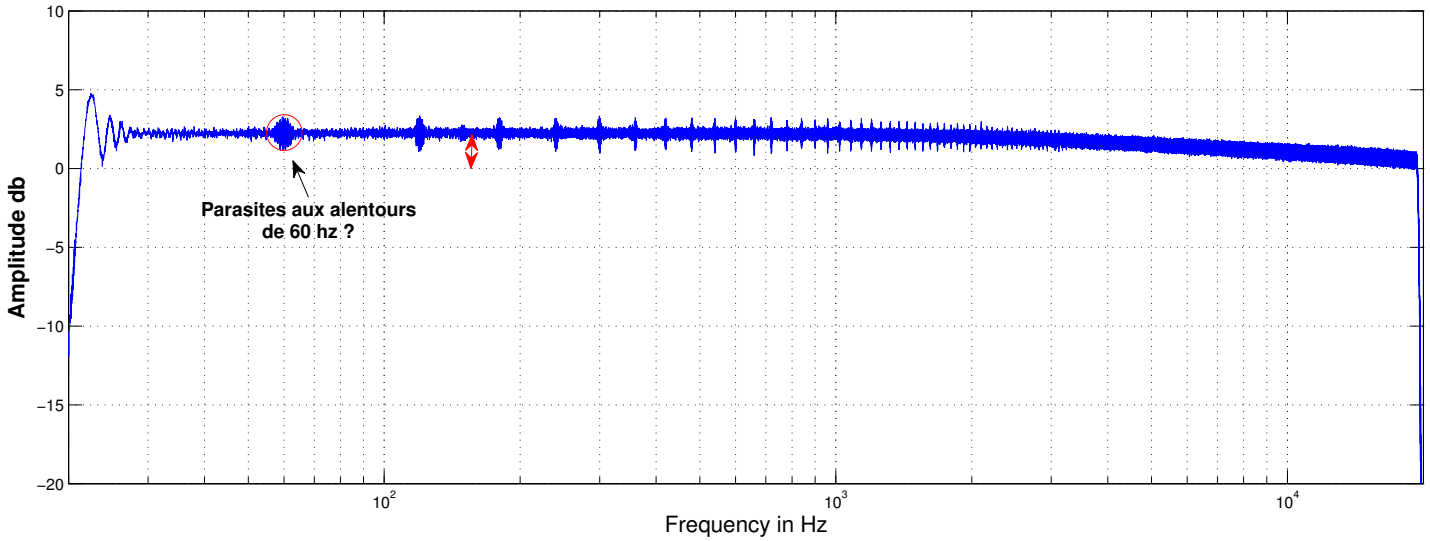


Figure 11.4 – Réponse fréquentielle de l'ampli Kenwood branché sur une charge résistive de 16 ohms

