

POLYTECH SORBONNE

Projet
**Vibrations, acoustique, musique :
Comparaison de méthodes pour l'analyse
numérique d'instruments à cordes**



BENELKATER MOHAMED
BURBANO PAULA
COMBARET LÉO
FIDA CYRILLE RUDIO
PREVOT ALEXIA
RAKOTOVAO JONATHAN

Encadrant :
Yann TEYTAUT
MAIN3
2021/2022

Table des matières

I Présentation du projet	1
1 Mise en contexte	1
2 Sujet	1
3 Objectif : comparer des méthodes de résolution numérique de l'équation d'onde	1
4 Idée de la démarche générale	1
II Étude de l'équation des ondes	2
1 Caractérisation du problème	2
1.1 Présentation de l'équation	2
2 Étude physique de la corde	2
III Discrétisation et modélisation	3
1 Principe	3
2 Méthodes d'Euler	3
2.1 Schéma Explicite	3
2.2 Schéma Implicite	5
3 Méthode de Runge-Kutta	6
IV Analyse des résultats	8
1 Étude de la stabilité	8
1.1 Méthodes d'Euler	8
1.1.1 Schéma explicite	8
1.1.2 Schéma implicite :	10
1.2 Méthode de Runge-Kutta	12
2 Comparaison des méthodes	13
2.1 Étude des erreurs	13
2.2 Ordre des erreurs	15
3 Conditions initiales	17
3.1 Variation des conditions initiales	17
3.2 Implémentation des variations des conditions initiales dans les modèles	18
V Conclusion	19
1 Conclusion du projet	19
2 Compétences acquises	19
Annexes	20

Première partie

Présentation du projet

1 Mise en contexte

Une onde est la propagation d'une perturbation qui se déplace à une certaine vitesse et possédant des propriétés physiques. Sans s'en rendre compte, nous avons à faire à ces ondes tous les jours lorsque nous regardons les ondulations d'une flaue d'eau produites par des gouttes d'eau, ou simplement dès lors qu'on ouvre les yeux et qu'on capte la lumière provenant des objets qui nous entourent. Ainsi, en savoir plus sur ces ondes et comprendre comment elles fonctionnent semble être un travail intéressant.

Lors de ce projet pluridisciplinaire, nous nous penchons sur l'étude d'ondes mécaniques se propageant sur le long d'une corde d'un instrument à cordes.

2 Sujet

Dans le cadre de ce projet, nous cherchons à appliquer des méthodes d'analyse numérique à un instrument à corde. En effet, nous pouvons nous demander comment modéliser numériquement une onde dans une corde de guitare.

3 Objectif : comparer des méthodes de résolution numérique de l'équation d'onde

L'objectif premier est de se familiariser avec le domaine physique des ondes et de chercher comment modéliser l'onde se propageant le long d'instruments à cordes. Nous serons confrontés à une équation dite équation d'onde ou équation de D'Alembert. Il sera alors nécessaire de l'étudier, la comprendre et utiliser des méthodes d'analyse numérique afin d'approcher une solution à cette équation. Par ailleurs, il sera intéressant de comparer différentes méthodes d'analyse numérique en prenant en compte des critères de stabilité, rapidité, efficacité, etc.

4 Idée de la démarche générale

Pour cela, nous allons étudier et comprendre différentes méthodes d'analyse numérique basée sur la discrétisation puis les appliquer mathématiquement pour notre équation.

Les différentes méthodes sont les suivantes :

- la méthode d'Euler explicite;
- la méthode d'Euler implicite;
- la méthode de Runge-Kutta.

Ensuite on les implémentera en code python et visualisera les résultats. Nous pourrons alors analyser la stabilité des modèles en faisant varier les paramètres (conditions initiales, pas de discrétisation). Enfin, nous nous confronterons les méthodes de discrétisation en comparant les résultats, les erreurs et nous ferons varier les conditions initiales.

Deuxième partie

Étude de l'équation des ondes

1 Caractérisation du problème

1.1 Présentation de l'équation

L'équation de D'Alembert (appelée aussi équation d'onde ou équation des ondes) est l'équation générale qui décrit la propagation d'une onde, qui peut être représentée par une grandeur scalaire ou vectorielle. Elle s'écrit :

$$\frac{\partial^2 u}{\partial t^2} = c^2 \frac{\partial^2 u}{\partial x^2}$$

avec c la célérité d'onde

Pour la résoudre, nous avons besoin de la condition de Neumann :

$$\frac{\partial u}{\partial t}(0, x) = f(x)$$

et de la condition de Dirichlet : $u(0, x) = g(x)$

où f et g sont des fonctions quelconques d'une seule variable.

Cette équation peut être résolue mathématiquement (solution dite exacte) pour des modèles simples tels que celui d'une corde que l'on étudiera. La solution exacte de l'équation d'onde est de la forme : $u(t, x) = f(x - ct) + g(x + ct)$ avec f et g des fonctions quelconques d'une seule variable.

Cependant, dans ce projet, on s'intéressera à appliquer des méthodes d'analyse numérique pour résoudre l'équation d'onde (solutions approchées).

Des ondes stationnaires sont créées sur la corde par le musicien. Comme dans le problème, la corde est fixée, il faut que la fonction $u(x, t)$ s'annule sur ses bornes.

On obtient alors les conditions initiales suivantes :

$$\begin{cases} u(0, t) = 0 \\ u(L, t) = 0 \end{cases}$$

2 Étude physique de la corde

La vitesse de propagation de l'onde sur la corde dépend de la tension de celle-ci :

$$c = \sqrt{\frac{\tau}{\rho}}$$

avec

τ qui représente la tension,

ρ la masse par unité de longueur, ou densité linéique.

Plus une corde est tendue, plus la vitesse de propagation d'une onde y sera élevée.

On prendra par la suite comme valeurs 0,5 m pour la longueur l de cette corde et $5,8 \cdot 10^{-3}$ kg.m⁻¹ pour la densité linéique ρ .

On pourra alors faire varier la tension de la corde et en observer les conséquences.

Troisième partie

Discrétisation et modélisation

1 Principe

On considère une équation différentielle ordinaire (EDO) $u'(t) = f(t, u(t))$. Lorsqu'on ne connaît pas de solution exacte à cette EDO, on essaye d'en avoir une bonne approximation par des méthodes numériques.

Cette équation décrit comment varie une fonction, en un point donné (un instant ou un point de l'espace), connaissant la valeur de cette fonction mathématique, le problème à résoudre s'écrit :

$$\begin{cases} u'(t) = f(t, u(t)) \\ u(t = t_0) = u_0 \end{cases}$$

Nous allons par exemple observer u sur un intervalle de temps régulier de pas h . Nous allons essayer d'obtenir une suite d'approximations u_n de $u(t_n)$.

Considérons le problème mono-dimensionnel de la vibration d'une corde de guitare. Le schéma de la vibration de guitare vérifie l'équation de D'Alembert :

$$\frac{\partial^2 u}{\partial t^2} = c^2 \frac{\partial^2 u}{\partial x^2}$$

où c est la vitesse de l'onde.

Ici, il ne s'agit pas d'une EDO mais d'une EDP (Equation aux Dérivées Partielles). Nous allons donc utiliser la méthode des différences finies.

La première étape consiste à discrétiser l'espace et le temps en des nombres finis d'intervalles de dimension connue appelé pas de discrétisation, respectivement Δx et Δt . C'est le maillage.

On remplace ensuite les dérivées apparaissant dans l'équation par des quotients aux différences obtenus à partir d'un développement de Taylor à un ordre fixé selon la précision recherchée. Il s'agit de la méthode des différences finies qui permet d'approcher les valeurs des dérivées.

2 Méthodes d'Euler

L'intervalle $[0, L]$ est discrétisé en $Nx + 1$ noeuds de coordonnées x_i (i variant de 0 à N) régulièrement espacés. Notons Δx le pas d'espace. L'intervalle de temps $[0, T]$ est discrétisé quant à lui en Nt intervalles de pas constant Δt . Notons u_i^n l'amplitude de la corde au noeud $x_i = i\Delta x$ et à l'instant $t_n = n\Delta t$.

2.1 Schéma Explicite

La méthode d'Euler explicite consiste à considérer que, d'un point t_i^n au point t_{i+1}^{n+1} , la fonction évolue linéairement, avec une trajectoire qui est celle qu'on peut calculer au point t_i .

On peut alors approximer la fonction $(u(t_n, x_i))$ en calculant u_{i+1}^{n+1} connaissant u_i^n .

Le problème se résout donc de la façon suivante :

- on connaît la fonction f , un point t_n où on connaît u_i^n
- on peut donc calculer $u'(t) = f(t, u(t))$
- on estime alors la valeur de u au point $t_{n+1} = t_n + \Delta t : u_{n+1} \approx u_n + u'_n \Delta t$
- on peut alors itérer (résoudre pas à pas) pour passer au point suivant. Le problème est initialisé en partant de t_0 où on connaît u_0 (condition à la limite).

