



EURECOM
Department of Digital Security
Campus SophiaTech
CS 50193
06904 Sophia Antipolis cedex
FRANCE

Research Report RR-25-347

Comparative Analysis of Finite Difference and Finite Element Method for Audio Waveform Simulation

February 3rd, 2025

Author

Juliette FLORIN

Advisors

Massimiliano Todisco, Michele Panariello, and Nicholas Evans

Tel : (+33) 4 93 00 81 00
Fax : (+33) 4 93 00 82 00
Email : juliette.florin@eurecom.fr

¹EURECOM's research is partially supported by its industrial members: BMW Group, IABG, Orange, SAP, NortonLifeLock.

Comparative Analysis of Finite Difference and Finite Element Method for Audio Waveform Simulation

Juliette FLORIN

Abstract

In many industries, including aerospace and defense, waveform analysis is commonly conducted to compute the resonance of physical objects, with the Finite Element Method (FEM) being the standard approach. The Finite Difference Method (FDM) is seldom used, and this preference is often stated without formal justification in the literature. In this work, the accuracy, feasibility, and time of simulation of FEM and FDM are compared by simulating the vibration of a guitar string. Python simulations for both methods are implemented, and their results are compared against analytical solutions and experimental data. Additionally, FDM is applied to analyze the sound of a cycling bell to assess its reliability compared to a real cycling bell. Final results show that both FEM and FDM yield similar error margins and accurately predict the system's behavior. Moreover, the errors from FEM and FDM follow the same periodicity with a phase shift when varying the assumed analytical tension and without a phase shift when changing the time interval. However, FEM converges faster with increasing mesh complexity, whereas FDM demonstrates quicker computational performance and achieves stable solutions even with bigger time intervals. Despite this FDM is limited to simpler configurations and often demands extensive mathematical formulation, which can become cumbersome for intricate shapes. For example, modeling a hemispherical object using FDM results in significant simulation times and big calculations. In conclusion, while FDM may offer faster convergence and computation time in certain cases, FEM remains the preferred method in industrial contexts due to its flexibility, scalability, and ease of implementation for complex geometries.

Index Terms

finite difference method, finite element method, resonance, acoustics, wave propagation, fundamental frequency, harmonics, D'Alembert equation, natural frequencies, vibration modes, Fourier transform, Euler-Bernoulli beam equation, hemisphere simulation, string resonance, longitudinal wave, damped oscillator, physical modelling synthesis

Contents

1	Introduction	1
1.1	Objective	1
1.2	Context and Motivation	1
1.3	Models and Theory	2
1.4	Finite Element Method (FEM)	2
1.5	Finite Difference Method (FDM)	2
2	Pressure to sound	3
2.1	Acoustic pressure model	3
2.2	Conversion of acoustic pressure to audio amplitude	4
3	The guitar string without damping and without modulus of elasticity	5
3.1	Model	5
3.2	Theory	6
3.3	Analytical solution	6
3.4	Parameters for a nylon string B3 (247 Hz)	8
3.5	Finite Difference Method	8
3.5.1	Theory	9
3.5.2	Computation and result	10
3.6	Finite Element Method	11
4	Comparing FEM and FDM guitar string	14
4.1	Variation of applied tension	14
4.2	Variation of the time interval	18
4.3	Variation in the number of nodes	22
4.4	Analyzing the effects of modified parameters on the time of simulation	26
5	To compare to reality: guitar with damping and without the modulus of elasticity	28
5.1	Comparison to Reality	29
5.2	Important note on the simulation	33
6	The cycling bell with damping and modulus of elasticity	34
6.1	Model	34
6.2	Theory	34
6.3	Analytical resolution	35
6.3.1	First case	38
6.3.2	Second case	39
6.3.3	Numerical test	41
6.4	Comparing analytical to reality	41
6.5	Finite Difference Method	43

7	Conclusion	47
7.1	General conclusion	47
7.2	Additional possible work	48

List of Figures

1	FEM representation example where the problem is split into multiple (numbered) elements	2
2	FDM representation example where the problem is split into multiple (numbered) nodes	3
3	Representation of R during pressure calculation of a string	4
4	Example string for a mass of 1 kg, the tension is $T = 9.81\text{N}$	5
5	Shape in time of a nylon string B3 (247 Hz) plucked at 0.18 m for one second with 81 nodes ($N = 80$)	10
6	Pressure in time for a nylon string B3 (247 Hz) for one second with 81 nodes	11
7	FFT of the signal of the guitar	11
8	Zero-mean error between the measured and analytical fundamental frequency according to tension N	15
9	Zero-mean error of third harmonic frequency according to tension N	16
10	Zero-mean error of fifth harmonic frequency according to tension N	16
11	Error of fundamental frequency according to tension	17
12	Error of third harmonic frequency according to tension	17
13	Error of fifth harmonic frequency according to tension	18
14	Zero-mean error of the fundamental frequency compared to analytical when changing the time step	19
15	Zero-mean error of the second harmonic compared to analytical when changing the time step	19
16	Zero-mean error of the third harmonic compared to analytical when changing the time step	20
17	Fundamental error FFT frequency interval compared to Amplitude of errors when changing the time step	21
18	Error of the fundamental frequency compared to analytical when changing the time step	21
19	Error of the second harmonic compared to analytical when changing the time step	22
20	Error of the third harmonic compared to analytical when changing the time step	22
21	Positive relative error of the fundamental frequency compared to analytical according to the number of nodes	23
22	Positive relative error of the second harmonic compared to analytical according to the number of nodes	24
23	Positive relative error of the third harmonic compared to analytical according to the number of nodes	24
24	Positive relative error of the fourth harmonic compared to analytical according to the number of nodes	25
25	Positive relative error of the fifth harmonic compared to analytical according to the number of nodes	25