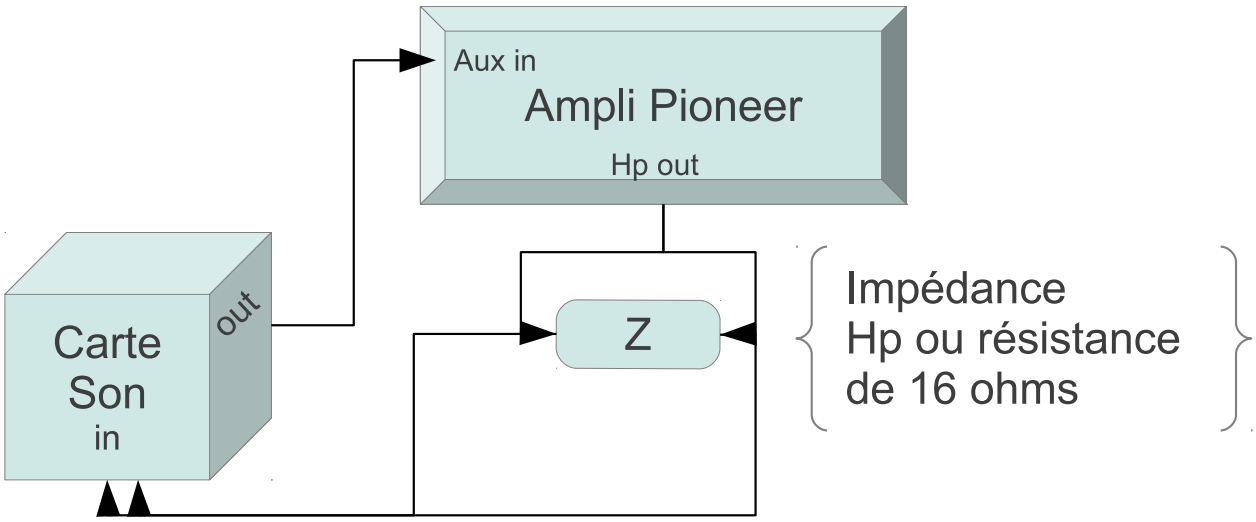


Figure 11.5 – Schéma pour l'évaluation de la réponse en fréquence de l'amplificateur Kenwood

11.1.3 Microphone MCE 320 [38]

Microphone metravib 0.1db suivit par un conditionneur de signal opus (0.1db fréquence de coupure 0.3hz)

11.1.4 Emulation de l'impédance du haut-parleur.

Comme nous l'avons vu pour l'ampli Kenwood, la réponse fréquentielle de l'amplificateur lui-même ne dépend pas tellement de l'impédance de la charge (comparer la figure 11.3 où l'impédance de sortie est une résistance de 16 ohms à la figure 11.4 où l'impédance de sortie est celle d'un haut-parleur de "16 ohms").

Pour un amplificateur à lampe de guitare, le comportement est un peu différent car la résistance d'anode de la dernière triode est remplacée par un transformateur en élévateur d'impédance dont le secondaire se connecte aux bornes du haut-parleur. La réponse fréquentielle est donc fortement dépendante de la charge connectée en sortie de l'ampli.

Il faut donc émuler l'impédance du haut-parleur le mieux possible afin de ne pas altérer la réponse fréquentielle de l'ampli.

Deux solutions sont envisageables :

1. Construire une mini chambre sourde, le haut-parleur serait placé dans une espèce de boite à fort coefficient d'absorbtion comme sur la figure 11.6
2. Simuler l'impédance du haut-parleur à l'aide d'un circuit RLC (voir figure 11.7 et 11.8)

Figure 11.6 – Isolation acoustique d'un haut-parleur dans une BOX

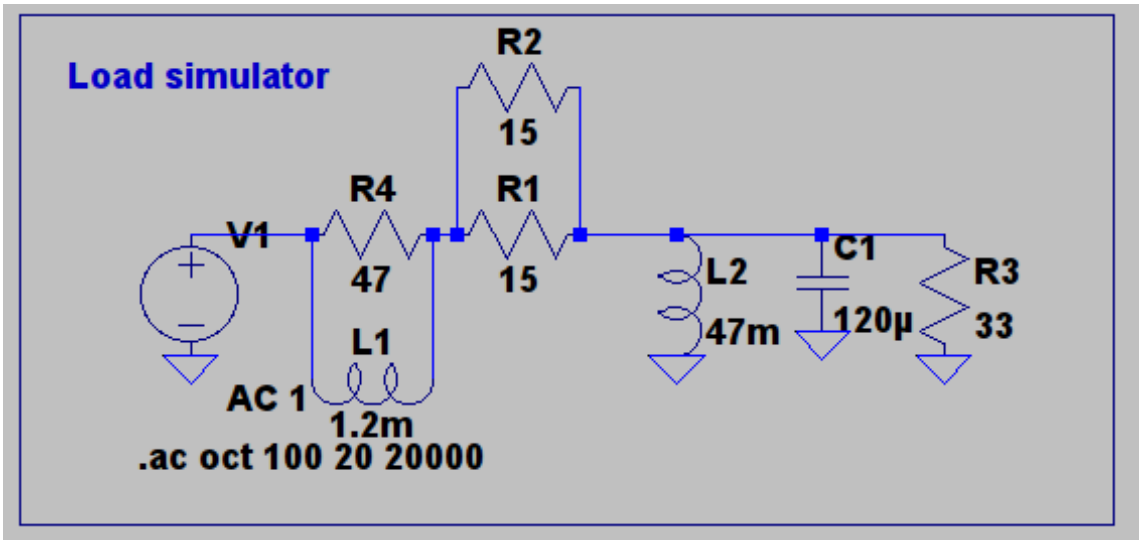


Figure 11.7 – Circuit RLC simulant l'impédance du haut-parleur

La première solution nous semble la plus fidèle, nous pensons également qu'il n'est pas nécéssaire de placer le même haut-parleur que celui que nous cherchons à simuler, dans la boite. Un haut-parleur "Modèle" conviendra pour tous, ceci reste néamoins à prouver.