On obtient à partir de la méthode des différences finies les schémas centré d'ordre 2 pour les dérivées spatiale et temporelle :

$$\left(\frac{\partial^2 u}{\partial t^2}\right)_i^n = \frac{u_i^{n+1} - 2u_i^n + u_i^{n-1}}{\Delta t^2}$$

$$\left(\frac{\partial^2 u}{\partial x^2}\right)_i^n = \frac{u_{i+1}^n - 2u_i^n + u_{i-1}^n}{\Delta x^2}$$

$$\frac{u_i^{n+1} - 2u_i^n + u_i^{n-1}}{\Delta t^2} = c^2 \frac{u_{i+1}^n - 2u_i^n + u_{i-1}^n}{\Delta x^2}$$

On pose :

$$\lambda = c \frac{\Delta t}{\Delta x}$$

ce qui donne

$$u_i^{n+1} - 2u_i^n + u_i^{n-1} = \lambda^2(u_{i+1}^n - 2u_i^n + u_{i-1}^n)$$

$$\Leftrightarrow \boxed{u_i^{n+1} = \lambda^2 u_{i+1}^n + (2 - 2\lambda^2)u_i^n - u_{i-1}^{n-1} + \lambda^2 u_{i-1}^n}$$

Posons $U = \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_{N-2} \\ u_{N-1} \end{pmatrix}$, $A = \begin{pmatrix} (2 - 2\lambda^2) & \lambda^2 & 0 & \dots & \dots & 0 \\ \lambda^2 & (2 - 2\lambda^2) & \lambda^2 & 0 & \dots & \vdots \\ 0 & \ddots & \ddots & \ddots & \dots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & 0 \\ \vdots & \vdots & 0 & \lambda^2 & (2 - 2\lambda^2) & \lambda^2 \\ 0 & \dots & \dots & 0 & \lambda^2 & (2 - 2\lambda^2) \end{pmatrix}$ et $C = \begin{pmatrix} u_{l0} \\ 0 \\ \vdots \\ 0 \\ u_{IN} \end{pmatrix}$

alors

$$U_i^{n+1} = A.U_i^n - U_i^{n-1} + C_i$$

On va donc résoudre cette équation en prenant des conditions initiales cohérentes. La corde de guitare est fixée à ses extrémités et admet un momentum initial lorsque $t = 0$ (que l'on définira lors des expérimentations).

Conditions initiales :

$$\begin{cases} u_0^n = 0 & \forall n \\ u_L^n = 0 & \forall n \end{cases}$$

Donc $C = 0$

Soit

$$\boxed{U_i^{n+1} = A.U_i^n - U_i^{n-1}}$$

On a donc la formule de récurrence cherchée pour approximer la suite u_i^n .

De plus, nous avons besoin de déterminer $u_i^1 \forall i$ pour utiliser notre schéma d'Euler. Ainsi, l'on utilise les conditions initiales du problème pour obtenir ces valeurs :

$$\begin{cases} \frac{\partial u}{\partial t}(0, x) = f(x) \\ u(0, x) = g(x) \end{cases}$$

On a donc par la méthode des différences finies :

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = \frac{\partial u}{\partial t}$$

$$\begin{aligned}
&\Leftrightarrow \frac{u_i^1 - u_i^0}{\Delta t} = \frac{\partial u(x, 0)}{\partial t} \\
&\Leftrightarrow \frac{\partial u(x, 0)}{\partial t} = \frac{u_i^1 - u_i^0}{\Delta t} \\
&\Leftrightarrow u_i^1 = \Delta t \frac{\partial u(x, 0)}{\partial t} + u_i^0
\end{aligned}$$

D'où :

$$\left\{
\begin{array}{ll}
u_i^0 = u(x, 0) & \forall n \\
u_i^1 = \Delta t \frac{\partial u(x, 0)}{\partial t} + u(x, 0) & \forall n
\end{array}
\right.$$

La description de l'algorithme utilisé est à retrouver en annexe 3.

Le code python est accessible via le lien Github en annexe 4.

2.2 Schéma Implicite

On reprend ici les mêmes notations. La méthode d'Euler implicite consiste à chercher la valeur approchée à l'instant t_{n+1} avec la relation suivante : $u_i^{n+1} \approx u_i^n + u_i'^{n+1} \Delta t$

On va donc utiliser une approche pour discréteriser l'équation au noeud x_i et à l'itération $n+1$:

$$(\frac{\partial u}{\partial t})_i^{n+1} = c^2 (\frac{\partial^2 u}{\partial t^2})_i^{n+1}$$

Nous utilisons un schéma arrière d'ordre 2 pour évaluer la dérivée seconde temporelle :

$$(\frac{\partial^2 u}{\partial t^2})_i^{n+1} = \frac{u_i^{n+1} - 2u_i^n + u_i^{n-1}}{\Delta t^2}$$

Ainsi qu'un schéma centré d'ordre 2 pour la dérivée seconde en espace :

$$(\frac{\partial^2 u}{\partial x^2})_i^{n+1} = \frac{u_{i+1}^{n+1} - 2u_i^{n+1} + u_{i-1}^{n+1}}{\Delta x^2}$$

On pose :

$$\lambda = c \frac{\Delta t}{\Delta x}$$

Alors d'après l'équation de discréétisation :

$$\begin{aligned}
&(\frac{\partial u}{\partial t})_i^{n+1} = c^2 (\frac{\partial^2 u}{\partial t^2})_i^{n+1} \\
&\Leftrightarrow \frac{u_i^{n+1} - 2u_i^n + u_i^{n-1}}{\Delta t^2} = c^2 \frac{u_{i+1}^{n+1} - 2u_i^{n+1} + u_{i-1}^{n+1}}{\Delta x^2} \\
&\Leftrightarrow u_i^{n+1} - 2u_i^n + u_i^{n-1} = \lambda^2 (u_{i+1}^{n+1} - 2u_i^{n+1} + u_{i-1}^{n+1}) \\
&\Leftrightarrow u_i^{n+1} - 2u_i^n + u_i^{n-1} = \lambda^2 (u_{i+1}^{n+1} - 2u_i^{n+1} + u_{i-1}^{n+1}) \\
&\Leftrightarrow -2u_i^n = \lambda^2 (u_{i+1}^{n+1} - 2u_i^{n+1} + u_{i-1}^{n+1}) - (u_i^{n+1} + u_i^{n-1}) \\
&\Leftrightarrow u_i^n = \frac{-\lambda^2}{2} (u_{i+1}^{n+1} - 2u_i^{n+1} + u_{i-1}^{n+1}) + \frac{1}{2} (u_i^{n+1} + u_i^{n-1}) \\
&\Leftrightarrow \boxed{u_i^n = \frac{-\lambda^2}{2} u_{i+1}^{n+1} + (\lambda^2 + \frac{1}{2}) u_i^{n+1} - \frac{\lambda^2}{2} u_{i-1}^{n+1} + \frac{1}{2} u_i^{n-1}}
\end{aligned}$$

Écriture de l'équation sous forme matricielle :

$$\text{Posons } U = \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_{N-2} \\ u_{N-1} \end{pmatrix}, \quad A = \begin{pmatrix} (\lambda^2 + \frac{1}{2}) & -\frac{\lambda^2}{2} & 0 & \cdots & \cdots & 0 \\ -\frac{\lambda^2}{2} & (\lambda^2 + \frac{1}{2}) & -\frac{\lambda^2}{2} & 0 & \cdots & \vdots \\ 0 & \ddots & \ddots & \ddots & \cdots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & 0 \\ \vdots & \vdots & 0 & -\frac{\lambda^2}{2} & (\lambda^2 + \frac{1}{2}) & -\frac{\lambda^2}{2} \\ 0 & \cdots & \cdots & 0 & -\frac{\lambda^2}{2} & (\lambda^2 + \frac{1}{2}) \end{pmatrix} \text{ et } C = \begin{pmatrix} u_g \\ 0 \\ \vdots \\ 0 \\ u_d \end{pmatrix}$$

alors

$$U_i^n = A \cdot U_i^{n+1} - \frac{\lambda^2}{2} \cdot C_i + \frac{1}{2} \cdot U_i^{n-1}$$

On va donc résoudre cette équation en prenant des conditions initiales cohérentes.

La corde de guitare est fixée à ses extrémités donc les points lorsque $x = 0$ et $x = L$ gardent leur position initiale.

Conditions initiales sur x :

$$\begin{cases} u(0, t) = u_g = 0 & \forall n \\ u(L, t) = u_d = 0 & \forall n \end{cases}$$

Donc $C = 0$

Soit

$$A \cdot U_i^{n+1} = U_i^n - \frac{1}{2} \cdot U_i^{n-1} \text{ et } U_i^{n+1} = A^{-1} \left(U_i^n - \frac{1}{2} \cdot U_i^{n-1} \right)$$

D'autre part, la corde admet un momentum initial $\forall x$ à l'instant $t = 0$. La position de la corde à $t = 0$ sera définie lors de l'expérimentation car il existe plusieurs modèles de momentum initial qui peuvent faire varier les résultats.

