

Modélisation et Simulation d'une corde de guitare

Tableau des Matières

1. **Introduction**
 - 1.1. Contexte et objectifs du projet
 - 1.2. Présentation générale de la démarche
2. **Modélisation Physique de la Corde**
 - 2.1. Hypothèses de base et équation des ondes
 - 2.2. Modes propres et fréquence fondamentale
 - 2.3. Impact des paramètres physiques (tension, masse linéique)
3. **Modélisation de la Corde Réelle : Raideur et Amortissements**
 - 3.1. Intégration des termes de raideur et d'amortissement
 - 3.2. Analyse de Von Neumann et conditions de stabilité
 - 3.3. Détermination des paramètres d'amortissement (0 et 1)
4. **Discretisation et Schéma Numérique Implicite**
 - 4.1. Discretisation spatiale et temporelle
 - 4.2. Rutriment par différences finies (formules dérivées aux secondes et quatrièmes)
 - 4.3. Traitement des conditions aux limites et points fantômes
 - 4.4. Formulation matricielle du schéma (matrices A, B, C)
 - 4.5. Expression des matrices via les œssiens Dxx et Dxxxx
5. **Paramétrage et Analyse du modèle**
 - 5.1. Calcul des paramètres d'amortissement à partir de la perte de la justice
 - 5.2. Discussion sur l'influence des interventions sur la vibration et le timbre
6. **Conclusion et perspectives**

Annexes

A. Simulation Complète et Restitution Sonore (Question 6)

- A.1. Code Scilab complet de la simulation
- A.2. Explication et utilisation du programme (animation du profil de la corde, enregistrement stéréo, restitution sonore)

B. Analyse Spectre et Étude Qualitative du Son (Question 7)

- B.1. Code Scilab complet pour l'analyse FFT
- B.2 Explication et utilisation du programme d'analyse (transformée de Fourier, identification de la fréquence fondamentale, interprétation du spectre)

Introduction

Le présent projet vise à explorer la modélisation et la simulation numérique de la vibration d'une corde de guitare. L'objectif principal est de concevoir un outil numérique capable de reproduire, de manière fidèle et interactive, le comportement acoustique d'une corde vibrante, depuis sa déformation initiale (pincement) jusqu'à la génération du son audible. Pour ce faire, nous utilisons l'environnement Scilab, qui offre des outils puissants pour la résolution d'équations aux dérivées partielles et l'animation de phénomènes dynamiques.

Dans un premier temps, la corde est modélisée de façon idéale en considérant une équation des ondes simplifiée. Cette approche permet d'établir les modes propres de vibration ainsi que

la fréquence fondamentale en fonction des paramètres physiques tels que la tension, la longueur et la masse linéique. Par la suite, le modèle est enrichi en intégrant des phénomènes réels tels que la raideur de la corde et les amortissements dus aux frottements et aux processus dissipatifs internes. Ces améliorations conduisent à une équation aux dérivées partielles plus complexe, dont la stabilité est étudiée via l'analyse de Von Neumann.

La discrétisation spatiale et temporelle de la corde, associée à un schéma implicite par différences finies, permet de simuler l'évolution du profil de la corde au fil du temps. Parallèlement, l'enregistrement du signal aux positions de plusieurs micros – modélisés à l'aide de techniques d'interpolation – offre une restitution sonore fidèle du phénomène. L'analyse spectrale du son produit, notamment par transformée de Fourier, permet de vérifier la correspondance entre la fréquence fondamentale simulée et la note attendue, tout en mettant en évidence l'influence des paramètres d'inharmonicité et de la position de pincement sur le timbre.

Ce travail s'inscrit dans une démarche pédagogique visant à familiariser les étudiants aux défis liés à la simulation numérique de systèmes vibratoires complexes, tout en intégrant une approche pluridisciplinaire allant de la modélisation physique à l'analyse acoustique. Au-delà de la validation théorique des résultats obtenus, le projet offre également une perspective pratique pour la conception d'instruments virtuels, où l'optimisation des paramètres de simulation permettrait d'affiner la qualité sonore et l'authenticité de l'instrument.

2 .Modélisation physique de la corde

Hypothèses de modélisation (corde idéale vs corde réelle)

Cas supposé sans raideur et de diamètre nul

Modes propres de la vibration d'une corde

On cherche une solution de l'équation (1) sous la forme $u(x, t) = U(x) \cos(\omega t)$.

On part de l'équation des ondes pour la corde :

$$\frac{\partial^2 u}{\partial t^2} = \gamma^2 \frac{\partial^2 u}{\partial x^2} \quad \text{avec} \quad \gamma = \sqrt{\frac{T}{\mu}}.$$

Hypothèse de séparation des variables

On propose une solution sous la forme :

$$u(x, t) = U(x) \cos(\omega t)$$

où $U(x)$ est la partie spatiale et $\cos(\omega t)$ la partie temporelle.

Insertion dans l'équation des ondes

Calculons les dérivées :

- La dérivée seconde par rapport au temps :

$$\frac{\partial^2 u}{\partial t^2} = -\omega^2 U(x) \cos(\omega t).$$

- La dérivée seconde par rapport à x :

$$\frac{\partial^2 u}{\partial x^2} = U''(x) \cos(\omega t).$$

En insérant ces expressions dans l'équation des ondes, on obtient :

$$-\omega^2 U(x) \cos(\omega t) = \gamma^2 U''(x) \cos(\omega t).$$

Puisque $\cos(\omega t)$ n'est pas identiquement nul, on peut le simplifier :

$$U''(x) + \frac{\omega^2}{\gamma^2} U(x) = 0.$$

Résolution de l'équation différentielle pour $U(x)$

L'équation différentielle obtenue :

$$U''(x) + k^2 U(x) = 0, \quad \text{avec} \quad k = \frac{\omega}{\gamma},$$

est une équation linéaire homogène à coefficients constants.

La solution générale est :

$$U(x) = A \cos(kx) + B \sin(kx),$$

où A et B sont des constantes déterminées par les conditions aux limites.

Application des conditions aux limites

La corde est fixée aux deux extrémités, ce qui impose :

$$u(0,t)=0 \quad \text{et} \quad u(L,t)=0.$$

À $x=0$

On a :

$$u(0,t) = U(0) \cos(\omega t) = A \cos(0) + B \sin(0) = A.$$

Pour que $u(0,t)=0$ pour tout t, il faut que :

$$A = 0$$

Donc, la solution se réduit à :

$$U(x) = B \sin(kx)$$

À $x=L$

On impose :

$$u(L, t) = U(L) \cos(\omega t) = B \sin(kL) \cos(\omega t) = 0 \text{ pour tout } t.$$

Pour avoir une solution non triviale (avec $B \neq 0$), il faut que :

$$\sin(kL) = 0$$

Cette condition se réalise lorsque :

$$\sin(kL) = 0$$

Ainsi, k doit vérifier :

$$k = \frac{n\pi}{L}.$$

Détermination des fréquences propres.