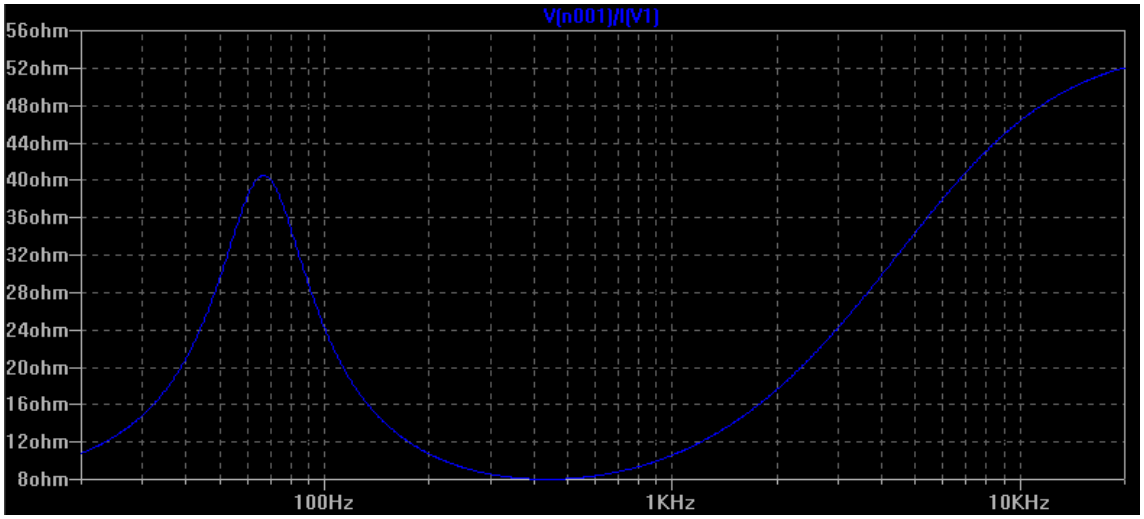


Figure 11.8 – Impédance du circuit RLC de la figure 11.7

Table des matières

1 Introduction **4**
1.1 Simulation d'un haut-parleur 4

2 Etat de l'art **7**
2.1 Le haut-parleur électrodynamique 7
2.1.1 Équations de fonctionnement [18],[16] 7
2.1.2 Non-linéarité du haut-parleur 9
2.1.2.1 Non-linéarité du champ magnétique B 9
2.1.2.2 Non-linéarité de la rigidité K_M des suspensions 9
2.1.2.3 Non-linéarité de l'inductance de la bobine 10
2.1.2.4 Non-linéarité de la résistance de la bobine 10
2.2 Modélisations des systèmes 10
2.2.1 Propriétés des systèmes 10
2.2.2 Produit de convolution 11
2.2.3 Réponse impulsionnelle 11
2.3 Obtention de la réponse impulsionnelle 12
2.3.1 Déconvolution 12
2.3.2 Sinesweep 14
2.3.2.1 Sweep linéaire 14
2.3.2.2 Sweep logarithmique 14
2.3.3 Maximum length sequence 16
2.4 Système non-linéaire et série de Volterra 16
2.5 Simulation d'un haut-parleur 18
2.5.1 Le simulateur analogique de haut-parleur 18
2.5.2 Le simulateur numérique à convolution de haut-parleur 19
2.5.3 Le simulateur de haut parleur non-linéaire 20

3 Modélisation non linéaire du haut parleur, étude théorique. **21**
3.1 Obtention des noyaux de Volterra 21
3.2 Artefact dans la réponse impulsionnelle mesurée 27
3.2.1 Cas général modèle Torras-Rossel et Jacobsen [53] 27

3.2.2 Position des répliques additionnelles . . . 27
3.2.3 Application des résultats de la section 3.2 à un cas concret . . . 29
3.3 Limitation des séries de Volterra . . . 33