De la même manière que pour la méthode explicite, on obtient les conditions initiales :

$$\begin{cases} u_i^0 = u(x, 0) & \forall n \\ u_i^1 = \Delta t \frac{\partial u(x, 0)}{\partial t} + u(x, 0) & \forall n \end{cases}$$

La description de l'algorithme utilisé est à retrouver en annexe 3.

Le code python est accessible via le lien Github en annexe 4.

3 Méthode de Runge-Kutta

Les méthodes de Runge-Kutta sont des schémas numériques à un pas qui permettent de résoudre des équations différentielles ordinaires. Elles sont appréciées pour leur précision grâce à des ordres plus élevés : 2 ou 4.

Dans le cas de l'équation d'onde, l'utilisation d'une méthode de Runge-Kutta n'est pas évidente étant donné qu'il s'agit d'une équation aux dérivées partielles d'ordre 2. Cependant, en utilisant la méthode des différences finies et quelques astuces, il est possible de la résoudre avec les méthodes de Runge-Kutta. Nous introduisons la même discrétisation de temps et d'espace que pour les méthodes d'Euler.

1. Discrétisation de l'espace avec un pas Δx et approximation de $\frac{\partial^2 u_i^n}{\partial x^2}$ par la méthode des différences finies.

$$\frac{\partial^2 u_i^n}{\partial x^2} = \frac{u_{i-1}^n - 2u_i^n + u_{i+1}^n}{\Delta x^2}$$

et donc l'équation d'onde devient :

$$\frac{\partial^2 u_i^n}{\partial t^2} = \left(\frac{c}{\Delta x}\right)^2 (u_{i-1}^n - 2u_i^n + u_{i+1}^n)$$

2. Réécrire l'équation sous la forme d'un système d'équations différentielles ordinaires.

Soit z une fonction de \mathbb{R} telle que

$$\begin{cases} \frac{\partial u}{\partial t} = z(t) \\ \frac{dz}{dt} = c^2 \frac{\partial^2 u_i^n}{\partial x^2} \end{cases}$$

Soient f et g deux fonctions de \mathbb{R} telles que

$$\begin{cases} g(t, u, z) = z(t) \\ f(t, u, z) = c^2 \frac{\partial^2 u_i^n}{\partial x^2} = (\frac{c}{\Delta x})^2 (u_{i-1}^n - 2u_i^n + u_{i+1}^n) \end{cases}$$

On obtient alors le système d'équations différentielles ordinaires :

$$\begin{cases} \frac{\partial u_i^n}{\partial t} = g(t, u, z) \\ \frac{dz}{dt} = f(t, u, z) \end{cases}$$

On peut résoudre ces deux équations par la méthode de Runge-Kutta d'ordre 4.

3. Résolution du système par Runge-Kutta d'ordre 4 avec un pas Δt

La méthode de Runge-Kutta d'ordre 4 nous permet d'obtenir les schémas itératifs suivants :

$$\begin{cases} u_i^{n+1} = u_i^n + \frac{1}{6}((k_0)_i + (k_1)_i + (k_2)_i + (k_3)_i) \\ z_i^{n+1} = z_i^n + \frac{1}{6}((l_0)_i + (l_1)_i + (l_2)_i + (l_3)_i) \end{cases}$$

avec :

$$\begin{cases} (k_0)_i = \Delta t \times g(tn, u_i^n, z_i^n) \\ (k_1)_i = \Delta t \times g(tn + \frac{\Delta t}{2}, u_i^n + \frac{(k_0)_i}{2}, z_i^n + \frac{(l_0)_i}{2}) \\ (k_2)_i = \Delta t \times g(tn + \frac{\Delta t}{2}, u_i^n + \frac{(k_1)_i}{2}, z_i^n + \frac{(l_1)_i}{2}) \\ (k_3)_i = \Delta t \times g(tn + \Delta t, u_i^n + (k_2)_i, z_i^n + (l_2)_i) \end{cases}$$

et

$$\begin{cases} (l_0)_i = \Delta t \times f(tn, u_i^n, z_i^n) \\ (l_1)_i = \Delta t \times f(tn + \frac{\Delta t}{2}, u_i^n + \frac{(k_0)_i}{2}, z_i^n + \frac{(l_0)_i}{2}) \\ (l_2)_i = \Delta t \times f(tn + \frac{\Delta t}{2}, u_i^n + \frac{(k_1)_i}{2}, z_i^n + \frac{(l_1)_i}{2}) \\ (l_3)_i = \Delta t \times f(tn + \Delta t, u_i^n + (k_2)_i, z_i^n + (l_2)_i) \end{cases}$$

4. Conditions initiales

Pour les méthodes de Runge-Kutta, les conditions initiales sont également très importantes. Dans le cas de l'équation d'onde, il faut que les valeurs de u_i^n et de $\frac{\partial u_i^n}{\partial t}$ en $t = 0$ soient connues afin de pouvoir calculer le reste des valeurs.

La description de l'algorithme utilisé est à retrouver en annexe 3.

Le code python est accessible via le lien Github en annexe 4.

Quatrième partie

Analyse des résultats

1 Étude de la stabilité

Dans cette partie, nous allons évaluer la stabilité de chaque modèle numérique qui nous permettra d'étudier les conditions de convergence de ces différents modèles expérimentaux à partir des paramètres de discréétisation.

En effet, la stabilité dépend du Courant-Friedrichs-Lowy (CFL), condition suffisante de la stabilité liant les pas d'espace Δx et de temps Δt . Le CFL est une condition de convergence pour résoudre certaines équations aux dérivées partielles.

Dans notre problème le CFL est défini de la façon suivante :

$$\lambda = c \frac{\Delta t}{\Delta x}$$

c : vitesse de l'onde x
 Δx : pas d'espace
 Δt : pas de temps

Les tests ont été effectués avec des valeurs de référence :

- $c=340$ m/s (vitesse arbitraire mais qui correspond à la vitesse d'une onde dans l'air)
- $L=0.5$ m (longueur moyenne d'une corde de guitare)
- Durée=0.001 s

Et avec les conditions initiales suivantes :

$$\begin{cases} u_i^0 = \sin(\frac{n\pi}{L} \times i \times \Delta x) \\ \frac{\partial u_i^0}{\partial t} = 0 \end{cases}$$

1.1 Méthodes d'Euler

Pour étudier la stabilité de cette méthode, nous allons faire varier d'abord Δt puis Δx pour obtenir les graphiques les plus précis pour chaque schéma explicite et implicite, l'objectif étant de trouver un compromis entre l'efficacité et la précision.

1.1.1 Schéma explicite

▷ Variation de Δt (en secondes) à Δx (en mètres) fixé :

a) $\Delta x = 5 * 10^{-4} m$

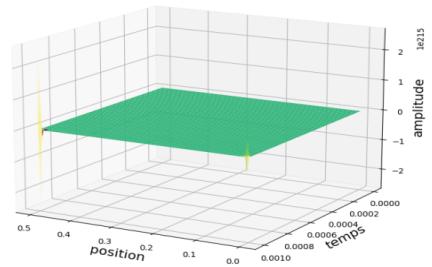
$\Delta t = 10^{-2} s$

$\lambda = 6,8 * 10^3$

⇒ On obtient aucune représentation.

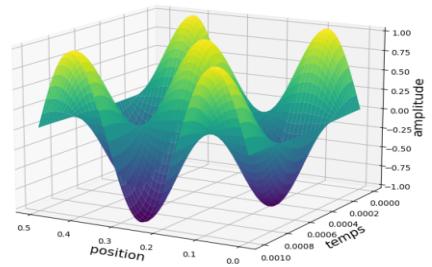
b) $\Delta x = 5 * 10^{-4} m$
 $\Delta t = 10^{-5} s$
 $\lambda = 6,8$

⇒ On obtient une surface plane.
Temps d'exécution = 0,11s



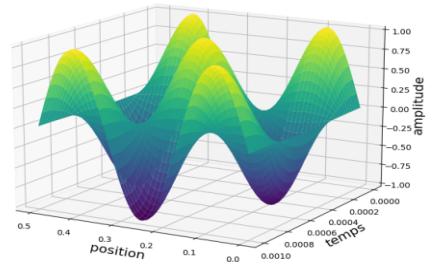
c) $\Delta x = 5 * 10^{-4} m$
 $\Delta t = 10^{-6} s$
 $\lambda = 6,8 * 10^{-1} s$

⇒ On obtient une courbe parfaitement sinusoïdale.
Temps d'exécution = 1,24s



d) $\Delta x = 5 * 10^{-4} m$
 $\Delta t = 10^{-8}$
 $\lambda = 6,8 * 10^{-3} s$

⇒ On obtient une courbe parfaitement sinusoïdale.
Temps d'exécution = 122s



D'après ces différents tests, on peut remarquer que le schéma est de plus en plus sinusoïdale lorsque le CFL se rapproche de 1.