26	Time of FEM and FDM simulations of a string according to tension	26
27	Time of FEM and FDM simulations according to the time interval chosen in the discretizations	27
28	Time of FEM and FDM simulations of a string according to the number of cuts	27
29	Real guitar of string length 0.655 m played on the fifth string at 0.18 m	30
30	Simulation result of this particular string	31
31	Damping looked at on <i>Audacity</i> , the upper waveform is the simu- lation and the bottom one is the real signal	32
32	FFT through <i>Audacity</i> of the real signal, peak at 252 Hz	32
33	FFT through <i>Audacity</i> on the simulated waveform, peak at 244 Hz	33
34	The model of cycling bell	34
35	$p(1)$ according to Ω in $m = 0$	39
36	Time signal of the audio file generated viewed on <i>Audacity</i>	41
37	Damping effect on the fundamental frequency and first harmonic in logarithmic scale with linear regression	42
38	Aluminium bell	42

1 Introduction

Accurate numerical modeling of sound-producing structures is fundamental in musical acoustics, audio engineering, virtual instrument synthesis, and many other domains. Among the available computational methods, the Finite Element Method (FEM) is commonly employed for simulating the vibrational behavior of complex structures, such as musical instruments [1]. However, its effectiveness compared to alternative approaches, such as the Finite Difference Method (FDM), remains underexplored.

1.1 Objective

This study aims to compare the performance and accuracy of the FDM and the FEM in simulating audio waveforms for various physical phenomena. Specifically in the domain of predicting the resonant frequencies, mode shapes, and acoustic response of physical objects. The goal is to generate and analyze the sound waveforms produced by two chosen objects: a guitar string and a simple cycling bell. The project will investigate how the physical properties of these objects influence the behavior of both models and the resulting audio simulations.

The results of this research will contribute to a deeper understanding of computational acoustics, which is used for theoretical and practical applications in musical instrument modeling, acoustic design, and digital sound synthesis. Moreover, it will assess the trade-off between FDM and FEM when choosing what method to use.

1.2 Context and Motivation

A wave is an oscillatory phenomenon (a quantity that varies over time) propagating through space. In acoustics, it refers to vibrations of matter, meaning a mechanical motion oscillating in a continuous medium, whether fluid or solid. Examples of waves include the vibration of a string, waves on the surface of water, sound propagating through air, etc. Each object, for instance, a hemispherical shell representing a cycling bell or a string of a guitar, when experiencing mechanical vibration, produces frequencies that correspond to specific resonances.

Studying these vibrations requires solving partial differential equations that describe different vibration types, such as longitudinal, transverse, flexural, and torsional. For simpler systems like strings or beams, these equations include the second-order D'Alembert equation and the fourth-order Euler-Bernoulli beam equation. The vibrational behavior of hemispherical and string structures depends on natural frequencies, vibration modes, geometry, and material properties. Understanding these properties is crucial for engineering and scientific applications, with various methods available for analysis.

In the industry today, FEM is mainly used [2]. In many reports, it is also mentioned to be easier to compute and more efficient than the FDM [1] [3]. The

statements comparing the two are made without any references. In the context of acoustic waves and computing the waveform from a vibrating object, we would like to provide insight on both methods.

1.3 Models and Theory

At least two methods exist that can be used to model a physical phenomenon. Other methods are mainly derived cases from these general ones. On the one hand, there is the FEM used to model complex objects without little simplifications. On the other hand, the FDM has the same purpose but is also implemented when a solution to an equation is only reachable through approximations. Those two methods used in modeling physical phenomena should give the same answer as the theory. However, they are numerical methods; thus, the results will convey an error, no matter the model.

1.4 Finite Element Method (FEM)

The FEM is a numerical technique for solving differential equations, particularly useful for complex geometries and variable material properties [4]. The process involves transforming the strong form of the governing equations into a weak form, followed by discretization into finite elements. This results in the need to compute global system matrices, such as the mass and stiffness matrices.