4 Mesure de réponses impulsionnelles sur un haut-parleur 35
4.1 Enceinte à caractériser . . . 35
4.2 Système de mesure . . . 36
4.2.1 Carte son 1 : waveterminal 2496 [20] . . . 37
4.2.1.1 Comparaison Lin et Log sweep . . . 39
4.2.1.2 Le fenêtrage . . . 41
4.3 Reconstructions des noyaux de Volterra . . . 46
4.3.1 Les étapes de l'obtention des noyaux de Volterra. . . . 46
4.3.2 Objectif . . . 46
4.3.3 Reconstruction des noyaux de Volterra pour le haut-parleur Blackheart . . . 47
4.3.3.1 Isoler les noyaux mesurés h'_i . . . 53
4.3.3.2 Reconstruire les différents noyaux de Volterra . . . 53
4.3.3.3 Comparaison de la réponse linéaire reconstruite H1v et de la réponse bas niveau Hlin2V mesurée via Lin sweep 57
4.3.3.4 Comparaison de la réponse à haut niveau reconstruite avec la réponse haut niveau Hlin31V mesurée via Lin sweep . . . 57
4.3.3.5 Note : . . . 60
4.3.4 Haut-parleur Eminence Wizard . . . 61
4.3.4.1 Réponse impulsionnelle h(t) via sweep logarithmique . . 61
4.3.4.2 Comparaison des réponses à bas et haut niveau (Hlin2V et Hlin25V) . . . 61
4.3.4.3 Comparaison des réponses linéaires H1v et Hlin2V . . . 62
4.3.4.4 Comparaison des réponses à hauts niveaux Htotv et Hlin25V . . . 63

5 Résultat audio 66
5.1 Système d'écoute . . . 66
5.1.1 Loadbox . . . 66
5.1.2 Réponse en fréquence de la Loadbox . . . 68
5.1.3 Carte son 2 [27] . . . 70
5.1.4 Amplificateur guitare . . . 73
5.1.5 Casque Audio . . . 73
5.2 Schéma du système d'écoute . . . 73
5.3 Le plugin . . . 76
5.3.1 Reaper . . . 76
5.3.2 Virtual Studio Technology . . . 76
5.3.2.1 Les paramètres associés au plugin . . . 77
5.4 Tests Audio . . . 78
5.4.1 Test 1 : comparaison en direct de la simulation linéaire et non-linéaire du haut-parleur . . . 78
5.4.2 Test 2 : rendu de la simulation . . . 78

5.4.3 Test 3 : comparaison d'enregistrements effectués à l'aide du plugin linéaire et du plugin non-linéaire . . . 79
5.4.4 Longueur de la réponse impulsionnelle . . . 80
5.4.4.1 Second test sur le choix de longueur de la réponse impulsionnelle . . . 83

6 Futurs travaux et perspectives 86
6.1 Liste des non-idéalités . . . 86
6.1.1 Non idéalités du modèle de Volterra . . . 86
6.1.2 Non idéalités des mesures . . . 86
6.1.3 Non idéalités d'écoutes . . . 87
6.2 Etude d'optimisation . . . 87
6.2.1 SNR . . . 87
6.2.2 Longueur des réponses impulsionnelles pour le plugin . . . 87
6.3 Optimisation de la charge cpu . . . 87
6.4 Un modèle tenant compte des artéfacts . . . 88
6.5 D'autres façons de mesurer les noyaux de Volterra . . . 88
6.6 Comparaison Mécanique et noyaux de Volterra . . . 88
6.7 Amélioration du plugin . . . 88
6.8 La charge . . . 89

7 Conclusion 90

8 Appendix A : code Matlab 92
8.1 Description des fonctions Matlab . . . 92
8.2 Exemple d'extraction des noyaux de Volterra pour une utilisation dans le plugin . . . 94

9 Appendix B : code Plugin 100
9.1 Installation de la sdk . . . 100
9.2 Boost et multi thread . . . 101
9.3 Principe de l'algorithme . . . 101
9.3.1 Buffer circulaire . . . 102
9.3.2 Schéma de l'algorithme . . . 103
9.4 Se servir du fichier ImpulsePlugin.txt pour créer un nouveau plugin . . . 107
9.5 Longueurs des différents impulses . . . 108

10 Appendix C : développements mathématiques 109
10.1 Développements mathématiques du chapitre 3 . . . 109
10.1.1 Obtention de la sortie du système (3.1) en page 22 . . . 109
10.1.2 Cas particulier du système (3.2) page 22 . . . 109
10.1.3 Obtention du système (10.12) de la page 113 . . . 110
10.1.4 Développements des 2 premiers termes de l'équation (3.10) page 25 . . . 113
10.1.5 Obtention de la transformée de Fourier des noyaux de Volterra en partant du système (3.12) page 26 . . . 114