▷ Variation de Δx à Δt fixé :

On fixe Δt à différentes valeurs :

- Pour $\Delta t = 10^{-3}$

a) $\Delta x = 3,4 * 10^{-2} m$
 $\Delta t = 10^{-4}$
 $\lambda = 1$

⇒ On obtient aucun affichage.

b) $\Delta x = 7 * 10^{-2} m$
 $\Delta t = 10^{-4}$
 $\lambda = 4,8 * 10^{-1} m$

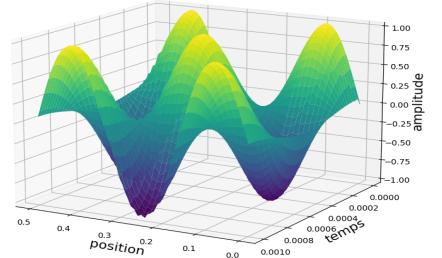
⇒ On obtient aucun affichage.

- Pour $\Delta t = 10^{-6}$

a) $\Delta x = 4 * 10^{-3} m$
 $\Delta t = 10^{-6}$
 $\lambda = 0,85$

⇒ On obtient la représentation d'une onde parfaitement sinusoïdale.

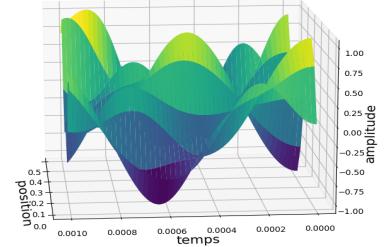
Temps d'exécution = 0,043s



b) $\Delta x = 4 * 10^{-2} m$
 $\Delta t = 10^{-6}$
 $\lambda = 0,0085$

⇒ On obtient une figure sinusoïdale.

Temps d'exécution = 0,016s



On constate à travers ces différents tests concernant la variation de l'espace à Δt fixé, que les représentations se modélisent de mieux en mieux lorsque notre Δt décroît.

1.1.2 Schéma implicite :

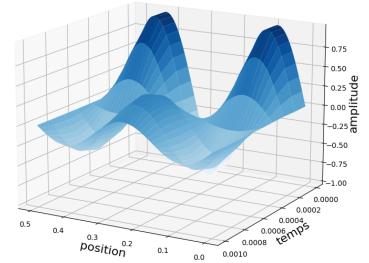
▷ Variation de Δt (en secondes) à Δx (en mètres) fixé :

a) $\Delta x = 5 * 10^{-4} m$
 $\Delta t = 10^{-2} s$
 $\lambda = 6,8 * 10^3$

⇒ Représentation non supportée donc aucun affichage car λ est trop grand.

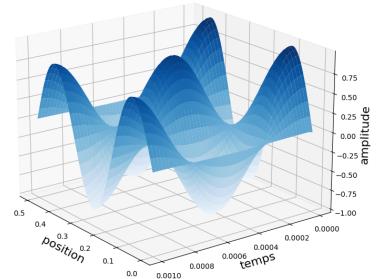
b) $\Delta x = 5 * 10^{-4} m$
 $\Delta t = 10^{-4} ms$
 $\lambda = 68$

⇒ En ayant un λ qui se rapproche de 1, on obtient une onde partiellement sinusoïdale.
Temps d'exécution = 0,207s



c) $\Delta x = 5 * 10^{-4} m$
 $\Delta t = 10^{-5} s$
 $\lambda = 6,8$

⇒ On obtient une surface parfaitement sinusoïdale lorsque le pas de temps est plus petit et donc par conséquent un λ plus petit se rapprochant de 1.
Temps d'exécution = 0,302s



d) $\Delta x = 5 * 10^{-4} m$
 $\Delta t = 10^{-8}$
 $\lambda = 6,8 * 10^{-3} s$

⇒ On obtient aucune représentation de l'onde car le pas de λ est trop petit.

Le coefficient CFL joue un rôle très important dans la stabilité du modèle, ce dernier liant les pas de temps et d'espace nous permet d'avoir des représentations plus ou moins sinusoïdales en fonction de la variation de Δt .

▷ Variation de Δx à Δt fixé :

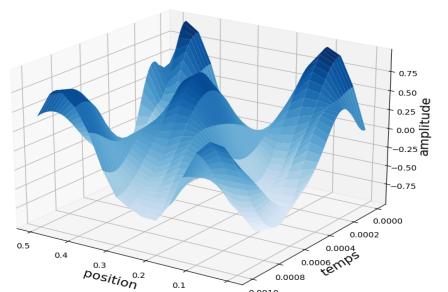
- Pour $\Delta t = 10^{-3}$

a) $\Delta x = 5 * 10^{-3} m$
 $\Delta t = 10^{-3}$
 $\lambda = 680$

⇒ On obtient aucun affichage.

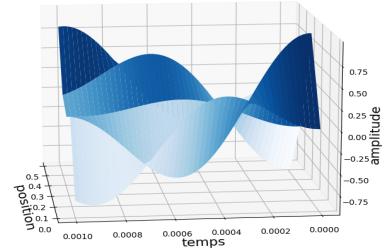
b) $\Delta x = 3 * 10^{-2} m$
 $\Delta t = 10^{-5}$
 $\lambda = 1,1 * 10^{-1}$

⇒ On commence à avoir une forme d'onde sinusoïdale.



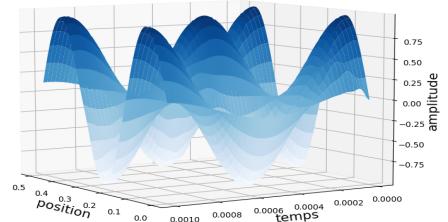
- Pour $\Delta t = 10^{-5}$

- c) $\Delta x = 9 * 10^{-2} m$
 $\Delta t = 10^{-5}$
 $\lambda = 0,03$
 \Rightarrow L'onde commence à apparaître de manière imprécise.

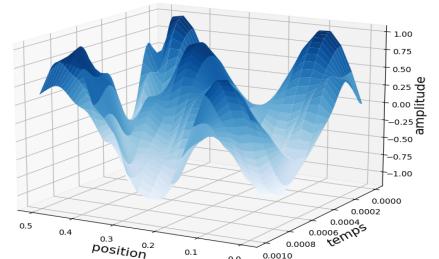


- Pour $\Delta t = 10^{-6}$

- d) $\Delta x = 7 * 10^{-3} m$
 $\Delta t = 10^{-6}$
 $\lambda = 0,048 s$
 \Rightarrow On obtient une représentation parfaitement sinusoïdale.



- e) $\Delta x = 2,4 * 10^{-2} m$
 $\Delta t = 10^{-6}$
 $\lambda = 1,4 * 10^{-2}$
 \Rightarrow On a une représentation sinusoïdale.



On obtient comme précédemment, une modélisation représentative de l'onde lorsque le pas de temps diminue mais également en ayant un pas d'espace d'ordre 10^{-2}

Conclusion : On peut remarquer d'après cette série de tests, que la stabilité est obtenu généralement avec un coefficient CFL proche de 1 par conséquent un pas d'espace Δx de l'ordre 10^{-2} au minimum et un pas de temps de l'ordre 10^{-5} au minimum.

1.2 Méthode de Runge-Kutta

Le principe d'étude de la stabilité du modèle reste le même que pour les schémas d'Euler. Il faut donc faire varier les pas Δx et Δt de manière à avoir les meilleurs graphiques. Mais, il apparaît assez évident que le modèle risque de diverger avec le terme $(\frac{c}{\Delta x})^2$.