In FEM, division into small elements is done to approximate the solution using interpolation functions (shape functions) within each element. The unknowns, typically nodal displacements or field variables, are determined by assembling the contributions from individual elements into a global system of equations. We can see in the figure 1 an example of how to cut down a string into small elements. Each element has given length, weight, and properties, and must represent a small portion of the general object described.

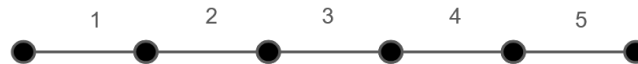


Figure 1: FEM representation example where the problem is split into multiple (numbered) elements

1.5 Finite Difference Method (FDM)

Sometimes referred to as the Euler Method, the FDM is not widely adopted in the industry but remains highly effective when applied to well-understood problems. FDM is a numerical technique for solving differential equations by directly

approximating derivatives at discrete points within the computational domain. Unlike the FEM, FDM does not require a weak formulation but instead discretizes the strong form of the governing equations.

This method approximates derivatives using finite difference approximations, including forward, backward, and central differences. These approximations transform the differential equations into algebraic equations, linking function values at discrete grid points. We can see in the figure 2 how small nodes can represent a string.



Figure 2: FDM representation example where the problem is split into multiple (numbered) nodes

2 Pressure to sound

Once we have the amplitude in time of a given model, we can compute the waveform at a distance, given the pressure felt at that distance in time. In an acoustic simulation, the pressure generated by a source can be calculated using various physical parameters. If SI units are used for all quantities, the resulting acoustic pressure is expressed in Pascals (Pa). However, to create a digital audio file, it is necessary to convert this acoustic pressure into amplitude and to quantize it and normalize it within the range $[-1, 1]$.

2.1 Acoustic pressure model

The acoustic pressure $p(t)$ generated at a distance R by a point source at position x is given in the equation 1.

$$p(t) = \frac{\rho_0 c_0}{4\pi} \frac{\partial u_R(x, t_r)}{\partial t} \quad (1)$$

where:

- $p(r, t)$ is the acoustic pressure in Pascals Pa
- ρ_0 is the air density in kg m^{-3}
- c_0 is the speed of sound in air in m s^{-1}
- $u_R(x, t_r)$ is the transverse displacement of the string in m with x being the position of the point source and t_r the time when the displacement happened
- $\frac{\partial u_R}{\partial t_r}$ is the transverse velocity at the distance R from the point source in m s^{-1}

Let us note:

$$[p(t)] = \text{kg m}^{-3} \cdot \text{m s}^{-1} \cdot \text{m s}^{-1} = \text{Pa} \quad (2)$$

The distance R , in m, appears when calculating $\frac{\partial u_R(x, t_r)}{\partial t}$. In this project, the position of the calculated pressure is fixed at 1 m from one given point of the source. For each point x of the source, the corresponding R is calculated with the Pythagorean theorem. Figure 3 illustrates R in the case of a string.

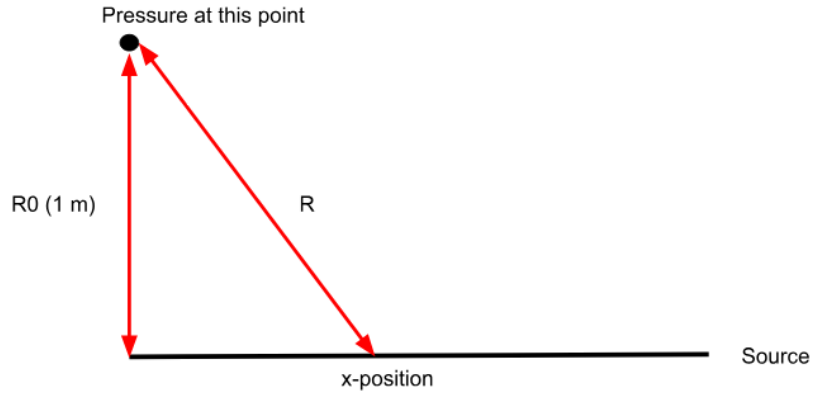


Figure 3: Representation of R during pressure calculation of a string

As the distance increases, the transverse velocity will decay by $1/R$. Thus with $\frac{\partial u_R(x, t_r)}{\partial t} = \frac{1}{R} \frac{\partial u(x, t - R/c_0)}{\partial t}$ the calculated pressure is:

$$p(t)_R = \frac{\rho_0 c_0}{4\pi R} \frac{\partial u(x, t - R/c_0)}{\partial t} \quad (3)$$

Finally, to generate the pressure in time of a source, we add all the pressure generated at a given time by each point of a source. Each point of the source generates a given $p(t)$ depending on its calculated R as mentioned in equation 3.

2.2 Conversion of acoustic pressure to audio amplitude

Several steps must be followed to convert the calculated acoustic pressure into an amplitude compatible with a digital audio recording. The first step is to normalize the acoustic pressure $p(t)$ by a reference pressure p_{\max} , to avoid saturation in the audio file. The normalization formula is:

$$A(t) = \frac{p(t)}{p_{\max}} \quad (4)$$

where p_{max} is the maximum pressure in the simulation, and $A(t)$ is the normalized amplitude within the range $[-1, 1]$.