11 Appendix D : Appareils de mesures et datasheets **116**
11.1 Description des appareils de mesure. 116
11.1.1 Audio-précision 116
11.1.2 Ampli Kenwood [33] 117
11.1.3 Microphone MCE 320 [38] 119
11.1.4 Emulation de l'impédance du haut-parleur. 119

Table des figures

1.1 Photo d'un cabinet 4*12 5

2.1 Vue en coupe du haut-parleur électrodynamique 8
2.2 Transformée de Fourier de la réponse impulsionnelle d'un ballon de baudruche explosant dans une pièce (chambre à coucher). 13
2.3 Réponse impulsionnelle du haut parleur obtenue par déconvolution de la réponse du système lorsqu'on lui applique un sweep logarithmique de 20 secondes et application de la formule (2.20) 16
2.4 Simulateur de cab analogique [45] 18
2.5 Courbe de réponse fréquentielle du simulateur de cab analogique (trait plein) et courbe de la réponse fréquentielle du cabinet que les concepteurs cherchaient à simuler (pointillé) 19
2.6 Écran de contrôle pour la prise de nouveaux impulses dynamiques dans le plugin Nebula 20

3.1 Modèle non-linéaire numéro 1 du haut-parleur et de son espace acoustique. 21
3.2 Modèle non-linéaire numéro 2 du haut parleur. 23
3.3 Réponse impulsionnelle du haut-parleur obtenue en déconvoluant le signal de sortie du système lorsqu'on lui applique un sweep logarithmique à son entrée. 29
3.4 Fonction d'auto-corrélation de la réponse impulsionnelle linéaire du haut parleur. 31

4.1 La courbe noire représente la réponse en fréquence du haut-parleur "Wizard16" (voir condition de mesure dans la datasheet [1]) et la courbe grise, l'impédance du haut parleur. 36
4.2 Schéma de mesure pour la prise de réponses impulsionnelles 37
4.3 Réponse fréquentielle de la carte son Waveterminal (*type d'excitation : Logsweep* 38
4.4 Réponse fréquentielle de la carte son Waveterminal (*type d'excitation : sweep linéaire*) 39

4.5 Transformée de Fourier du Logsweep convolué avec son filtre inverse . . 40
4.6 Transformée de Fourier d'un sweep linéaire convolué avec son filtre inverse 40
4.7 Transformée de Fourier du Linsweep de durée T=100 secondes (soit une durée 5 fois plus importante par rapport au sweep utilisé pour la figure 4.6) convolué avec son filtre inverse. 41
4.8 Transformée de Fourier de la fonction rectangle, pour des longueurs de rectangles variant entre 2 et 6 secondes 43
4.9 Représentation temporelle et spectrale de quelques fenêtres communes . 44
4.10 Transformée de Fourier d'un sweep linéaire convolué avec son filtre inverse, avec fade in de 0.01 seconde au lieu de 0.1 s (comme à la figure 4.6). 45
4.11 Générateur de sweep Aurora dans adobe audition 45
4.12 Transformée de Fourier de la réponse impulsionnelle du haut-parleur Blackheart obtenue par déconvolution d'un sweep linéaire de bas et haut niveau appliqués en entrée. 48
4.13 Réponse impulsionnelle du système obtenue en déconvoluant un sweep logarithmique de 31 *Vrms* appliqué en entrée. 49
4.14 Sine sweep logarithmique et *fade in* 50
4.15 Filtre inverse du sine sweep logarithmique 50
4.16 Transformée de Fourier du filtre inverse d'un sweep logarithmique. 51
4.17 Réplique artéfact décalée de Δ_{22} (théorie de la section 3.2). 52
4.18 Réplique artéfact décalée de Δ_{20} (théorie de la section 3.2). 53
4.19 Comparaison des transformées de Fourier des signaux y(t) (simulation) et z(t) (mesure) 55
4.20 Algorithme pour la détermination du facteur A 56
4.21 Comparaison de : (1) La réponse fréquentielle reconstruite à bas niveau H1v via les noyaux de Volterra. (2) La transformée de Fourier de la réponse impulsionnelle du haut-parleur obtenue en déconvoluant la sortie du système lorsqu'on applique un sweep linéaire de bas niveau à son entrée. 57
4.22 Comparaison de : (1) la réponse fréquentielle reconstruite à haut (Hvtot) et bas niveau (H1v) via les noyaux de Volterra. (2) La transformée de Fourier de la réponse impulsionnelle du haut-parleur obtenue en déconvoluant la sortie du système lorsqu'on applique un sweep linéaire de haut niveau à son entrée. 58
4.23 Diminution de l'amplitude du signal récupéré par le microphone dans le temps (probablement du à l'échauffement de la bobine) pour un sinus à 1000hertz et d'amplitude 31Vrms 59
4.24 Réponse impulsionnelle du haut-parleur *Eminence Wizard* obtenue en déconvoluant un sweep logarithmique de 25 *Vrms* appliqué à son entrée. 61
4.25 Transformées de Fourier des réponses impulsionnelles du haut-parleur *Eminence Wizard* obtenues en déconvoluant les sweeps linéaires de 2 et 25 V appliqués à son entrée. 62
4.26 Comparaison des réponses fréquentielles du haut-parleur *Eminence Wizard* : H1v et Hlin2V 63
4.27 Comparaison des réponses fréquentielles : H1v,Htotv et Hlin25V 65