Rappelons que $K = \frac{\omega}{y}$, En substituant :

$$\frac{\omega}{y} = \frac{n\pi}{L} \Rightarrow \omega_n = \frac{n\pi y}{L}$$

La fréquence associée à un mode est donnée par :

$$f_n = \frac{\omega_n}{2\pi} = \frac{n\pi y}{2\pi L} = \frac{n}{2L} y = \frac{n}{2L} \sqrt{\frac{T}{\mu}}.$$

Les modes propres de la corde sont donc :

$$U_n(x) = B_n \sin\left(\frac{n\pi x}{L}\right),$$

et la solution générale (par principe de superposition) est :

$$u(x, t) = \sum_{n=1}^{+\infty} B_n \sin\left(\frac{n\pi x}{L}\right) \cos\left(\frac{n\pi y t}{L}\right).$$

Les coefficients B_n sont déterminés par la condition initiale $u(x, 0)$ via le développement en série de Fourier :

$$B_n = \frac{2}{L} \int_0^L u(x, 0) \sin\left(\frac{n\pi x}{L}\right) dx$$

Étude de la fréquence fondamentale f_1

La fréquence fondamentale correspond au mode $n = 1$:

$$f_1 = \frac{1}{2L} \sqrt{\frac{T}{\mu}}.$$

Analysons comment f_1 varie en fonction des paramètres :

Variation avec la longueur L :

- Si la longueur L **augmente**, alors f_1 **diminue** . Cela signifie que la note devient plus grave.
- Si la longueur est **divisée par 2**, alors f_1 **double**. La note devient donc plus aiguë.

Variation avec la masse linéique μ :

- Si μ **diminue**, l'expression $\sqrt{\frac{T}{\mu}}$ **augmente**, donc f_1 **augmente**. La corde plus légère vibre plus rapidement et produit une note plus aiguë.

Variation avec la tension T :

- Si T **augmente**, alors $\sqrt{\frac{T}{\mu}}$ **augmente**, et ainsi f_1 **augmente**. Une plus grande tension entraîne une vibration plus rapide, ce qui correspond à une note plus aiguë.

3. Cas de la corde réelle avec raideur et amortissements

Approximation de la dérivée seconde spatiale

On souhaite montrer que :

$$\frac{\partial^2 u}{\partial x^2}(x, t) = \frac{u(x+h, t) - 2u(x, t) + u(x-h, t)}{h^2} + o(1).$$

Procédons avec le développement en série de Taylor :

Pour $u(x+h, t)$ on a :

$$u(x+h, t) = u(x, t) + h u_x(x, t) + \frac{h^2}{2} u_{xx}(x, t) + \frac{h^3}{6} u_{xxx}(x, t) + \frac{h^4}{24} u_{xxxx}(x, t) + o(h^4).$$

De même, pour $u(x-h, t)$:

$$u(x-h, t) = u(x, t) - h u_x(x, t) + \frac{h^2}{2} u_{xx}(x, t) - \frac{h^3}{6} u_{xxx}(x, t) + \frac{h^4}{24} u_{xxxx}(x, t) + o(h^4).$$

En sommant, les termes u_x et u_{xxx} s'annulent :

$$u(x+h, t) + u(x-h, t) = 2u(x, t) + h^2 u_{xx}(x, t) + \frac{h^4}{12} u_{xxxx}(x, t) + o(h^4).$$

Isolons alors $u_{xx}(x, t)$:

$$u_{xx}(x, t) = \frac{u(x+h, t) - 2u(x, t) + u(x-h, t)}{h^2} - \frac{h^2}{12} u_{xxxx}(x, t) + o(h^2).$$

À l'ordre dominant (lorsque h est petit), on obtient :

$$\frac{\partial^2 u}{\partial x^2}(x, t) \approx \frac{u(x+h, t) - 2u(x, t) + u(x-h, t)}{h^2}.$$

En notation discrète, si on note :

$u(x_l, t_k) \approx u_l^n$ avec $x_l = l h$ et $t_k = n k$, on a :

$$\frac{\partial^2 u}{\partial x^2}(x_l, t_k) \approx \frac{u_{l+1}^n - 2u_l^n + u_{l-1}^n}{h^2}.$$

Approximation de la dérivée seconde temporelle

De même, en appliquant la formule de Taylor en t (pour k petit) :

$$u(x, t+k) = u(x, t) + k u_t(x, t) + \frac{k^2}{2} u_{tt}(x, t) + \frac{k^3}{6} u_{ttt}(x, t) + \frac{k^4}{24} u_{tttt}(x, t) + o(k^4),$$

$$u(x, t-k) = u(x, t) - k u_t(x, t) + \frac{k^2}{2} u_{tt}(x, t) - \frac{k^3}{6} u_{ttt}(x, t) + \frac{k^4}{24} u_{tttt}(x, t) + o(k^4).$$

En sommant, les termes impairs s'annulent :

$$u(x, t+k) + u(x, t-k) = 2u(x, t) + k^2 u_{tt}(x, t) + o(k^4).$$

D'où

$$\frac{\partial^2 u}{\partial t^2}(x, t) \approx \frac{u(x, t+k) - 2u(x, t) + u(x, t-k)}{k^2}.$$

En notation discrète :

$$\frac{\partial^2 u}{\partial t^2}(x_l, t_k) \approx \frac{u_l^{n+1} - 2u_l^n + u_l^{n-1}}{k^2}.$$

Approximation de la dérivée quatrième spatiale

Utilisons à nouveau le développement de Taylor pour :

$$u(x+2h, t) \text{ et } u(x-2h, t).$$

Développements en Taylor

Pour $u(x + 2h, t)$:

$$u(x + 2h, t) = u(x, t) + 2h u_x + 2h^2 u_{xx} + \frac{4}{3}h^3 u_{xxx} + \frac{2}{3}h^4 u_{xxxx} + o(h^4).$$

De même pour $u(x - 2h, t)$:

$$u(x - 2h, t) = u(x, t) - 2h u_x + 2h^2 u_{xx} - \frac{4}{3}h^3 u_{xxx} + \frac{2}{3}h^4 u_{xxxx} + o(h^4).$$

Calculons la combinaison suivante :

$$S = u(x + 2h, t) + u(x - 2h, t) - 4[u(x + h, t) + u(x - h, t)] + 6u(x, t).$$

Nous avons déjà établi que :

$$u(x + h, t) + u(x - h, t) = 2u(x, t) + h^2 u_{xx}(x, t) + \frac{h^4}{12} u_{xxxx}(x, t) + o(h^4).$$

De plus,

$$u(x + 2h, t) + u(x - 2h, t) = 2u(x, t) + 4h^2 u_{xx}(x, t) + \frac{4}{3}h^4 u_{xxxx}(x, t) + o(h^4).$$

Ainsi,

$$\begin{aligned} S &= \left[2u + 4h^2 u_{xx} + \frac{4}{3}h^4 u_{xxxx} \right] - 4 \left[2u + h^2 u_{xx} + \frac{h^4}{12} u_{xxxx} \right] + 6u \\ &= 2u + 4h^2 u_{xx} + \frac{4}{3}h^4 u_{xxxx} - 8u - 4h^2 u_{xx} - \frac{h^4}{3} u_{xxxx} + 6u \\ &= [2u - 8u + 6u] + [4h^2 u_{xx} - 4h^2 u_{xx}] + \left[\frac{4}{3}h^4 u_{xxxx} - \frac{h^4}{3} u_{xxxx} \right] \\ &= h^4 u_{xxxx}(x, t) + o(h^4). \end{aligned}$$