Once the pressure is normalized, the amplitudes can be adapted to match digital audio formats. The goal is to map to the digital audio format. For example, for a 16-bit PCM audio file, the normalized amplitude $A(t)$ can be converted to an amplitude within the range $[-32768, 32767]$ as follows:

$$\text{Amplitude 16-bit} = \lfloor A(t) \cdot 32767 \rfloor \quad (5)$$

Using Python, we can directly write the audio file in WAV format. Following these steps, the acoustic pressure calculated in a simulation can be converted into an appropriate audio amplitude, creating a digital audio file compatible with common standards (such as 16-bit PCM). This conversion ensures that the results from acoustic simulations can be easily analyzed or used in real-world audio applications.

3 The guitar string without damping and without modulus of elasticity

We are looking for the acoustic waves created when playing a guitar string without the flexion or the damping part of the equation [5].

3.1 Model

Consider a string as seen in figure 4 stretched along the horizontal axis between two fixed nodes, A and B . To visualize the tension T of the string, imagine the string is attached at point A and pulled by a mass m via a pulley at point B . The tension T is given by: $T = mg$ where g is 9.81 N/kg is the acceleration due to gravity. When the string is at rest in equilibrium, the forces of tension in both directions cancel each other out (T and $-T$). The length of the string between its two fixed nodes is denoted as a product of A and B . The linear mass density string (mass per unit length) is represented by μ .

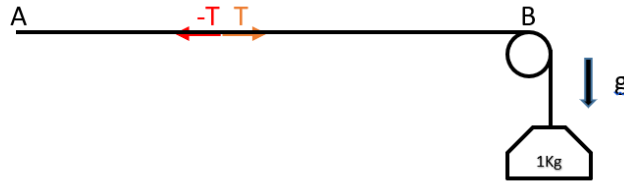


Figure 4: Example string for a mass of 1 kg, the tension is $T = 9.81\text{N}$.

Damping is not considered, as it does not affect the natural frequencies of the waveform. The modulus of elasticity is also neglected in the case of a nylon string,

which means flexion is not included in the model. This simplification is valid for nylon due to its low stiffness, but not for metal strings, where the modulus of elasticity significantly contributes to flexural effects and must be accounted for [5].

3.2 Theory

The wave equation for transverse vibrations of the string, derived by D'Alembert, is given by:

$$\frac{\partial^2 y(x, t)}{\partial t^2} = \frac{T}{\mu} \frac{\partial^2 y(x, t)}{\partial x^2} \quad (6)$$

where:

- $y(x, t)$ represents the transverse displacement of the string at position x and time t
- T is the tension of the string (in N)
- μ is the linear mass density of the string (in kg m^{-1})
- x is the spatial coordinate along the length of the string (in m)

This equation 6 describes the string's motion under the influence of tension and its mass distribution.

The speed c (in m s^{-1}) of a transverse wave on the string is related to the tension T and the linear mass density μ by the following equation:

$$c = \sqrt{\frac{T}{\mu}} \quad (7)$$

This relationship shows that the wave speed increases with the tension and decreases with the mass per unit length of the string. The wave propagates faster for a string with higher tension, while a string with higher mass density will slow down the wave.

3.3 Analytical solution

The equation describing the transverse vibrations of a stretched string is the wave equation:

$$\frac{\partial^2 y(x, t)}{\partial t^2} = c^2 \frac{\partial^2 y(x, t)}{\partial x^2} \quad (8)$$

where $y(x, t)$ is the transverse displacement of the string at position x and time t .

For a vibrating string fixed at both ends (i.e. $y(0, t) = 0$ and $y(L, t) = 0$), we seek a solution in the form of standing waves. Thus, we assume a solution of the form:

$$y(x, t) = f(x)g(t) \quad (9)$$

where $f(x)$ is a spatial function that depends on the position x along the string, and $g(t)$ is a temporal function that depends on time t . Substituting $y(x, t) = f(x)g(t)$ into the wave equation, we get:

$$f(x)g''(t) = c^2 f''(x)g(t) \quad (10)$$

Dividing both sides of this equation by $f(x)g(t)$ gives:

$$\frac{g''(t)}{g(t)} = c^2 \frac{f''(x)}{f(x)} \quad (11)$$

Since the left-hand side depends only on t and the right-hand side depends only on x , both sides must be equal to a constant, which we will call $-\omega^2$, where ω is the angular frequency. This gives us two separate equations:

$$g''(t) + \omega^2 g(t) = 0 \quad (12)$$

$$f''(x) + \frac{\omega^2}{c^2} f(x) = 0 \quad (13)$$

The general (real-valued) solution of equation 12 and 13 are respectively:

$$g(t) = A \cos(\omega t) + B \sin(\omega t) \quad (14)$$

$$f(x) = C \cos\left(\frac{\omega}{c}x\right) + D \sin\left(\frac{\omega}{c}x\right) \quad (15)$$