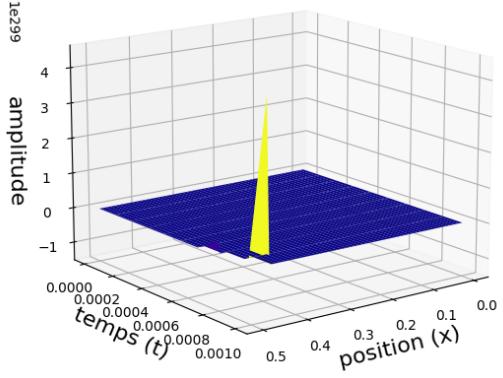


FIGURE 1 – $\Delta x = 5^{-3}m$

$$\Delta t = 10^{-4}s$$

$$\lambda = 6.8$$

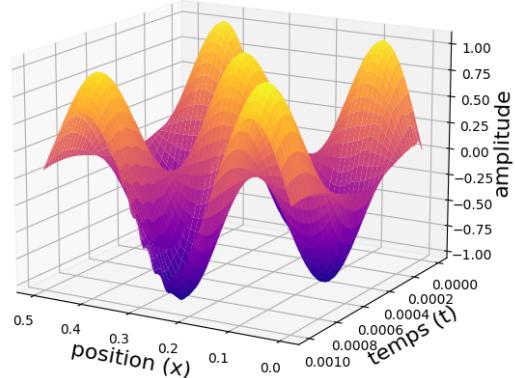


FIGURE 2 – $\Delta x = 5^{-3}m$

$$\Delta t = 10^{-5}s$$

$$\lambda = 0.68$$

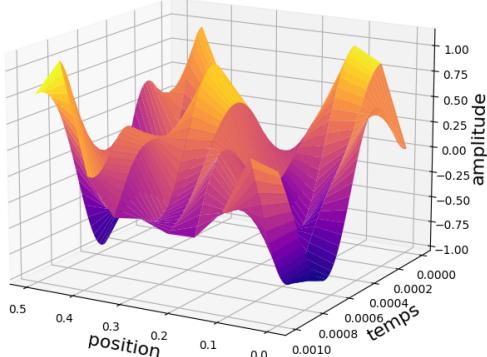


FIGURE 3 – $\Delta x = 5^{-2}m$

$$\Delta t = 10^{-6}s$$

$$\lambda = 0.0068$$

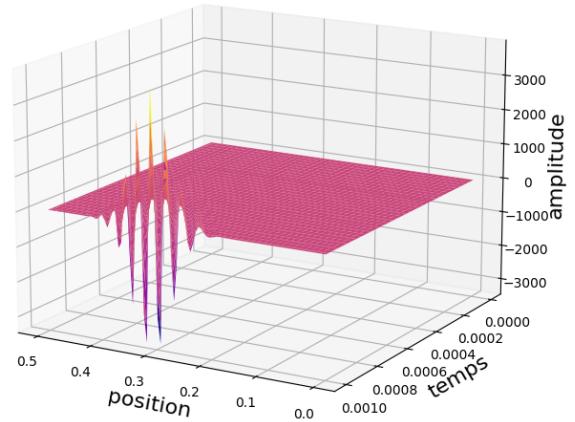


FIGURE 4 – $\Delta x = 5^{-3}m$

$$\Delta t = \frac{\Delta x}{c}s = 1.47 \cdot 10^{-5}$$

$$\lambda = 1$$

A partir des simulations effectuées (les autres étant en annexe 5), est possible d’observer expérimentalement les conditions de stabilité du modèle construit.

1. Par rapport au CFL : $\lambda = c \frac{\Delta t}{\Delta x}$

On peut remarquer que le modèle est "correct" uniquement lorsque $\lambda < 1$ (Figure 2).

Dans les autres cas, le résultat est loin de ce qui attendu. Cela est logique étant donné qu'il s'agit ici d'une méthode à deux pas, donc le CFL est un paramètre important.

2. Par rapport aux valeurs de Δt et Δx

On peut logiquement remarquer que la stabilité du modèle augmente lorsque le pas Δt diminue, il faut que Δt soit plus petit que Δx et que Δx soit assez petit mais trop le diminuer ferait diverger le modèle. Cela découlerait du facteur $(\frac{c}{\Delta x})^2$ qui augmente énormément lorsque Δx diminue.

2 Comparaison des méthodes

L’objectif principal du projet est de comparer les différentes méthodes de résolution numérique pour l’équation d’onde appliquée à une corde de guitare. Pour cela, nous allons comparer expérimentalement les erreurs obtenues pour chaque modèle pour des conditions de stabilité optimales et pour une solution exacte donnée.

2.1 Étude des erreurs

Pour effectuer les calculs d’erreurs, nous avons choisi la fonction suivante :

$$u_{exact}(x, t) = \cos\left(\frac{3\pi c}{L}t\right)\sin\left(\frac{3\pi}{L}x\right)$$

Et donc l'erreur numérique est simplement donnée par : $E = |u_{exact} - u_{num}|$

Ainsi, on obtient comme conditions initiales pour les trois modèles :

$$\begin{cases} u_i^0 = \sin\left(\frac{n\pi}{L} \times i \times \Delta x\right) \\ \frac{\partial u_i^0}{\partial t} = 0 \end{cases}$$

Et les paramètres des modèles sont les suivants :

- $c=340$ m/s (vitesse arbitraire mais qui correspond à la vitesse d'une onde dans l'air)
- $L=0.5$ m (longueur moyenne d'une corde de guitare)
- $Dure=0.001$ s

Et les paramètres de discréttisation pour les méthodes d'Euler sont :

- $\Delta x = 0.0005$
- $\Delta t = \frac{\Delta x}{c}$

Ceux de la méthode de Runge-Kutta sont :

- $\Delta x = 0.005$
- $\Delta t = 0.000001$

L'on pourra retrouver les graphiques des solutions pour ces 4 méthodes de calcul en annexe 2.

On peut déjà remarquer que pour les 3 modèles, les solutions numériques sont très proches de la solution exacte en terme de forme de surface mais aussi en terme d'amplitude.

Voici les résultats donnés par les calculs d'erreurs numériques en 3 dimensions.

Erreur en fonction de x et t pour Euler explicite

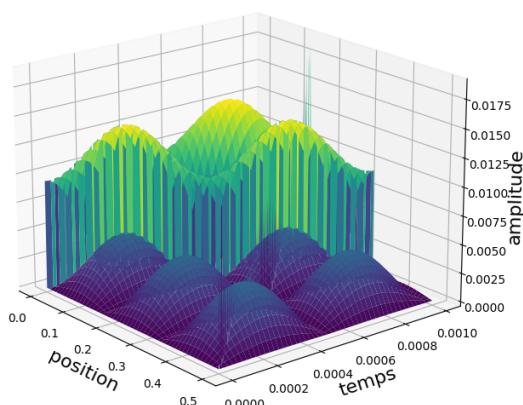


FIGURE 5 – Euler explicite

Erreur en fonction de x et t pour Euler implicite

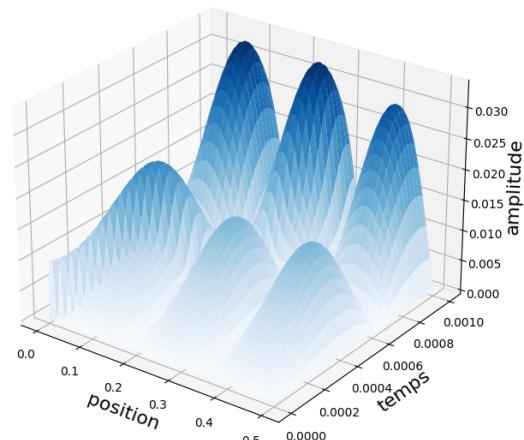


FIGURE 6 – Euler implicite

Erreur en fonction de x et t pour Runge-Kutta

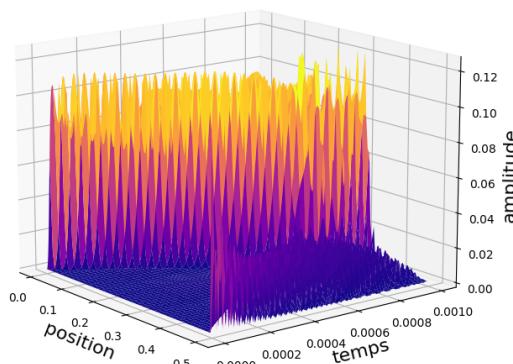


FIGURE 7 – Runge-Kutta

Les erreurs pour chaque modèle prennent des formes différentes. Pour les méthodes d'Euler, on peut observer des structures "sinusoïdales" quasiment périodiques. On peut cependant observer sur le schéma explicite une diminution générale de l'erreur à partir d'une certaine position. Pour le schéma de Runge-Kutta, on retrouve une forme similaire à celle de la méthode d'Euler explicite avec également la même diminution de l'erreur.

Mais il est plus intéressant d'évaluer les erreurs en 2 dimensions selon le temps. Ainsi, nous avons étudié les moyennes des erreurs pour chaque temps donné.

Les moyennes des erreurs pour chaque t donnés nous donnent alors les graphiques suivants :

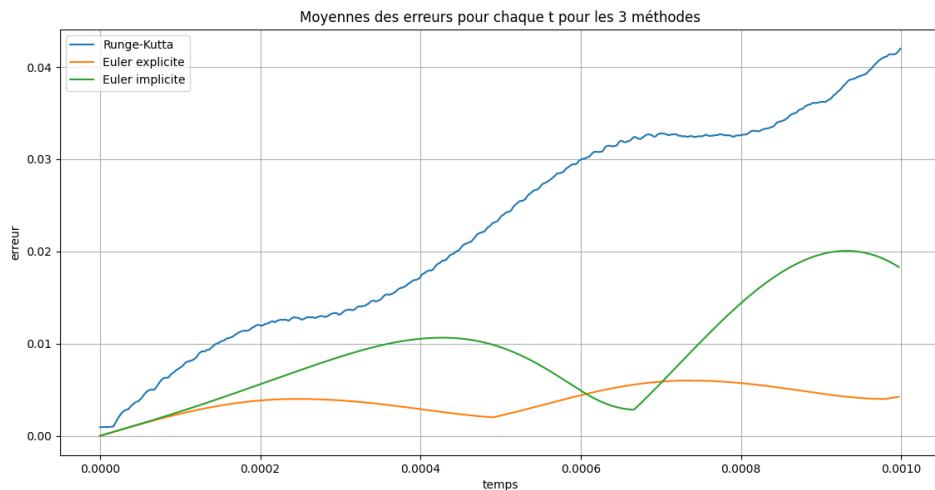


FIGURE 8 – Les erreurs pour les 3 méthodes

Les erreurs en fonction de t nous permettent de voir que le modèle le plus précis est le modèle d'Euler explicite. Globalement les erreurs ont bien des profils sinusoïdaux mais elles augmentent en fonction du temps, ce qui montre que les modèles présentent des facteurs de propagation d'erreur. Le modèle le moins précis est le modèle de Runge-Kutta. L'ordre des erreurs est de 10^{-2} pour les modèles mais le modèle de Runge-Kutta a une erreur qui croît beaucoup plus vite.