4.28 Réponses fréquentielles des différents noyaux de Volterra 65

5.1 Loadbox, schéma original de Claude Cavallo 67
5.2 Loadbox face avant : volume, switch inverseur de phase et les deux sorties jacks en symétriques 68
5.3 Loadbox face arrière : connecteur 220VAc, les 2 prises jacks sont montées en parallèles et sont à raccorder à la sortie de l'ampli et au haut-parleur (ou à une charge équivalente). 68
5.4 Réponse en fréquence de la loadbox connectée au haut-parleur (sweep amplifié avec ampli Kenwood). 69
5.5 Réponse en fréquence de la loadbox connectée à une charge résistive de 16 ohms (sweep amplifié avec ampli Kenwood). 70
5.6 Carte son 2 à base de pcm2912 71
5.7 Circuit de la carte son 2 71
5.8 Schéma électrique de la carte son 2 72
5.9 Réponse en fréquence de la carte son 2 72
5.10 Amplificateur guitare 50W 73
5.11 Diagramme de fonctionnement des tests acoustiques. 75
5.12 Plugin de simulation non linéaire du cab Blackheart (interface graphique) 77
5.13 Plugin de simulation linéaire du cab blackheart (interface graphique) . . 78
5.14 Différences perçues par des guitaristes entre le sample 1 et les autres samples 82
5.15 Transformée de Fourier de la réponse impulsionnelle du haut-parleur tronquée après 16000 échantillons 84
5.16 Transformée de Fourier de la réponse impulsionnelle du haut-parleur tronquée après 3100 échantillons 84
5.17 Transformée de Fourier de la réponse impulsionnelle du haut-parleur tronquée après 246 échantillons 85

9.1 Principe général de l'algorithme 105
9.2 Principe de la fonction ConvTab 106
9.3 Fichier ImpulsePlugin.txt 107
9.4 Fichier Convolution.cpp 107

11.1 Fréquence envoyée par l'audio précision en fonction du pas en cours . . . 117
11.2 Appareil de mesure "*Audio-précision*" 117
11.3 Réponse fréquentielle de l'ampli Kenwood branché sur le haut-parleur Eminence Wizard 118
11.4 Réponse fréquentielle de l'ampli Kenwood branché sur une charge résistive de 16 ohms 118
11.5 Schéma pour l'évaluation de la réponse en fréquence de l'amplificateur Kenwood 119
11.6 Isolation acoustique d'un haut-parleur dans une BOX 120
11.7 Circuit RLC simulant l'impédance du haut-parleur 120
11.8 Impédance du circuit RLC de la figure 11.7 121

Bibliographie

[1] *The wizard 16, eminence speaker*. www.eminence.com/pdf/Wizard_16.pdf.

[2] Abel Jonathan S. ; Bryan Nicholas J. ; Kolar Miriam A. Impulse response measurements in the presence of clock drift. In *Audio Engineering Society Convention 129*, 11 2010.

[3] David P. Abel, Jonathan S. ; Berners. A technique for nonlinear system measurement. In *Audio Engineering Society Convention 121*, 10 2006.