The boundary conditions are that the string is fixed at both ends:

$$f(0) = 0 \quad \text{and} \quad f(L) = 0 \quad (16)$$

We apply these conditions to the solution for $f(x)$. This gives us that the quantized values for ω are:

$$\omega_n = \frac{n\pi c}{L} \quad \text{where} \quad n = 1, 2, 3, \dots \quad (17)$$

We also have that at time 0, there is no speed:

$$g'(0) = 0 \quad (18)$$

With the boundary conditions, we get the result for $g(t)$, according to the equation 14:

$$g(t) = A \cos(\omega t) \quad (19)$$

The general solution for the displacement of the string is thus an infinite sum of the vibrating modes:

$$y(x, t) = \sum_{n=1}^{\infty} [A_n \cos(\omega_n t)] \sin\left(\frac{n\pi}{L}x\right) \quad (20)$$

where the coefficients A_n are determined by the initial displacement.

Modes of vibration The angular frequency is related to the frequency and gives us all the superposing modes in the final waveform. This tells us what the fundamental frequency and the harmonics will be [6] [7].

$$\omega_n = 2\pi f_n = \frac{n\pi c}{L} \quad (21)$$

The fundamental wavelength of vibration, the basic mode, corresponds to the string having two vibration nodes located at the fixed point. The length of the string corresponds to half the wavelength. Hence, the fundamental frequency of vibration is given by:

$$f_1 = \frac{1}{2L} \sqrt{\frac{T}{\mu}} \quad (22)$$

where:

- L is the length of the string (in m)

The harmonics of this fundamental will be nf_1 with n an integer. The harmonics depend on where the string is plugged. D'Alembert's wave equation for transverse vibrations of a plucked string describes the string's motion under tension. The fundamental frequency of vibration is determined by the tension of the string, the linear mass density, and the length of the string. This theory is widely used to understand the behavior of musical strings.

3.4 Parameters for a nylon string B3 (247 Hz)

The parameters for the nylon string chosen are [8] [9] [10]:

- $L = 0.65$ m
- $\mu = 0.000582$ kg m⁻¹
- $T = 60$ N

Using the wave speed formula (equation 7) for a string and the fundamental frequency calculation (equation 22), we calculate the wave speed to 321 m s⁻¹ and the fundamental frequency to 247 Hz

3.5 Finite Difference Method

We start by using FDM.

3.5.1 Theory

In this method, derivatives and second derivatives are replaced by their estimates:

$$\frac{\partial u}{\partial x} \approx \frac{u(x + \delta_x, t) - u(x, t)}{\delta_x}, \quad (23)$$

$$\frac{\partial u}{\partial t} \approx \frac{u(x, t + \delta_t) - u(x, t)}{\delta_t}, \quad (24)$$

$$\frac{\partial^2 u}{\partial x^2} \approx \frac{u(x + \delta_x, t) - 2u(x, t) + u(x - \delta_x, t)}{\delta_x^2}, \quad (25)$$

$$\frac{\partial^2 u}{\partial t^2} \approx \frac{u(x, t + \delta_t) - 2u(x, t) + u(x, t - \delta_t)}{\delta_t^2}. \quad (26)$$

Here, δ_x and δ_t are tiny intervals in space and time, respectively. We discretize the space in length x and time t , and rewrite D'Alembert's equation:

$$\begin{aligned} & \frac{u(t + \delta_t, x) - 2u(t, x) + u(t - \delta_t, x)}{\delta_t^2} \\ & - c^2 \frac{u(t, x + \delta_x) - 2u(t, x) + u(t, x - \delta_x)}{\delta_x^2} = 0 \end{aligned} \quad (27)$$

By setting $\gamma = \frac{c^2 \delta_t^2}{\delta_x^2}$ in equation 27, we get:

$$u(t + \delta_t, x) = \gamma(u(t, x - \delta_x) + u(t, x + \delta_x)) + 2(1 - \gamma)u(t, x) - u(t - \delta_t, x) \quad (28)$$

We divide the string of length L into $N - 1$ segments of length $\delta_x = \frac{L}{N-1}$, and the observation time A into K intervals of duration $\delta_t = \frac{A}{K-1}$. We study vibrations at the corresponding node and instants:

$$t_k = (k - 1)\delta_t, \quad x_n = (n - 1)\delta_x, \quad u_{n,k} = u(x_n, t_k).$$

For a guitar string, the initial deformation is due to the plucking of the string at a position x_p . This can be expressed as:

$$u(x, 0) = \begin{cases} h \cdot \frac{x}{x_p}, & \text{if } 0 \leq x \leq x_p, \\ h \cdot \frac{L-x}{L-x_p}, & \text{if } x_p \leq x \leq L, \end{cases} \quad (29)$$

where h is the amplitude of the pluck, L is the length of the string, and x_p is the position of the pluck. The initial velocity is zero everywhere:

$$\frac{\partial u}{\partial t}(x, 0) = 0, \quad \forall x \in [0, L] \quad (30)$$