2.2 Ordre des erreurs

Pour finir la comparaison des modèles, nous pouvons déterminer et comparer l'ordre de leurs erreurs. Pour cela, nous pouvons étudier les expressions suivantes :

1. $\max|e_i^n|$ pour $0 < i < Nx$ et $0 < n < Nt$ en faisant varier le nombre de point de temps Nt (Figure 9)
2. $\max|e_i^n|$ pour $0 < i < Nx$ et $0 < n < Nt$ en faisant varier le nombre de point d'espace Nx (Figure 10)

Voici les résultats obtenus :

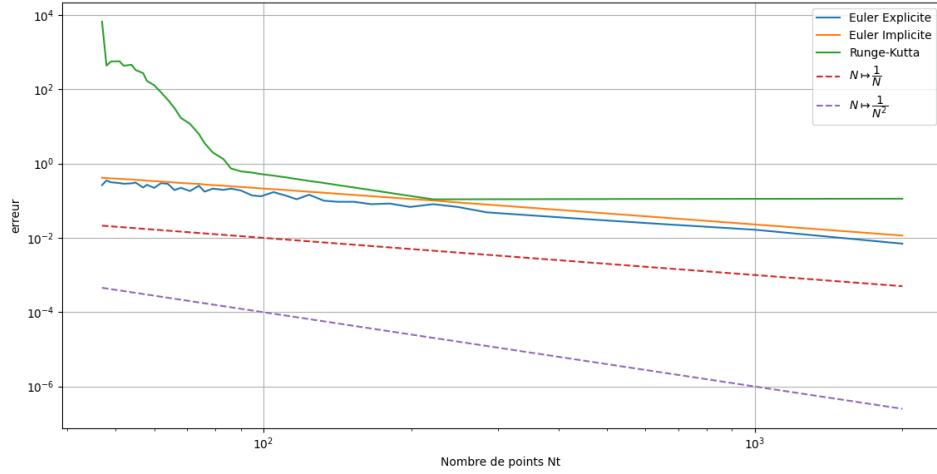


FIGURE 9 – Les erreurs pour des variation de Nt

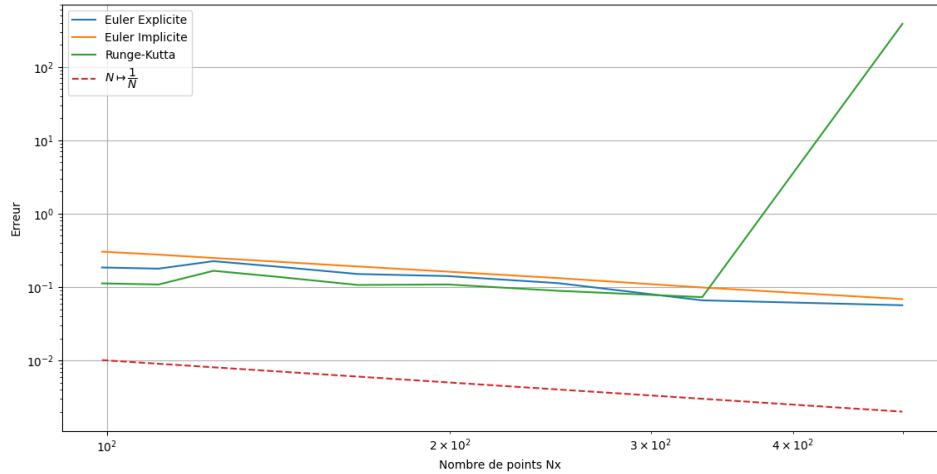


FIGURE 10 – Les erreurs pour des variation de Nx

A partir de ces résultats, on peut déduire les ordres des erreurs :

- Les méthodes d'Euler ont des erreurs selon x et t d'ordre $O(N)$
- La méthode de Runge-Kutta a des erreurs d'ordre $O(N^2)$ puis constante selon t et d'ordre $O(N)$ selon x . On peut déjà remarquer que la précision n'augmente plus en diminuant Δt à partir d'une certaine valeur. On peut également remarquer que selon x , l'erreur augmente brutalement au bout d'une certaine valeur, cela est dû au fait que la condition de stabilité du modèle n'est plus respectée. On peut remarquer grâce
- Globalement la méthode d'Euler explicite est celle qui donne la meilleure précision.

Au final, on peut déduire que, dans notre cas, la méthode d'Euler explicite est la plus intéressante puisqu'elle produit l'erreur la plus petite mais est également efficace en terme d'algorithme contrairement à la méthode de Runge-Kutta qui demande beaucoup plus de calculs pour avoir un résultat cohérent.

3 Conditions initiales

3.1 Variation des conditions initiales

L'analyse des méthodes utilisées pour la modélisation d'une onde de corde de guitare repose aussi sur le choix des conditions initiales comme vu précédemment. Un bon choix des conditions initiales est donc indispensable pour avoir une modélisation la plus proche de la réalité ou tout simplement celle qu'on souhaite.

Dans cette partie, l'objectif est de tester les différents schémas pour différents profils initiaux de l'onde, que l'on obtiendra en modifiant les conditions initiales.

Ainsi, on pourra visualiser les réactions des différents modèles par rapport aux variations de conditions initiales et constater que, tant que les conditions initiales respectent les conditions imposées par le problème ($u(t, x = 0) = u(t, x = L) = 0$), on obtient une solution numérique cohérente.

Dans un premier temps, on peut considérer une corde pincée en point de l'espace, qui est bien attachée sur les bords. On peut établir 2 modèles mathématiques différents pour la corde pincée du schéma suivant :

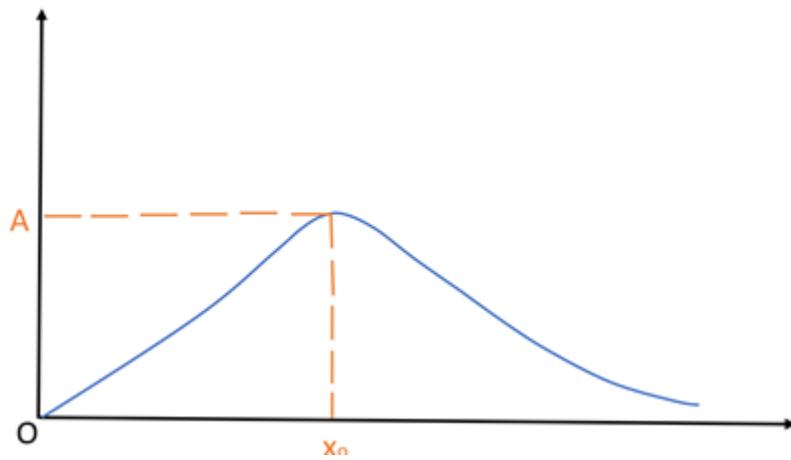


FIGURE 11 – Schéma d'une corde pincée

1. Modèle triangulaire

$$u(x, t = 0) = \begin{cases} A \frac{x}{x_0} & x \leq x_0 \\ A \frac{x_0}{x_0} & x \geq x_0 \end{cases} \quad (1)$$

2. Modèle complet (harpe)

$$u(x, t = 0) = A \frac{x}{x_0} \frac{L - x}{L} \quad (2)$$

Dans un deuxième temps, nous allons étudier un troisième modèle où la corde n'est pas accrochée aux bords, c'est-à-dire que, $u(t = 0, x = 0)$ on a une amplitude A différente de 0. Le résultat ne devrait alors pas être correct étant donné qu'il ne respecte plus les conditions du problème.