[4] A.Farina. *AURORA plug-ins*. http://pcfarina.eng.unipr.it/Aurora_XP/index.htm.

[5] Finn T. Agerkvist. Volterra series based distortion effect. In *Audio Engineering Society Convention 129*, 11 2010.

[6] F.T. Agerkvist. Modelling loudspeaker non-linearities. In *Proc. AES 32nd AES Conf. DSP for loudspeakers, Hillerød Denmark*, 2007.

[7] E. Armelloni, A. Farina, and A. Bellini. Non-linear convolution : a new approach for the auralization of distorting systems. In *Proc. AES 110th Conv*, 2001.

[8] Pingshun Bai, Mingsian R. ; Zeung. Fast convolution technique using a nonuniform sampling scheme : Algorithm and applications in audio signal processing. *J. Audio Eng. Soc*, 52(5) :496–505, 2004.

[9] Delphine Bard. Horn loudspeakers nonlinearity comparison and linearization using volterra series. In *Audio Engineering Society Convention 124*, 5 2008.

[10] Göran Bard, Delphine ; Sandberg. Modeling of nonlinearities in electrodynamic loudspeakers. In *Audio Engineering Society Convention 123*, 10 2007.

[11] J. Borish and J.B. Angell. An efficient algorithm for measuring the impulse response using pseudorandom noise. *J. Audio Eng. Soc*, 31(7) :478–488, 1983.

[12] S. Boyd, YS Tang, and L. Chua. Measuring volterra kernels. *Circuits and Systems, IEEE Transactions on*, 30(8) :571–577, 1983.

[13] L.N.O. Chua and Y. Liao. Measuring volterra kernels (ii). *International journal of circuit theory and applications*, 17(2) :151–190, 1989.

[14] D. Ciric, A. Pantic, and D. Radulovic. Transient noise effects in measurement of room impulse response by swept sine technique. In *Telecommunication in Modern Satellite Cable and Broadcasting Services (TELSIKS), 2011 10th International Conference on*, pages 269–272. IEEE, 2011.

[15] Stan Guy-Bart ; Embrechts Jean-Jacques ; Archambeau Dominique. Comparison of different impulse response measurement techniques. *J. Audio Eng. Soc*, 50(4) :249–262, 2002.

[16] J.J. Embrechts. Introduction aux techniques audio et vidéo. pages 38–41, 2006.

[17] J.J. Embrechts. Acoustique appliquée : fondements et lutte contre le bruit. 2008.

[18] J.J. Embrechts. Electroacoustique,chapitre 4. 2008.

[19] Blackheart Eminence. *bh112, guitar speaker cabinet*. http://www.blackhearteng.com/pdf/BH112_OM.pdf.

[20] esi audio. *Carte son, waveterminal 2496*. http://www.esi-audio.com/products/wt2496/.

[21] A. Farina. Simultaneous measurement of impulse response and distortion with a swept-sine technique. *Preprints-Audio Engineering Society*, 2000.

[22] A. Farina. Advancements in impulse response measurements by sine sweeps. In *122th AES Convention, Vienna, Austria*, pages 5–8, 2007.

[23] A. Farina. Silence sweep : a novel method for measuring electro-acoustical devices. In *126th Convention of the Aud. Eng. Soc.(2009 May)*, 2009.

[24] W. Frank, R. Reger, and U. Appel. Loudspeaker nonlinearities-analysis and compensation. In *Signals, Systems and Computers, 1992. 1992 Conference Record of The Twenty-Sixth Asilomar Conference on*, pages 756–760. IEEE, 1992.

[25] W.G. Gardner. Efficient convolution without input-output delay. *Journal of the Audio Engineering Society*, 43(3) :127–136, 1995.

[26] Andrei Alexandrescu Herb Sutter. *Boost, C++ librairies*. http://www.boost.org/.

[27] Texas Instruments. *AUDIO CODEC WITH USB INTERFACE, MONO MICROPHONE INPUT AND STEREO HEADPHONE OUTPUT*. http://focus.ti.com/lit/ds/symlink/pcm2912.pdf.

[28] Texas Instruments. *Dual Low-Noise JFET-Input General-Purpose Operational Amplifier*. http://www.ti.com/product/tl072.

[29] Frantisek Kadlec. A new method for identification of nonlinear systems using miso model with swept-sine technique : Application to loudspeaker analysis. In *Audio Engineering Society Convention 124*, 5 2008.