In discrete notation, the initial conditions become:

$$u_{i,0} = \begin{cases} h \cdot \frac{x_i}{x_p}, & \text{if } 0 \leq x_i \leq x_p \\ h \cdot \frac{L-x_i}{L-x_p}, & \text{if } x_p \leq x_i \leq L \end{cases} \quad (31)$$

$$u_{i,1} = u_{i,0}, \quad \forall i \in [0, N] \quad (32)$$

For a guitar string fixed at both ends, the boundary conditions are for all t :

$$u(0, t) = u(L, t) = 0 \quad (33)$$

In discrete notation, this becomes:

$$u_{0,j} = u_{N,j} = 0, \quad \forall j \in [0, K] \quad (34)$$

3.5.2 Computation and result

Figure 5 represents the string changing over time after being plucked. Figure 6 shows the pressure felt at 1m of the string over time. This will be used to create the waveform. Figure 7 represents the Fast Fourier Transform (FFT) of the waveform of the sound of a plucked guitar string. The spectrum observed contains a large quantity of harmonics. Noticeably, the relative amplitudes of these harmonics depend on where the string is plucked. This phenomenon can be explained by the physics of standing waves and the resulting interaction between the fundamental frequency and its harmonics. For example, if you pluck the string at half its length (i.e. at its midpoint), the even harmonics are absent. This relationship can be understood by considering the Fourier series representation of the wave in equation 20. The position where the string is plucked determines which harmonics are most effectively excited, as different positions correspond to different initial conditions for the sinusoidal components of the wave. In half the distance, the sinusoid is null for the even harmonics.

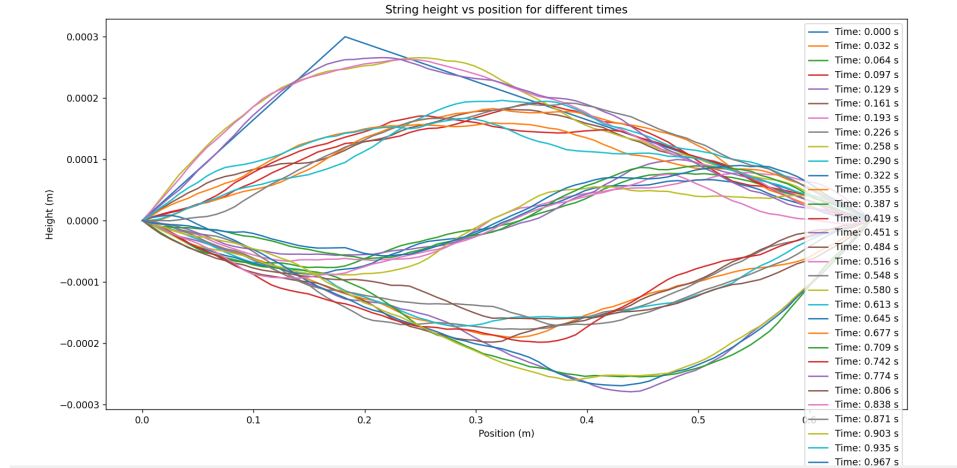


Figure 5: Shape in time of a nylon string B3 (247 Hz) plucked at 0.18 m for one second with 81 nodes ($N = 80$)

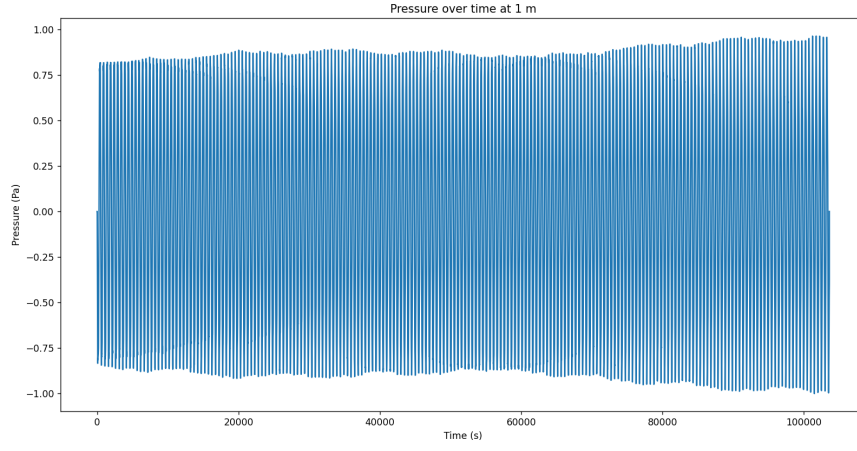


Figure 6: Pressure in time for a nylon string B3 (247 Hz) for one second with 81 nodes