Ainsi,

$$\frac{\partial^4 u}{\partial x^4}(x, t) \approx \frac{u(x + 2h, t) - 4u(x + h, t) + 6u(x, t) - 4u(x - h, t) + u(x - 2h, t)}{h^4}.$$

En notation discrète :

$$\frac{\partial^4 u}{\partial x^4}(x_l, t_k) \approx \frac{u_{l+2}^n - 4u_{l+1}^n + 6u_l^n - 4u_{l-1}^n + u_{l-2}^n}{h^4}.$$

Approximation de la dérivée mixte $\frac{\partial}{\partial t} \left(\frac{\partial^2 u}{\partial x^2} \right)$

L'approximation spatiale est :

$$\frac{\partial^2 u}{\partial x^2}(x, t) \approx \frac{u(x+h, t) - 2u(x, t) + u(x-h, t)}{h^2}.$$

Pour toute fonction $f(t)$, on a :

$$\frac{df}{dt}(t) \approx \frac{f(t+k) - f(t-k)}{2k} + o(1).$$

Appliquons ceci à $f(t) = \frac{\partial^2 u}{\partial x^2}(x, t)$ on obtient :

$$\begin{aligned} \frac{\partial}{\partial t} \left(\frac{\partial^2 u}{\partial x^2} \right)(x, t) &\approx \frac{1}{2k} \left[\frac{u(x+h, t+k) - 2u(x, t+k) + u(x-h, t+k)}{h^2} \right. \\ &\quad \left. - \frac{u(x+h, t-k) - 2u(x, t-k) + u(x-h, t-k)}{h^2} \right] \\ &= \frac{1}{2k h^2} \left[(u(x+h, t+k) - 2u(x, t+k) + u(x-h, t+k)) \right. \\ &\quad \left. - (u(x+h, t-k) - 2u(x, t-k) + u(x-h, t-k)) \right]. \end{aligned}$$

En notation discrète, on a :

$$\boxed{\frac{\partial}{\partial t} \left(\frac{\partial^2 u}{\partial x^2} \right)(x_l, t_k) \approx \frac{1}{2k h^2} \left[(u_{l+1}^{n+1} - 2u_l^{n+1} + u_{l-1}^{n+1}) - (u_{l+1}^{n-1} - 2u_l^{n-1} + u_{l-1}^{n-1}) \right].}$$

Constitution du schéma implicite pour l'EDP (2)

L'équation complète est :

$$\frac{\partial^2 u}{\partial t^2} = \gamma^2 \frac{\partial^2 u}{\partial x^2} - \kappa^2 \frac{\partial^4 u}{\partial x^4} - 2\sigma_0 \frac{\partial u}{\partial t} + 2\sigma_1 \frac{\partial^3 u}{\partial t \partial x^2}.$$

Remplacer chaque terme par son approximation discrète

Dérivée seconde en temps :

$$\frac{\partial^2 u}{\partial t^2}(x_l, t_k) \approx \frac{u_l^{n+1} - 2u_l^n + u_l^{n-1}}{k^2}.$$

Dérivée seconde en espace :

$$\frac{\partial^2 u}{\partial x^2}(x_l, t_k) \approx \frac{u_{l+1}^n - 2u_l^n + u_{l-1}^n}{h^2}.$$

Dérivée quatrième en espace :

$$\frac{\partial^4 u}{\partial x^4}(x_l, t_k) \approx \frac{u_{l+2}^n - 4u_{l+1}^n + 6u_l^n - 4u_{l-1}^n + u_{l-2}^n}{h^4}.$$

Première dérivée en temps :

$$\frac{\partial u}{\partial t}(x_l, t_k) \approx \frac{u_l^{n+1} - u_l^{n-1}}{2k}.$$

Dérivée mixte $\frac{\partial}{\partial t}(\frac{\partial^2 u}{\partial x^2})$:

D'après la section précédente :

$$\frac{\partial}{\partial t} \left(\frac{\partial^2 u}{\partial x^2} \right) (x_l, t_k) \approx \frac{1}{2k h^2} \left[(u_{l+1}^{n+1} - 2u_l^{n+1} + u_{l-1}^{n+1}) - (u_{l+1}^{n-1} - 2u_l^{n-1} + u_{l-1}^{n-1}) \right].$$

Discrétisation de l'EDP

En multipliant l'équation par k^2 pour alléger l'écriture, on regroupe les termes selon le niveau temporel ($n+1$, n , $n-1$) :

Terme en u^{n+1}

Du terme $\frac{\partial^2 u}{\partial t^2} : +u_l^{n+1}$.

Du terme d'amortissement $-2\sigma_0 \frac{\partial u}{\partial t}$:

$$-2\sigma_0 \times \frac{k}{2} (u_l^{n+1} - u_l^{n-1}) = -\sigma_0 k u_l^{n+1} + \sigma_0 k u_l^{n-1}.$$

Du terme mixte $+2\sigma_1 \frac{\partial^3 u}{\partial t \partial x^2}$:

$$+2\sigma_1 \times \frac{1}{2k h^2} (u_{l+1}^{n+1} - 2u_l^{n+1} + u_{l-1}^{n+1} - [u_{l+1}^{n-1} - 2u_l^{n-1} + u_{l-1}^{n-1}]).$$

Pour le niveau $n + 1$, cela apporte :

$$+ \frac{\sigma_1}{k h^2} (u_{l+1}^{n+1} - 2u_l^{n+1} + u_{l-1}^{n+1}).$$

Regroupant, on définit :

$$a_1 = -\frac{\sigma_1 k}{h^2}, \quad a_2 = 1 + \sigma_0 k + \frac{2\sigma_1 k}{h^2},$$

ce qui permet d'écrire, pour le niveau $n+1$:

$$a_1 u_{l-1}^{n+1} + a_2 u_l^{n+1} + a_1 u_{l+1}^{n+1}.$$

Terme u^n

Du terme $\frac{\partial^2 u}{\partial t^2} : -2u_l^n.$

Du terme spatial $\gamma^2 \frac{\partial^2 u}{\partial x^2} :$
 $+ \gamma^2 k^2 \frac{u_{l+1}^n - 2u_l^n + u_{l-1}^n}{h^2}.$

Du terme de raideur $-\kappa^2 \frac{\partial^4 u}{\partial x^4} :$
 $-\kappa^2 k^2 \frac{u_{l+2}^n - 4u_{l+1}^n + 6u_l^n - 4u_{l-1}^n + u_{l-2}^n}{h^4}.$

On regroupe ces contributions sous les coefficients :

$$b_1 = \frac{\kappa^2 k^2}{h^4}, \quad b_2 = -\frac{\gamma^2 k^2}{h^2} - \frac{4\kappa^2 k^2}{h^4}, \quad b_3 = -2 + \frac{2\gamma^2 k^2}{h^2} + \frac{6\kappa^2 k^2}{h^4}.$$

Ainsi, le terme de niveau n s'écrit :

$$b_1 u_{l-2}^n + b_2 u_{l-1}^n + b_3 u_l^n + b_2 u_{l+1}^n + b_1 u_{l+2}^n.$$

Terme en u^{n-1}

De façon analogue, en tenant compte des contributions de $\frac{\partial^2 u}{\partial t^2}$, du terme d'amortissement et du terme mixte, on obtient :

$$c_1 = \frac{\sigma_1 k}{h^2}, \quad c_2 = 1 - \sigma_0 k - \frac{2\sigma_1 k}{h^2}.$$