[30] A.J.M. Kaizer. Modeling of the nonlinear response of an electrodynamic loudspeaker by a volterra series expansion. *J. Audio Eng. Soc*, 35(6) :421–433, 1987.

[31] Harel Kemp, Jonathan ; Primack. Impulse response measurement of nonlinear systems : Properties of existing techniques and wide noise sequences. *J. Audio Eng. Soc*, 59(12) :953–963, 2011.

[32] Michael J. Kemp. Analysis and simulation of analogue dynamic compressors and limiters in the digital domain. In *Audio Engineering Society Convention 109*, 9 2000.

[33] Kenwood. *Manual Kenwood Amplifier ka-3300d*. http://audiohoho.wordpress.com/2011/11/01/kenwood-ka-3300d-manual-specifications/.

[34] W. Klippel. Nonlinear large-signal behavior of electrodynamic loudspeakers at low frequencies. *J. Audio Eng. Soc*, 40(6) :483–496, 1992.

[35] W. Klippel. Distortion analyzer-a new tool for assessing and improving electrodynamic transducer. *PREPRINTS-AUDIO ENGINEERING SOCIETY*, 2000.

[36] W. Klippel. Prediction of speaker performance at high amplitudes. *PREPRINTS-AUDIO ENGINEERING SOCIETY*, 2001.

[37] W. Klippel. Loudspeaker nonlinearities-causes, parameters, symptoms. *J. Audio Eng. Soc*, 54(10) :907–939, 2006.

[38] Metravib. *Microphone type MCE 320, metravib 01db*. http://www.scribd.com/doc/95435196/fr-MCE320-specifs1.

[39] C. Müller-Tomfelde. Low-latency convolution for real-time applications. In *Proc. Audio Eng. Soc. Int. Conference*, pages 454–459, 1999.

[40] ni.com. *Optimiser les FFT grâce aux fonctions de fenêtrage*. http://www.ni.com/white-paper/4844/fr.

[41] A.V. Oppenheim, A.S. Willsky, and S.H. Nawab. *Signals and systems*, volume 2. Prentice-Hall Englewood Cliffs, NJ, 1983.

[42] M.A. Poletti. Linearly swept frequency measurements, time-delay spectrometry, and the wigner distribution. *J. AES*, 36 :457–468, 1988.

[43] M.J. Reed and M.O. Hawksford. Practical modelling of nonlinear audio systems using the volterra series. *PREPRINTS-AUDIO ENGINEERING SOCIETY*, 1996.

[44] MJ Reed and MOJ Hawksford. Identification of discrete volterra series using maximum length sequences. In *Circuits, Devices and Systems, IEE Proceedings-*, volume 143, pages 241–248. IET, 1996.

[45] runoffgroove. simulateur de cab analogique , le condor. http://www.runoffgroove.com/condor.html.

[46] T.E. Rutt and R.B. NJ. Vacuum tube triode nonlinearity as part of the electric guitar sound. In *Audio Engineering Society ; the 76th Convention*, 1984.

[47] M. Schetzen. The volterra and wiener theories of nonlinear systems. 1980.

[48] Thomas Schmitz. *CD-rom, contenant le code Matlab, le code C++ du plugin, le rapport du TFE au format pdf, les samples audio.*

[49] Sennheiser. *Casque Sennheiser HD25-1*. http://www.sennheiser.be/belux/home_fr.nsf/root/private_headphones_dj-headphones_502188.

[50] R. Sepulchre. Modélisation et analyse des systèmes. 2007-2008.

[51] Steinberg. *Vst, Virtual Studio Technology*. http://www.steinberg.net/en/home.html.

[52] A. Torras Rosell and F. Jacobsen. Measuring long impulse responses with pseudo-random sequences and sweep signals. In *39th Int. Congress on Noise Control Eng., Proceedings*, 2010.

[53] A. Torras-Rosell and F. Jacobsen. A new interpretation of distortion artifacts in sweep measurements. *Journal of the Audio Engineering Society*, 59(5), 2011.

[54] Lamberto Tronchin. Non linear convolution and its application to audio effects. In *Audio Engineering Society Convention 131*, 10 2011.

[55] Vanderbemden. Analyse et conception des systèmes de mesures électriques. pages 204–205, 2007.