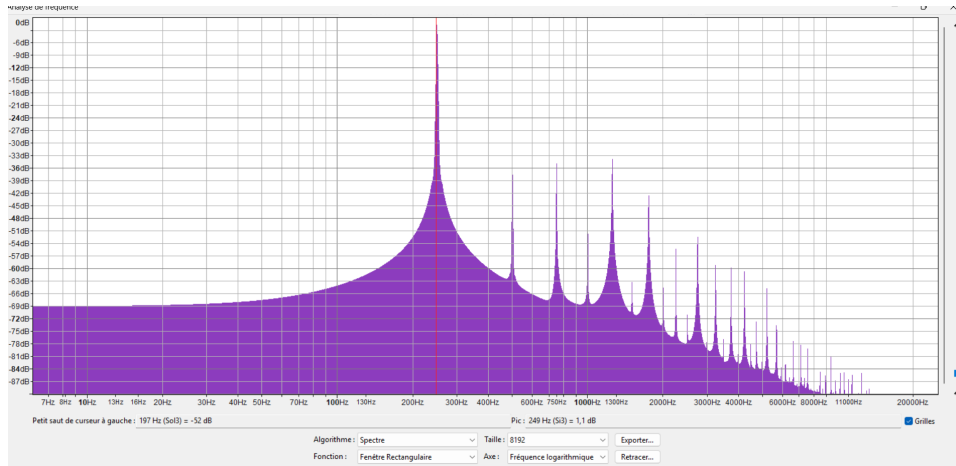


Figure 7: FFT of the signal of the guitar

3.6 Finite Element Method

The weak form of the wave equation is given as [4] [11] [12]:

$$M\ddot{U}(t) + KU(t) = 0 \quad (35)$$

where:

- M : Global mass matrix (from inertia term),

- K : Global stiffness matrix
- $U(t)$: Vector of nodal displacements,
- $\ddot{U}(t)$: Vector of nodal accelerations.

The goal is to compute U^{n+1} , the displacement at the next time step, given the current displacements (U^n ,) and displacements at the previous time step (U^{n-1}).

To discretize the wave equation over time, we approximate the second derivative $\ddot{U}(t)$ using the central difference scheme:

$$\ddot{U} \approx \frac{U^{n+1} - 2U^n + U^{n-1}}{\Delta t^2} \quad (36)$$

Substitute this into the weak form:

$$\frac{M}{\Delta t^2}(U^{n+1} - 2U^n + U^{n-1}) + KU^n = 0 \quad (37)$$

Simplify for U^{n+1} :

$$U^{n+1} = 2U^n - U^{n-1} - \Delta t^2 M^{-1} K U^n \quad (38)$$

with:

- $2U^n$: The current displacement contributes the most to the next time step.
- $-U^{n-1}$: The inertia (past displacement) opposes changes.
- $-\Delta t^2 M^{-1} K U^n$: The stiffness and mass interaction modifies the displacement over time.

Now that we have U^{n+1} depending on previous U we can compute M and K. To do that we first need to compute their local matrix. To compute the local mass matrix (M_{local}) and local stiffness matrix (K_{local}) for a single finite element in a 1D string using the FEM, follow these steps:

For a 1D element with linear shape functions, we define:

$$N_1(x) = \frac{x_{e+1} - x}{\Delta x}, \quad N_2(x) = \frac{x - x_e}{\Delta x} \quad (39)$$

where:

- x_e and x_{e+1} are the positions of the two nodes of the element,
- $\Delta x = x_{e+1} - x_e$ is the length of the element.

The mass matrix M_{local} is derived from the term:

$$\int_{x_e}^{x_{e+1}} \rho N_i(x) N_j(x) dx \quad (40)$$

For a two-node linear element ($i, j = 1, 2$), the result is:

$$M_{local} = \frac{\rho \Delta x}{6} \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \quad (41)$$

- The diagonal terms (2) represent the inertia contribution of each node.
- The off-diagonal terms (1) represent the coupling between the two nodes.

The stiffness matrix K_{local} is derived from the terms in equation 42 [4] [11] [12].

$$\int_{x_e}^{x_{e+1}} T \frac{\partial N_i}{\partial x} \frac{\partial N_j}{\partial x} dx \quad (42)$$

For linear elements, the derivatives of the shape functions are constant:

$$\frac{\partial N_1}{\partial x} = -\frac{1}{\Delta x}, \quad \frac{\partial N_2}{\partial x} = \frac{1}{\Delta x} \quad (43)$$

Substitute these into the integral:

$$K_{\text{local}} = \frac{T}{\Delta x} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \quad (44)$$

- The diagonals with the value 1 represent the stiffness contribution at each node.
- The off-diagonal terms with the value -1 represent the coupling between the two nodes.

The global mass matrix (M) and global stiffness matrix (K) are assembled by summing the contributions from all elements. Each local matrix is added to the global matrix by mapping its nodes (degrees of freedom) to the global indices. For a string that is fixed at both ends ($u(0, t) = u(L, t) = 0$), the boundary conditions are applied as follows:

- Modify the global mass matrix (M) and stiffness matrix (K) such that the rows and columns corresponding to the boundary nodes are set to zero.
- Set the diagonal entries for the boundary nodes to 1, ensuring their displacements remain zero.