Ce qui donne, pour le niveau n-1 :

$$c_1 u_{l-1}^{n-1} + c_2 u_l^{n-1} + c_1 u_{l+1}^{n-1}.$$

Le schéma implicite global s'écrit alors :

$$\boxed{a_1 u_{l-1}^{n+1} + a_2 u_l^{n+1} + a_1 u_{l+1}^{n+1} + b_1 u_{l-2}^n + b_2 u_{l-1}^n + b_3 u_l^n + b_2 u_{l+1}^n + b_1 u_{l+2}^n + c_1 u_{l-1}^{n-1} + c_2 u_l^{n-1} + c_1 u_{l+1}^{n-1} = 0,}$$

avec les coefficients :

- Pour $n + 1$:

$$a_1 = -\frac{\sigma_1 k}{h^2}, \quad a_2 = 1 + \sigma_0 k + \frac{2\sigma_1 k}{h^2},$$

- Pour n :

$$b_1 = \frac{\kappa^2 k^2}{h^4}, \quad b_2 = -\frac{\gamma^2 k^2}{h^2} - \frac{4\kappa^2 k^2}{h^4}, \quad b_3 = -2 + \frac{2\gamma^2 k^2}{h^2} + \frac{6\kappa^2 k^2}{h^4},$$

- Pour $n - 1$:

$$c_1 = \frac{\sigma_1 k}{h^2}, \quad c_2 = 1 - \sigma_0 k - \frac{2\sigma_1 k}{h^2}.$$

4. Discrétisation et Schéma Numérique Implicite

Élimination des points fantômes

La discrétisation se fait sur les points x_l pour $l=0,1,\dots,N$ avec les conditions aux limites.

$$u_0^n = u_N^n = 0.$$

De plus, on impose que la dérivée seconde en x soit nulle aux extrémités :

$$\frac{\partial^2 u}{\partial x^2}(0, t) = 0 \quad \text{et} \quad \frac{\partial^2 u}{\partial x^2}(L, t) = 0.$$

À gauche, en $x=0$

On approxime la dérivée seconde au point $x=0$ par :

$$u_{xx}(0, t) \approx \frac{u(x_1, t) - 2u(0, t) + u(x_{-1}, t)}{h^2}.$$

Comme $u(0, t) = 0$ et que l'on impose $u_{xx}(0, t) = 0$, on obtient :

$$\frac{u_1^n + u_{-1}^n}{h^2} = 0 \quad \implies \quad u_{-1}^n = -u_1^n.$$

À droite, en $x=L$

De même, en $x=L$

$$u_{xx}(L, t) \approx \frac{u(x_{N+1}, t) - 2u(x_N, t) + u(x_{N-1}, t)}{h^2}.$$

Ici, $u_N^n = 0$ et $u_{xx}(L, t) = 0$ donnent :

$$\frac{u_{N+1}^n + u_{N-1}^n}{h^2} = 0 \implies u_{N+1}^n = -u_{N-1}^n.$$

Pour chaque niveau temporel n ,

$$a_1 u_{l-1}^{n+1} + a_2 u_l^{n+1} + a_1 u_{l+1}^{n+1} + b(\text{termes en } u^n) + c_1 u_{l-1}^{n-1} + c_2 u_l^{n-1} + c_1 u_{l+1}^{n-1} = 0,$$

avec des coefficients issus de la discrétisation des dérivées en t et x.

Les expressions faisant intervenir u_{-1}^n et u_{N+1}^n se transforment en :

$$u_{-1}^n = -u_1^n \quad \text{et} \quad u_{N+1}^n = -u_{N-1}^n.$$

On peut écrire le système sous la forme

$$A u^{n+1} + B u^n + C u^{n-1} = 0,$$

Où les matrices A, B et C sont de taille $(N+1) \times (N+1)$

Les matrices proposées dans l'énoncé sont donc :

$$A = \begin{pmatrix} 1 & 0 & 0 & 0 & \cdots & 0 \\ a_1 & a_2 & a_1 & 0 & \cdots & 0 \\ 0 & a_1 & a_2 & a_1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & a_1 & a_2 & a_1 \\ 0 & \cdots & 0 & 0 & 0 & 1 \end{pmatrix},$$

$$C = \begin{pmatrix} 1 & 0 & 0 & 0 & \cdots & 0 \\ c_1 & c_2 & c_1 & 0 & \cdots & 0 \\ 0 & c_1 & c_2 & c_1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & c_1 & c_2 & c_1 \\ 0 & \cdots & 0 & 0 & 0 & 1 \end{pmatrix},$$

$$B = \begin{pmatrix} 1 & 0 & 0 & 0 & \dots & 0 \\ b_2 & b_3 - b_1 & b_2 & b_1 & 0 & \dots \\ b_1 & b_2 & b_3 & b_2 & b_1 & \dots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & b_1 & b_2 & b_3 - b_1 & b_2 \\ 0 & \dots & 0 & 0 & 0 & 1 \end{pmatrix}.$$

5. Paramétrage et Analyse du Modèle

On considère la version réduite du vecteur solution,

$$u^n = \begin{pmatrix} u_1^n \\ u_2^n \\ \vdots \\ u_{N-1}^n \end{pmatrix},$$

puisque $u_0^n = u_N^n = 0$.

Par ailleurs, on définit la matrice de différences centrées secondes (de type Toeplitz)

$$D_{xx} = \frac{1}{h^2} \begin{pmatrix} -2 & 1 & 0 & \dots & 0 \\ 1 & -2 & 1 & \ddots & \vdots \\ 0 & 1 & -2 & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & 1 \\ 0 & \dots & 0 & 1 & -2 \end{pmatrix},$$

et on pose

$$D_{xxxx} = D_{xx}D_{xx},$$

Expression matricielle des sous-matrices

Matrice A

Dans le schéma, les coefficients pour les termes en u^{n+1} (après élimination des points fantômes) sont donnés par

- $a_1 = -\frac{\sigma_1 k}{h^2}$ (sur les sous-diagonales)
- $a_2 = 1 + \sigma_0 k + \frac{2\sigma_1 k}{h^2}$ (sur la diagonale).

Maintenant, considérons la combinaison

$$(1 + \sigma_0 k)I - \sigma_1 k D_{xx}.$$

Sur la diagonale de D_{xx} , on a $-2/h^2$. Ainsi, la composante diagonale de $-\sigma_1 k D_{xx}$ vaut

$$-\sigma_1 k \left(-\frac{2}{h^2} \right) = \frac{2\sigma_1 k}{h^2}.$$

En ajoutant la contribution de $(1 + \sigma_0 k)I$, la diagonale devient

$$1 + \sigma_0 k + \frac{2\sigma_1 k}{h^2},$$

ce qui est exactement a_2 .

De même, sur les positions hors diagonale (voisins immédiats), la valeur de D_{xx} vaut $1/h^2$ et la contribution est

$$-\sigma_1 k \cdot \frac{1}{h^2} = -\frac{\sigma_1 k}{h^2},$$

soit a_1 .