3. Modèle "non accroché aux bords"

$$\begin{cases} u(x, t = 0) = \cos\left(\frac{n\pi x}{L}\right) \\ \frac{\partial u}{\partial t}(x, t = 0) = 0 \end{cases} \quad (3)$$

3.2 Implémentation des variations des conditions initiales dans les modèles

Méthode d'Euler explicite

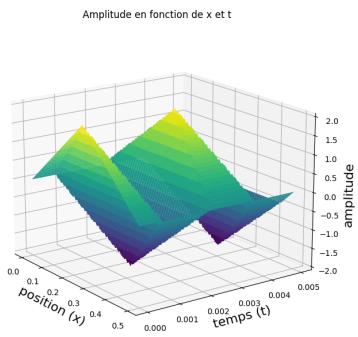


FIGURE 12 – Modèle (1)

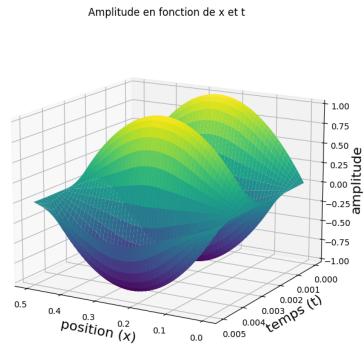


FIGURE 13 – Modèle (2)

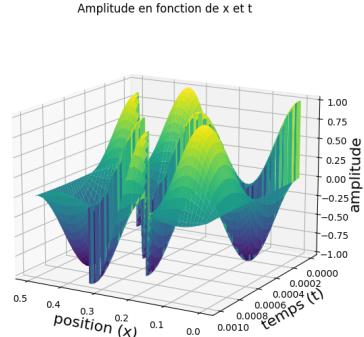


FIGURE 14 – Modèle (3)

Méthode d'Euler implicite

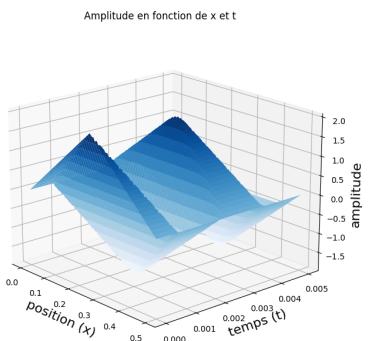


FIGURE 15 – Modèle (1)

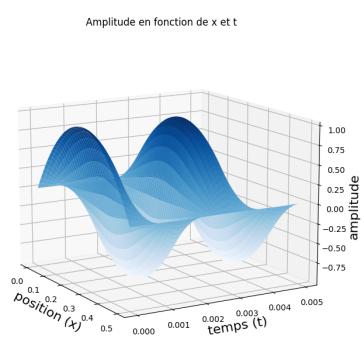


FIGURE 16 – Modèle (2)

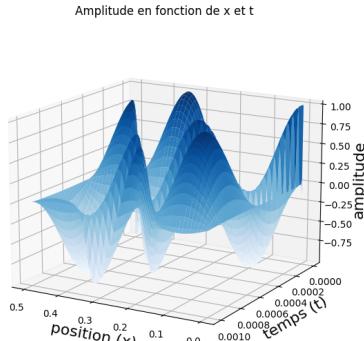


FIGURE 17 – Modèle (3)

Méthode de Runge-Kutta

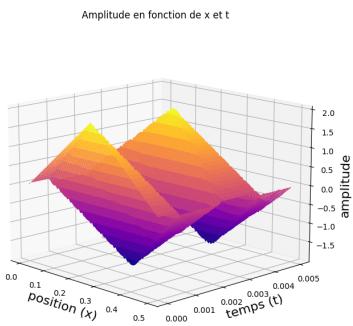


FIGURE 18 – Modèle (1)

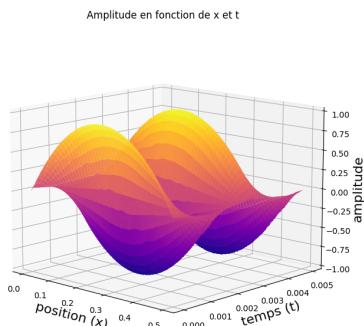


FIGURE 19 – Modèle (2)

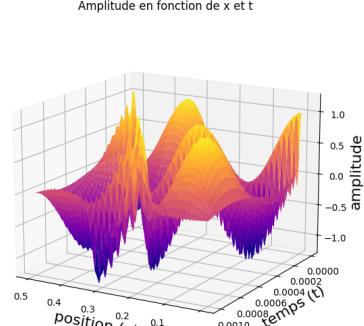


FIGURE 20 – Modèle (3)

Cinquième partie

Conclusion

1 Conclusion du projet

Au cours de ce projet, nous voulions réaliser des modèles propagation des ondes dans un instrument à corde afin de les comparer. Nous avons pu établir 3 modèles différents pour résoudre numériquement l'équation d'onde. Ces modèles mathématiques permettent d'obtenir des solutions numériques relativement proche de la solution exacte. Nous pouvons résumer les points importants de chaque méthode :

- Euler explicite : efficacité de calcul avec une formule de récurrence,utilisation de la vectorialisation (Numpy) et précision à 10^{-2}
- Euler implicite : efficacité de calcul avec résolution de système,utilisation de vectorialisation (Numpy) et précision à 10^{-2}
- Runge-Kutta : méthode non efficace à cause des coefficients de Runge-Kutta et l'absence de vectorialisation (Numpy), précision à 10^{-1} . La méthode n'est pas intéressante pour de la résolution d'équation aux dérivées partielles.

Donc,l'étude des modèles nous permet de conclure que la méthode la plus efficace pour résoudre l'équation d'onde est la méthode d'Euler explicite pour sa minimisation des erreurs et du temps de calculs.

2 Compétences acquises

Durant ces mois de travail sur ce projet, nous avons beaucoup appris et évolués :

* Travailler en équipe n'est pas chose aisée et cela nécessite une bonne organisation, coordination et cohésion d'équipe! Cette expérience aura été bénéfique pour nous tous et chacun a su amener ses qualités personnelles.

* Nous avons complété notre bagage scientifique avec de nouvelles connaissances tels que la discréttisation et la résolution numérique d'équation différentielle, l'utilisation de bibliothèques scientifiques en Python et la rédaction d'un rapport scientifique en LaTeX.

Annexes

1 Tableaux des différences finies

Pour la méthode différences finies, on a utilisé les tableaux suivants pour les schémas des différents ordres, tirés de "Méthodes Numériques, Équations aux Dérivées Partielles (EDP)" de l'institut d'optique Paritech.

Différences finies avant, ordre 1

	u_i	u_{i+1}	u_{i+2}	u_{i+3}	u_{i+4}
$\Delta x u'_i$	-1	1			
$\Delta x^2 u''_i$	1	-2	1		
$\Delta x^3 u'''_i$	-1	3	-3	1	
$\Delta x^4 u^{(4)}_i$	1	-4	6	-4	1

Différences finies arrière, ordre 1

	u_{i-4}	u_{i-3}	u_{i-2}	u_{i-1}	u_i
$\Delta x u'_i$				-1	1
$\Delta x^2 u''_i$				1	-2
$\Delta x^3 u'''_i$			-1	3	-3
$\Delta x^4 u^{(4)}_i$	1	-4	6	-4	1

Différences finies centré, ordre 2

	u_{i-2}	u_{i-1}	u_i	u_{i+1}	u_{i+2}
$2\Delta x u'_i$		-1		1	
$\Delta x^2 u''_i$		1	-2	1	
$2\Delta x^3 u'''_i$	-1	2	0	-2	1
$\Delta x^4 u^{(4)}_i$	1	-4	6	-4	1

Différences finies centré, ordre 4

	u_{i-3}	u_{i-2}	u_{i-1}	u_i	u_{i+1}	u_{i+2}	u_{i+3}
$12\Delta x u'_i$			1	-8	0	8	-1
$12\Delta x^2 u''_i$			-1	16	-30	16	-1
$8\Delta x^3 u'''_i$	-1	-8	13	0	-13	8	-1
$6\Delta x^4 u^{(4)}_i$	-1	12	-39	56	-39	12	-1