D'où nous obtenons :

$$A = (1 + \sigma_0 k)I - \sigma_1 k D_{xx}.$$

Matrice C

De façon analogue, les coefficients pour C sont :

- $c_1 = \frac{\sigma_1 k}{h^2}$
- $c_2 = 1 - \sigma_0 k - \frac{2\sigma_1 k}{h^2}$.

Alors, en considérant la combinaison

$$(1 - \sigma_0 k)I + \sigma_1 k D_{xx} ,$$

La contribution sur la diagonale (avec D_{xx} diagonale $-2/h^2$) est :

$$(1 - \sigma_0 k) + \sigma_1 k \left(-\frac{2}{h^2} \right) = 1 - \sigma_0 k - \frac{2\sigma_1 k}{h^2} ,$$

et sur les sous-diagonales la contribution est :

$$\sigma_1 k \cdot \frac{1}{h^2} = \frac{\sigma_1 k}{h^2} .$$

Ce qui montre que :

$$C = (1 - \sigma_0 k)I + \sigma_1 k D_{xx} .$$

Matrice B

Les coefficients pour les termes en u^n dans le schéma (après modifications aux bords) sont tels que, dans la partie centrale, on a :

- $b_1 = \frac{\kappa^2 k^2}{h^4}$ sur les diagonales secondaires (deux cases de part et d'autre),
- $b_2 = -\frac{\gamma^2 k^2}{h^2} - \frac{4\kappa^2 k^2}{h^4}$ sur les premières sous-diagonales,
- $b_3 = -2 + \frac{2\gamma^2 k^2}{h^2} + \frac{6\kappa^2 k^2}{h^4}$ sur la diagonale principale (avec une légère modification aux bords qui se retrouve dans la formule « réduite »).

De plus, $D_{xxxx} = D_{xx}^2$ est exactement l'opérateur correspondant à la formule

$$\frac{u_{l+2} - 4u_{l+1} + 6u_l - 4u_{l-1} + u_{l-2}}{h^4} .$$

Ainsi, considérons l'expression

$$-2I - \gamma^2 k^2 D_{xx} + \kappa^2 k^2 D_{xxxx} .$$

- La contribution diagonale de $-\gamma^2 k^2 D_{xx}$ est

$$-\gamma^2 k^2 \left(-\frac{2}{h^2} \right) = \frac{2\gamma^2 k^2}{h^2}.$$

- La contribution diagonale de $\kappa^2 k^2 D_{xxxx}$ est

$$\kappa^2 k^2 \left(\frac{6}{h^4} \right) = \frac{6\kappa^2 k^2}{h^4}.$$

- En ajoutant le terme $-2I$, la diagonale devient

$$-2 + \frac{2\gamma^2 k^2}{h^2} + \frac{6\kappa^2 k^2}{h^4},$$

ce qui correspond exactement à b_3 .

Pour la première sous-diagonale, D_{xx} donne

$$1/h^2 \text{ (contribution } -\gamma^2 k^2/h^2) \text{ et } D_{xxxx} \text{ donne } -4/h^4$$

et donc

$$-\frac{\gamma^2 k^2}{h^2} - \frac{4\kappa^2 k^2}{h^4},$$

ce qui correspond à b_2

Nous avons donc :

$$B = -2I - \gamma^2 k^2 D_{xx} + \kappa^2 k^2 D_{xxxx}.$$

Injection de la solution dans l'EDP

On part de l'EDP (2) :

$$\frac{\partial^2 u}{\partial t^2} = \gamma^2 \frac{\partial^2 u}{\partial x^2} - \kappa^2 \frac{\partial^4 u}{\partial x^4} - 2\sigma_0 \frac{\partial u}{\partial t} + 2\sigma_1 \frac{\partial^3 u}{\partial t \partial x^2}.$$

On cherche une solution de la forme

$$\tilde{u}(x, t) = e^{st+j\beta x}.$$

Calculons les dérivées :

Première et seconde dérivées par rapport au temps :

$$\frac{\partial \tilde{u}}{\partial t} = s e^{st+j\beta x}, \quad \frac{\partial^2 \tilde{u}}{\partial t^2} = s^2 e^{st+j\beta x}.$$

Dérivées par rapport à x :

$$\frac{\partial \tilde{u}}{\partial x} = j\beta e^{st+j\beta x}, \quad \frac{\partial^2 \tilde{u}}{\partial x^2} = -\beta^2 e^{st+j\beta x},$$

$$\frac{\partial^4 \tilde{u}}{\partial x^4} = \beta^4 e^{st+j\beta x}.$$

Dérivée mixte (on dérive d'abord par x puis par t ou inversement) :

$$\frac{\partial^3 \tilde{u}}{\partial t \partial x^2} = \frac{\partial}{\partial t} (-\beta^2 e^{st+j\beta x}) = -\beta^2 s e^{st+j\beta x}.$$

En injectant ces expressions dans l'EDP, et en divisant par est $e^{st+j\beta x}$ (non nul), on obtient :

$$s^2 = -\gamma^2 \beta^2 - \kappa^2 \beta^4 - 2\sigma_0 s + 2\sigma_1 (-\beta^2 s).$$

Autrement dit

$$s^2 + [2\sigma_0 + 2\sigma_1 \beta^2] s + (\gamma^2 \beta^2 + \kappa^2 \beta^4) = 0.$$

Analyse et développement en supposant σ_0, σ_1 petits

On considère l'équation quadratique en s :

$$s^2 + 2(\sigma_0 + \sigma_1 \beta^2) s + (\gamma^2 \beta^2 + \kappa^2 \beta^4) = 0.$$

Les solutions sont

$$s = \frac{-2(\sigma_0 + \sigma_1 \beta^2) \pm \sqrt{4(\sigma_0 + \sigma_1 \beta^2)^2 - 4(\gamma^2 \beta^2 + \kappa^2 \beta^4)}}{2},$$

Ce qui s'écrit

$$s = -(\sigma_0 + \sigma_1 \beta^2) \pm \sqrt{(\sigma_0 + \sigma_1 \beta^2)^2 - (\gamma^2 \beta^2 + \kappa^2 \beta^4)}.$$

Pour σ_0 et σ_1 petits, on peut écrire l'approximation petits,

$$s_{\pm} \approx -(\sigma_0 + \sigma_1 \beta^2) \pm j \sqrt{\gamma^2 \beta^2 + \kappa^2 \beta^4 - (\sigma_0 + \sigma_1 \beta^2)^2}.$$

Ainsi, on définit

- La partie réelle (la décroissance) :

$$\sigma(\beta) = -\sigma_0 - \sigma_1 \beta^2,$$

- La pulsation propre :

$$\omega(\beta) = \sqrt{\gamma^2 \beta^2 + \kappa^2 \beta^4 - (\sigma_0 + \sigma_1 \beta^2)^2}.$$

Pour de petites pertes, le terme $(\sigma_0 + \sigma_1 \beta^2)^2$ est négligeable devant $\gamma^2 \beta^2 + \kappa^2 \beta^4$, ce qui donne :

$$s_{\pm} \approx \sigma(\beta) \pm j \omega(\beta).$$

Expression de σ en fonction de la fréquence

$$\sigma(\omega) = -\sigma_0 - \sigma_1 \xi(\omega),$$

Avec

$$\xi(\omega) = -\gamma^2 + \frac{\sqrt{\gamma^4 + 4\kappa^2 \omega^2}}{2\kappa^2}.$$

Montrer enfin que pour 2 fréquences données $\omega_1 < \omega_2$ on a

$$\sigma_0 = \frac{6 \ln 10}{\xi(\omega_2) - \xi(\omega_1)} \left(\frac{\xi(\omega_2)}{T_{60}(\omega_1)} - \frac{\xi(\omega_1)}{T_{60}(\omega_2)} \right).$$

$$\sigma_1 = \frac{6 \ln 10}{\xi(\omega_2) - \xi(\omega_1)} \left(\frac{1}{T_{60}(\omega_2)} - \frac{1}{T_{60}(\omega_1)} \right).$$

On définit la constante de décroissance $T_{60}(\omega)$ par

$$T60(\omega) = -\frac{6 \ln 10}{\sigma(\omega)} = \frac{6 \ln 10}{\sigma_0 + \sigma_1 \xi(\omega)} .$$

Pour deux fréquences données $\omega_1 < \omega_2$, on a donc

$$\sigma_0 + \sigma_1 \xi(\omega_1) = \frac{6 \ln 10}{T60(\omega_1)} ,$$

$$\sigma_0 + \sigma_1 \xi(\omega_2) = \frac{6 \ln 10}{T60(\omega_2)} .$$

En soustrayant ces deux équations, on trouve

$$\sigma_1 [\xi(\omega_2) - \xi(\omega_1)] = \frac{6 \ln 10}{T60(\omega_2)} - \frac{6 \ln 10}{T60(\omega_1)} .$$

D'où

$$\sigma_1 = \frac{6 \ln 10}{\xi(\omega_2) - \xi(\omega_1)} \left(\frac{1}{T60(\omega_2)} - \frac{1}{T60(\omega_1)} \right) .$$

Ensuite, en réinjectant dans l'une des équations, on a

$$\sigma_0 = \frac{6 \ln 10}{T60(\omega_1)} - \sigma_1 \xi(\omega_1) .$$

Ou encore

$$\sigma_0 = \frac{6 \ln 10}{\xi(\omega_2) - \xi(\omega_1)} \left(\frac{\xi(\omega_2)}{T60(\omega_1)} - \frac{\xi(\omega_1)}{T60(\omega_2)} \right) .$$

Conclusion

Ce projet nous a permis de réaliser une modélisation complète de la vibration d'une corde de guitare, en passant d'une approche idéale (équation des ondes, modes propres) à un modèle réaliste intégrant la raideur et les amortissements. Grâce à une discrétisation rigoureuse et à l'implémentation d'un schéma numérique implicite sous Scilab, nous avons pu simuler l'évolution du profil de la corde, enregistrer virtuellement

le son via des « micros » placés à des positions stratégiques, et analyser le signal obtenu en transformée de Fourier pour valider que la fréquence fondamentale correspondait bien à celle attendue.

Sur le plan qualitatif et quantitatif, l'analyse du spectre nous a montré comment les paramètres physiques – tels que la tension, la masse linéique, le paramètre d'inharmonicité B ainsi que la position du pincement – influent sur le timbre et la richesse harmonique du son. En particulier, la prise en compte de l'inharmonicité et des conditions d'amortissement nous a permis de mieux comprendre et reproduire la complexité acoustique d'un instrument de musique réel.

Ce projet a revêtu une importance majeure pour nous, car il nous a offert l'opportunité de lier théorie et pratique : nous avons pu appliquer des notions mathématiques et physiques complexes à un problème concret, tout en développant nos compétences en programmation et en simulation numérique. L'expérience acquise nous prépare à aborder des problématiques similaires dans d'autres domaines, notamment en acoustique et en ingénierie, et ouvre la voie à la conception d'instruments virtuels où la précision du modèle numérique est essentielle pour reproduire fidèlement le son d'un instrument réel.

Annexes

A. Simulation Complète et Restitution Sonore

A.1. Code Scilab complet de la simulation

Le programme Scilab qui réalise la simulation implicite du schéma pour la corde, en utilisant les conditions initiales (fonction triangle pour u_0 , vitesse nulle) et en enregistrant les vibrations aux positions de deux micros (avec interpolation linéaire). Le code calcule également σ_0 et σ_1 à partir du vecteur $loss$, construit les matrices A , B , C à l'aide de l'opérateur de différences D_{xx} (et de son carré D_{xxxx}) via `toeplitz/spdiags`, puis exécute la boucle temporelle en affichant l'animation et en stockant les signaux stéréo pour restituer le son final.

```
// Programmation du schéma implicite en Scilab
```

```
clear; clc;
```

```
// ----- Paramètres d'entrée -----
```

```
SR = 44100; // Taux d'échantillonnage (Hz)
```

```
B = 0.001; // Paramètre d'inharmonicité
```

```
f = 110; // Fréquence fondamentale (Hz)
```

```
TF = 4; // Durée de la simulation (s)
```

```
xo = 0.1; // Position où la corde est pincée (m)
```

```
co = 1; // Hauteur du pincement
```

```
rp = [0.3, 0.7]; // Positions des micros (m)
```

```
loss = [100, 10; 1000, 8]; // [f (Hz), T60 (s)] pour 2 couples
```

```
// ----- Paramètres physiques de la corde -----
```

```
// Pour  $L = 1$ , d'après l'énoncé :
```

```
//  $\gamma = 2 \cdot f$  et  $\kappa = (2 \cdot f / \pi) \cdot \sqrt{B}$ 
```

```
 $\gamma = 2 \cdot f$ ;
```

```
 $\kappa = (2 \cdot f / \pi) \cdot \sqrt{B}$ ;
```

```
// Temps et discrétisation spatiale
```

```

k = 1/SR;    // Pas temporel
L = 1;      // Longueur de la corde (m)
N = 101;    // Nombre de points spatiaux (donc N+1 points avec bords)
// Les points sont : x0 = 0, x1, ..., xN = L
x = linspace(0, L, N+1);
h = x(2)-x(1);

// On considère le vecteur d'inconnues intérieur : u^n = [u^n_1, ..., u^n_{N-1}]
n_int = N - 1;
x_in = x(2:$); // x(2) à x(N) (c'est-à-dire de h à L-h)

// ----- Calcul des paramètres d'amortissement  $\sigma_0$  et  $\sigma_1$  -----
// Conversion en fréquence angulaire :  $\omega = 2\pi f$ 
omega1 = 2 * %pi * loss(1,1);
omega2 = 2 * %pi * loss(2,1);
T60_1 = loss(1,2);
T60_2 = loss(2,2);

// Calcul de  $\xi(\omega)$  selon :  $\xi(\omega) = -\gamma^2 + \sqrt{\gamma^4 + 4\kappa^2\omega^2}/(2\kappa^2)$ 
xi1 = -gamma^2 + sqrt(gamma^4 + 4 * kappa^2 * omega1^2)/(2*kappa^2);
xi2 = -gamma^2 + sqrt(gamma^4 + 4 * kappa^2 * omega2^2)/(2*kappa^2);

// Formules finales pour  $\sigma_0$  et  $\sigma_1$ 
sigma1 = (6*log(10)/(xi2 - xi1)) * ( -1/T60_1 + 1/T60_2 );
sigma0 = (6*log(10)/(xi2 - xi1)) * ( xi2/T60_1 - xi1/T60_2 );

disp("sigma0 = " + string(sigma0));
disp("sigma1 = " + string(sigma1));

// ----- Construction des matrices de différences -----
// On construit l'opérateur Dxx (différences centrées seconde)