2 Solutions pour la partie comparaison

Amplitude en fonction de x et t

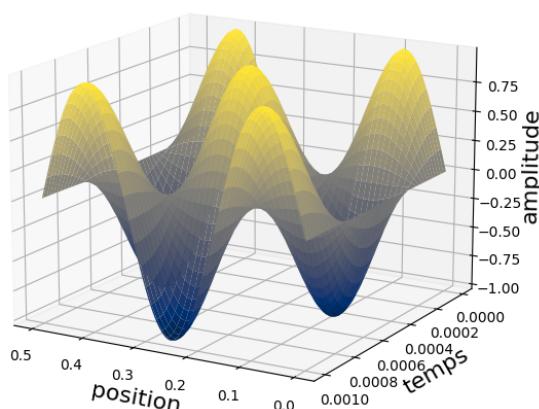


FIGURE 21 – Solution exacte

Amplitude en fonction de x et t

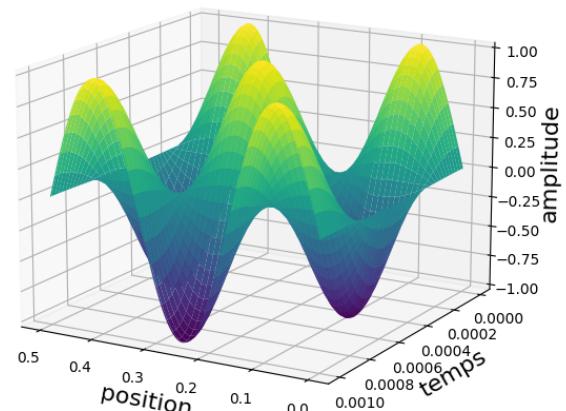


FIGURE 22 – Euler explicite

Amplitude en fonction de x et t

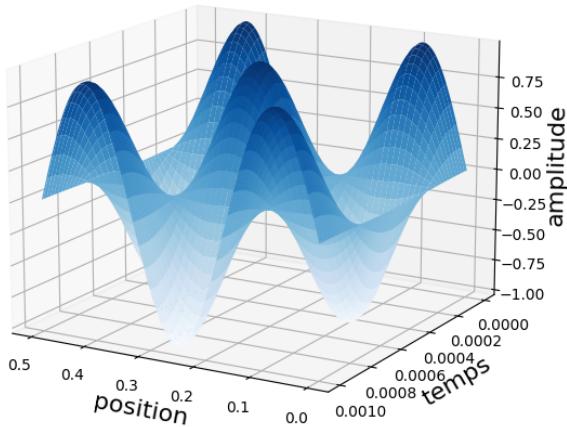


FIGURE 23 – Euler implicite

Amplitude en fonction de x et t

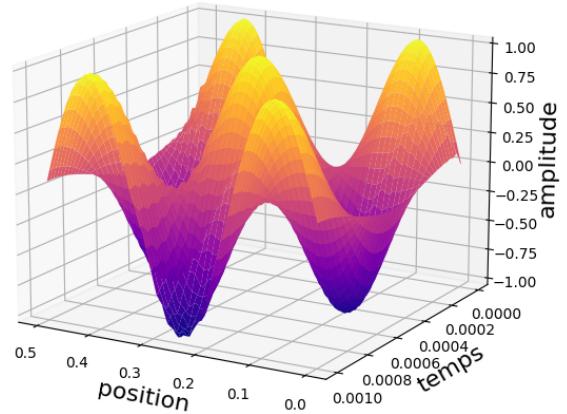


FIGURE 24 – Runge-Kutta

3 Algorithmes

3.1 Euler explicite

1. Initialisation des pas Δx et Δt
Initialisation des nombre de points du temps Nt et du nombre de points de l'espace Nx
2. Calculs des conditions initiales u_i^0 et de u_i^1
3. Calcul de la matrice trigonale A
4. Tant que $n < Nt$ faire :
 - Calculs de U_i^{n+1} : $U_i^{n+1} = A \cdot U_i^n - U_i^{n-1} + C_i$

3.2 Euler implicite

1. Initialisation des pas Δx et Δt
Initialisation des nombre de points du temps Nt et du nombre de points de l'espace Nx
2. Calculs des conditions initiales u_i^0 et de u_i^1
3. Calcul de la matrice trigonale A et de son inverse A^{-1}
4. Tant que $n < Nt$ faire :
 - Calcul de U_i^{n+1} : $U_i^{n+1} = A^{-1} (U_i^n - \frac{1}{2} \cdot U_i^{n-1})$

3.3 Runge-Kutta

Algorithme utilisé dans le code python :

1. Initialisation des pas Δx et Δt
Initialisation des nombre de points du temps Nt et du nombre de points de l'espace Nx
2. Calculs des conditions initiales u_i^0 et de $(\frac{\partial u_i^n}{\partial t})_i^0$
3. Tant que $n < Nt$ faire :
 - Tant que $i < Nx$ faire :
 - Calcul de $((k_0)_i)$
 - Calcul de $((l_0)_i)$
 - Calcul de $((k_1)_i)$
 - Calcul de $((l_1)_i)$
 - ...
 - Calculs de u_i^{n+1} et $(\frac{\partial u_i^n}{\partial t})_i^n$

4 Lien Github

Lien vers le répertoire Github contenant les différents codes python utilisés au cours du projet :
<https://github.com/Rudioo/Projet-Musique.git>

5 Tests complémentaires pour la méthode de Runge-Kutta

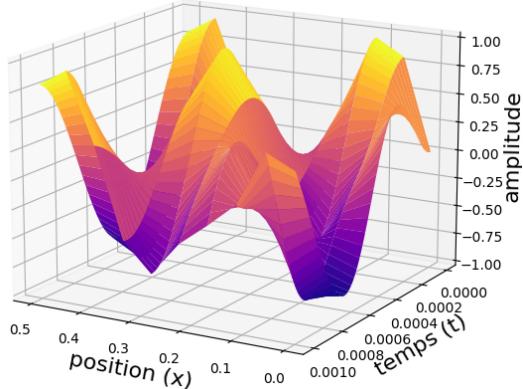


FIGURE 25 – $\Delta x = 5^{-2}m$
 $\Delta t = 10^{-7}s$
 $\lambda = 6.8 \cdot 10^{-4}$

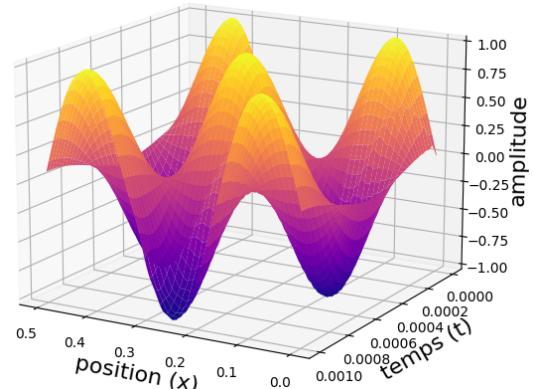


FIGURE 26 – $\Delta x = 5^{-3}m$
 $\Delta t = 10^{-6}s$
 $\lambda = 0.068$

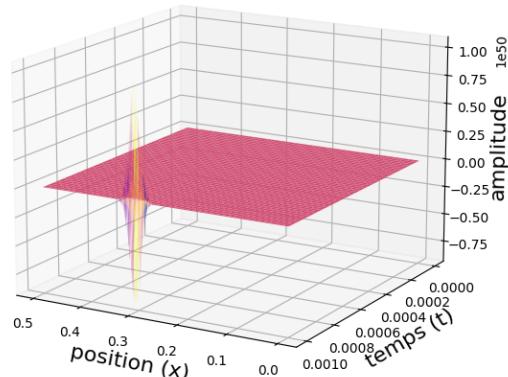


FIGURE 27 – $\Delta x = 5^{-4}m$
 $\Delta t = 1.47 \cdot 10^{-6}s$
 $\lambda = 1$

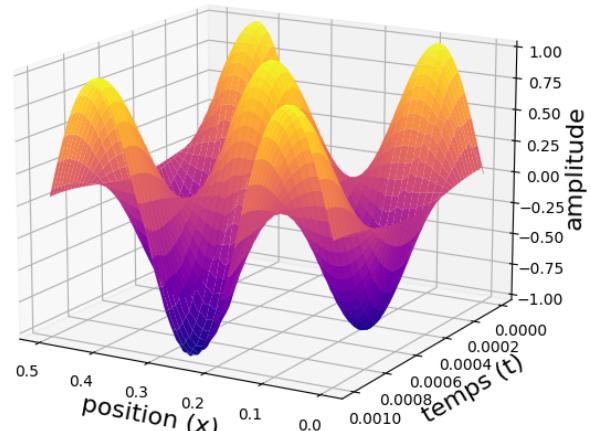


FIGURE 28 – $\Delta x = 3^{-3}m$
 $\Delta t = 10^{-6}s$
 $\lambda = 0.113$

Bibliographie

- [1] Help with using the runge-kutta 4th order method on a system of 2 first order ode's. <https://math.stackexchange.com/questions/721076/help-with-using-the-runge-kutta-4th-order-method-on-a-system-of-2-first-order-od>.
- [2] Runge-kutta method for pde. <https://math.stackexchange.com/questions/1341074/runge-kutta-method-for-pde>.
- [3] Wikipedia : Runge-kutta. <https://fr.wikipedia.org/wiki/M%C3%A9thode_de_Runge-Kutta>.
- [4] Marc BUFFAT. Équation des ondes. <https://perso.univ-lyon1.fr/marc.buffat/COURS/COURSDF_HTML/node27.html>, 2008.
- [5] Eric Goncalvès da Silva. Méthode et analyses numériques. Méthodes et Analyses Numériques. Engineering school. Institut Polytechnique de Grenoble, 2007, pp.99. <cel-00556967>, 2007.
- [6] Institut d'optique Paritech. Méthodes numériques, Équations aux dérivées partielles (edp).
- [7] Dominique Lefebvre. L'équation de d'alembert. <<http://www.tangentex.com/EquationAlembert.htm>>, 2016.
- [8] Jimmy Roussel. Méthode de runge-kutta. <<https://femto-physique.fr/analyse-numerique/runge-kutta.php>>, 2016.