```

```

// Taille : (n_int) x (n_int)
e = ones(n_int,1);
Dxx = spdiags([e, -2*e, e], [-1, 0, 1], n_int, n_int) / h^2;
Dxxxx = Dxx * Dxx; // Opérateur de la dérivée quatrième

// ----- Construction des matrices du schéma implicite -----
// D'après l'énoncé, pour le vecteur u^n (dimension n_int)
// A = (1+σ0*k)I - σ1*k*Dxx
// B = -2I - γ^2*k^2*Dxx + κ^2*k^2*Dxxxx
// C = (1-σ0*k)I + σ1*k*Dxx
I_int = eye(n_int);
A = (1 + sigma0*k)*I_int - sigma1*k*Dxx;
B = -2*I_int - gamma^2*k^2*Dxx + kappa^2*k^2*Dxxxx;
C = (1 - sigma0*k)*I_int + sigma1*k*Dxx;

// On peut, si désiré, rendre ces matrices creuses pour accélérer la résolution
A = sparse(A);
B = sparse(B);
C = sparse(C);

// ----- Conditions initiales -----
// u0(x) : fonction triangle, avec u0(0)=u0(L)=0
u0 = zeros(n_int,1);
for i = 1:n_int
    xi = x_in(i);
    if xi <= xo then
        u0(i) = co * xi / xo;
    else
        u0(i) = co * (1 - xi) / (1 - xo);
    end
end
end

```

```

v0 = zeros(n_int,1); // vitesse initiale nulle

// Conditions initiales pour le schéma
// Pour schéma d'ordre 2, avec v0 = 0, on prend u^0 = u0 et u^1 = u0 (approximation)
u_prev = u0;
u_curr = u0;

// ----- Simulation temporelle -----
NF = TF * SR; // Nombre total d'échantillons temporels

// Pré-allocation du signal enregistré par les 2 micros (stereo)
out = zeros(2, NF);

// Pour l'animation, on affiche périodiquement le profil de la corde
figure("name", "Animation de la corde");

// Fonction d'interpolation linéaire pour extraire la valeur à une position p
function up = interp_lin(u, xvec, p)
    // u : vecteur des valeurs sur xvec, p : position recherchée
    [val, idx] = min(abs(xvec - p));
    // Si p ne coïncide pas exactement, on effectue une interpolation linéaire
    if xvec(idx) <= p then
        j = idx;
    else
        j = max(1, idx-1);
    end
    if j < length(xvec) then
        up = u(j) + (u(j+1) - u(j))*(p - xvec(j))/(xvec(j+1) - xvec(j));
    else
        up = u(j);
    end
end

```



```

endfunction

// Boucle temporelle
disp("Simulation en cours ...");
tic;
for n = 1:NF
    // Enregistrement aux positions des micros par interpolation
    for m = 1:2
        out(m, n) = interp_lin(u_curr, x_in, rp(m));
    end

    // Animation : on affiche toutes les 1000 itérations
    if mod(n, 1000) == 0 then
        clf();
        // Reconstituer le profil complet avec conditions aux limites
        u_full = [0; u_curr; 0];
        plot(x, u_full, "b-", "LineWidth", 2);
        title("t = " + string(n*k) + " s");
        xlabel("x (m)"); ylabel("u(x,t)");
        drawnow();
    end

    // Schéma implicite :  $A \cdot u^{(n+1)} = -B \cdot u^{(n)} - C \cdot u^{(n-1)}$ 
    u_next = A \ ( - B*u_curr - C*u_prev );
    // Mise à jour pour le prochain pas de temps
    u_prev = u_curr;
    u_curr = u_next;
end
t_sim = toc;
disp("Simulation terminée en " + string(t_sim) + " secondes.");

// ----- Lecture du son en stéréo -----

```

```
// Normalisation du signal  
out = out / max(abs(out));  
sound(out, SR);
```

A.2. Explication et utilisation du programme (animation du profil de la corde, enregistrement stéréo, restitution sonore)

But du Programme :

- Simuler de façon itérative le profil de la corde de guitare à partir des conditions initiales (fonction triangle pour le pincement et vitesse nulle).
- Afficher en temps réel l'évolution du profil (animation).
- Enregistrer le signal de déplacement aux positions des micros virtuels (avec interpolation si nécessaire).
- Restituer le son en stéréo à la fin de la simulation.

Instructions d'Utilisation :

1. Préparation du Script :

- Ouvrez le script Scilab dédié à la question 6 dans l'éditeur de Scilab.
- Vérifiez que les paramètres d'entrée (taux d'échantillonnage, durée de simulation, paramètres physiques, positions des micros, conditions initiales) sont correctement définis selon vos besoins.

2. Exécution de la Simulation :

- Lancez le script. Le programme effectue alors la discrétisation spatiale et temporelle de la corde.
- Une boucle temporelle itérative calcule le profil $u(x, t)$ à chaque pas de temps en résolvant le schéma implicite.
- Une fenêtre graphique s'ouvre (avec les commandes `scf`, `clf`, `drawnow`) pour afficher périodiquement le profil de la corde. Vous devriez voir une courbe oscillante qui représente la déformation de la corde en temps réel.
- Pendant l'exécution, le programme enregistre les valeurs de déplacement aux positions définies par le vecteur `rprprp` dans une variable (par exemple, `out`).

3. Restitution Sonore :

- À la fin de la simulation, le signal enregistré est normalisé et joué en stéréo via la commande `sound(out, SR)`.
- Vérifiez l'audio pour confirmer que le son correspond à la vibration simulée.

4. Conseils Pratiques :

- Si l'animation semble trop rapide, vous pouvez ajuster le délai (avec `sleep()`) pour mieux observer les changements.
- Assurez-vous que la fenêtre graphique est bien ouverte et que Scilab est en mode interactif pour voir l'animation en temps réel.

B. Analyse Spectrale et Étude Qualitative du Son

B.1. Code Scilab complet pour l'analyse FFT

Le programme Scilab qui réalise l'analyse du signal enregistré (par exemple, s_1 provenant du micro 1) pour vérifier que la fréquence fondamentale correspond bien à celle entrée, et qui trace le spectre afin d'observer qualitativement et quantitativement l'influence des paramètres d'inharmonicité B et de la position du pincement x_0 .

```
// Supposons que 'out' est le signal stéréo obtenu lors de la simulation (2 x NF)

// Ici, nous analysons le signal du premier micro

// Extraction du signal s1 (première ligne de out)
s1 = out(1, :); // s1 est un vecteur ligne de taille NF
Nfft = length(s1); // nombre de points pour la FFT

// Calcul de la FFT
S = fft(s1);

// Recentrage du spectre (fftshift) pour avoir un spectre symétrique autour de 0 Hz
S_shift = fftshift(S);

// Construction du vecteur de fréquence
fvec = linspace(-SR/2, SR/2, Nfft);

// Calcul du spectre d'amplitude
Amp = abs(S_shift);

// Recherche de l'indice du pic maximum
[maxAmp, idx] = max(Amp);
f_peak = fvec(idx);
disp("Fréquence du pic maximum : " + string(f_peak) + " Hz");

// Affichage du spectre
figure(3);
```

```

plot(fvec, Amp, "r-");
xlabel("Fréquence (Hz)");
ylabel("Amplitude");
title("Spectre du signal s1");

// Comparaison avec la fréquence fondamentale entrée
disp("Fréquence fondamentale entrée f = " + string(f) + " Hz");

// -----
// Analyse qualitative et quantitative :
// -----

// 1. L'indice de pic (f_peak) doit être proche de la fréquence fondamentale f (ici 110 Hz).
// Cela permet de vérifier que la note simulée correspond bien à celle attendue.

// 2. Paramètre B (inharmonicité) :
// - Une valeur plus élevée de B augmente l'inharmonicité, ce qui se traduit par un étalement
// des harmoniques dans le spectre (les pics harmoniques sont moins exactement
// multipliés de f).

// 3. Position x0 du pincement :
// - Lorsque la corde est pincée près du centre (x0 proche de 0.5), certains harmoniques
// (généralement les harmoniques pairs)
// tendent à être atténués par symétrie.
// - Un pincement proche d'une extrémité (chevalet) fournit un spectre avec une répartition
// d'amplitude différente
// et un timbre plus riche en harmoniques.

// Ainsi, quantitativement, en modifiant B et x0, on peut observer (dans le spectre FFT) des
// variations dans
// la position et l'amplitude relative des pics harmoniques.

// Qualitativement, ces modifications du spectre influent sur le timbre perçu :
// - Une inharmonicité plus élevée (B augmentée) conduit à un son plus « brillant » ou «
// métallique ».
// - La position du pincement (x0) influe sur la coloration du son, modifiant la dominance des
// harmoniques,
// ce qui modifie la perception du timbre

```

B.2. Explication et utilisation du programme d'analyse (transformée de Fourier, identification de la fréquence fondamentale, interprétation du spectre)

But du Programme :

- Analyser le signal sonore enregistré (généralement celui produit par la question 6, par exemple le signal du premier micro) en effectuant une transformée de Fourier.
- Vérifier que la fréquence fondamentale mesurée correspond à celle attendue (par exemple, 110 Hz pour un LA).
- Étudier l'effet des paramètres (comme le paramètre d'inharmonicité BBB et la position du pincement `x0x_0x0`) sur le spectre et donc sur le timbre du son.

Instructions d'Utilisation :

1. Préparation du Script :

- Ouvrez le script Scilab dédié à la question 7 dans l'éditeur.
- Assurez-vous que le signal d'enregistrement (par exemple, la variable `s1` ou `out` provenant du programme de la question 6) est disponible. Vous pouvez sauvegarder le signal du programme Q6 dans un fichier ou le transférer directement dans le script Q7.

2. Exécution de l'Analyse Spectrale :

- Le programme calcule la transformée de Fourier (FFT) du signal (par exemple, `s1 = out(1,:)` pour le premier micro).
- Il utilise `fftshift` pour centrer le spectre autour de zéro, puis construit le vecteur de fréquences correspondant.
- Le script identifie l'indice du pic maximum dans le spectre et détermine ainsi la fréquence fondamentale mesurée.
- Le spectre est ensuite tracé dans une figure, vous permettant de visualiser l'amplitude des harmoniques.

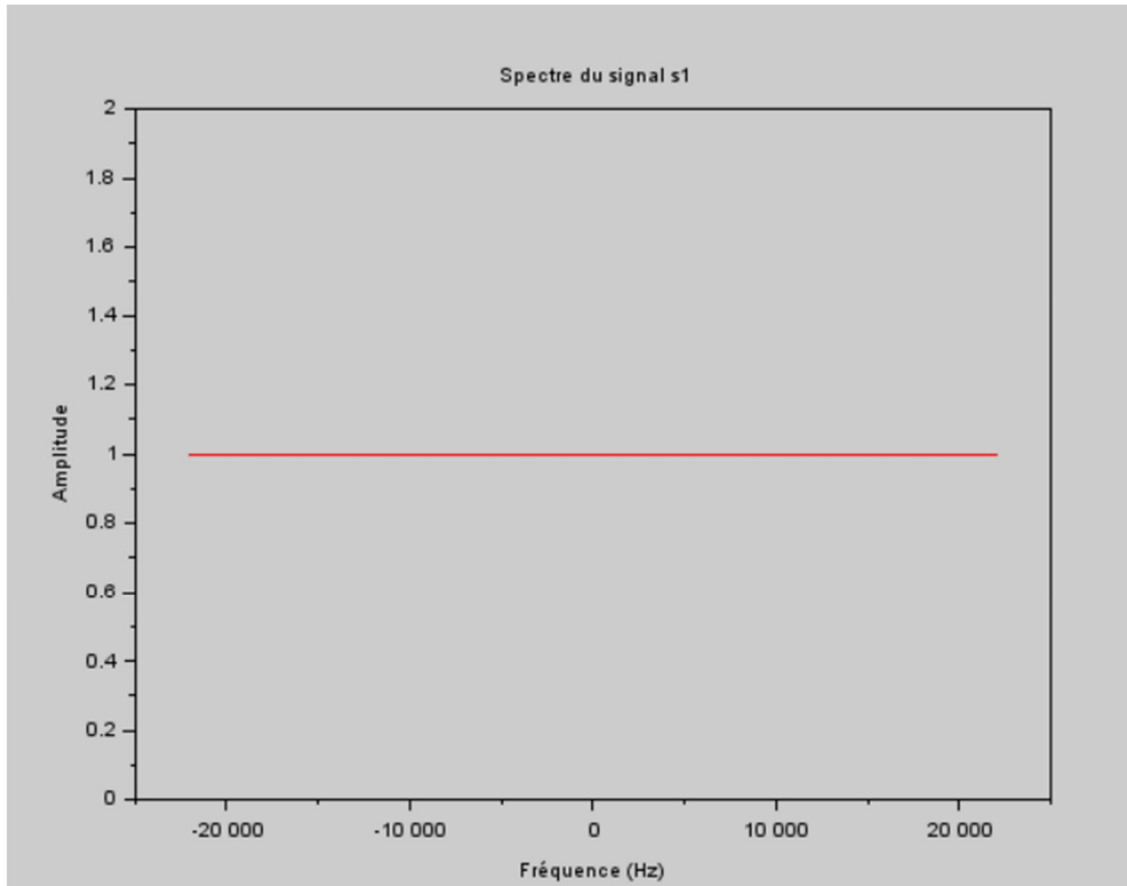
3. Interprétation des Résultats :

- Comparez la fréquence du pic maximum avec la fréquence fondamentale déclarée en entrée.
- Observez les effets des paramètres (en modifiant éventuellement les valeurs de BBB ou `x0x_0x0` et en relançant le script) sur la distribution des harmoniques dans le spectre.
- Ces observations permettent de comprendre qualitativement et quantitativement comment ces paramètres influent sur le timbre du son.

4. Conseils Pratiques :

- Assurez-vous que la fenêtre graphique du script Q7 s'affiche correctement et que l'échelle de fréquences est bien définie pour une bonne lecture du spectre.

- Vous pouvez ajouter des commandes xlabel, ylabel, et title pour personnaliser le graphe et faciliter l'interprétation.



Analyse du signal