When implementing the global matrices in Python we can write it:

$$M[0, :] = 0, \quad M[-1, :] = 0, \quad M[0, 0] = 1, \quad M[-1, -1] = 1$$

$$K[0, :] = 0, \quad K[-1, :] = 0, \quad K[0, 0] = 1, \quad K[-1, -1] = 1$$

This enforces the fixed displacement at the boundaries while keeping the system consistent. We now put this in a Python code and simulate it.

Now that we have the recursion terms, M and K , the algorithm used in the code will be:

1. Assemble the global mass and stiffness matrices:
 - M : coming from the weak form of the inertia term.

- K : coming from the weak form of the elastic term.

These are computed for all elements and assembled into the global system.

2. Boundary conditions: Apply boundary conditions to M and K , such as fixing the ends of the string (displacement = 0).
3. Iterative update: Use the update formula:

$$U^{n+1} = 2U^n - U^{n-1} - \Delta t^2 M^{-1} K U^n \quad (45)$$

Compute U^{n+1} at every time step, starting with initial conditions:

- U^0 : Initial displacement (e.g., triangular shape from the pluck),
- U^1 : First step, often approximated as $U^1 = U^0$.

4 Comparing FEM and FDM guitar string

With the implementation of both the FEM and the FDM in Python, the objective is to conduct a comparative analysis of the two approaches using various performance metrics. By systematically varying different parameters for each model, as detailed in the following sections, the waveform was computed for both FEM and FDM. Additionally, the expected theoretical frequencies (fundamental and harmonics) were calculated to be used as a reference. Using *Audacity*, each fundamental frequency and its harmonics were recovered. It was achieved using a rectangular FFT of 4096 in size. For all the FFT in this paper a rectangular window was chosen with a size of 4096. The different types of windows like the Bartlett window, Hamming window, rectangular window, Hann window, Blackman window, Blackman-Harris window, Welch window, and different Gaussian windows, were tried, and the rectangular window gave the best resolution of amplitude and frequency. The size of the window was then chosen according to the sampling rate. The goal was to keep the same window size along all the paper; thus, the window size was big enough to minimize the FFT error but small enough to show the frequencies when testing the impact of time interval. The relative error of the frequencies for each method was then computed and plotted across different scenarios, allowing for an evaluation of the accuracy and reliability of FEM and FDM in approximating the expected results.

4.1 Variation of applied tension

The tension is directly related to the frequency and the speed through an analytic formula as seen in equation 22. This is true for the fundamental frequency but also for the harmonics as they are related to the fundamental by nf_1 (named analytic frequencies in the following). To compare both methods, we are setting all the parameters and looking at the effect of a changing tension on the error of the measured frequencies compared to the analytical frequencies.

- Length of the string: $L = 0.655$ m
- Position of plucking: half its length, 0.3275m
- Number of nodes the string is split into: $N = 80$
- Time interval: $\Delta t = 1 \times 10^{-5}$ s
- Number of time points: $N_t = 1 \times 10^5$
- Total simulation time: $T = 1$ s
- Amplitude of plucking: $A = 3 \times 10^{-4}$ m
- Linear mass density: $\mu = 4.30 \times 10^{-4}$ kg m⁻¹

40 waveforms were computed for 40 different tension values between 42 N and 61.5 N every 0.5 step. The values were taken to ensure the different frequencies stayed in the range of a plucked guitar. The resolution is 2.5 Hz, and we plucked the guitar string at half its length at 0.3275 m. Playing at that location: we only have odd harmonic. Once we measured through *Audacity* the frequencies, we calculated the relative error compared to the analytical solution calculated previously.

The zero-mean error centered around zero (Fig.8, Fig.9 and Fig.10) was then plotted. A periodicity of the error is visible for both methods. Both models follow the same error variation with a phase shift as if they complemented each other. The periodicity of this error is very particular. The plots with the means are visible in Figures 11, 12, and 13.

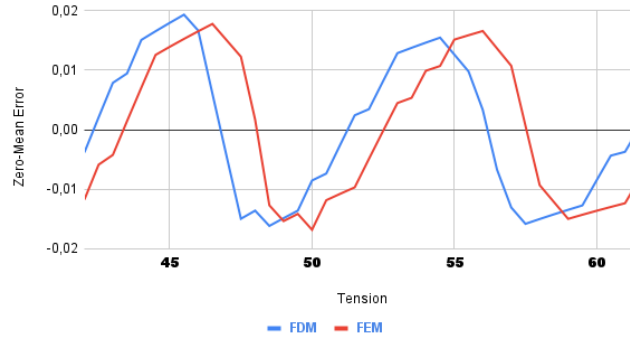


Figure 8: Zero-mean error between the measured and analytical fundamental frequency according to tension N

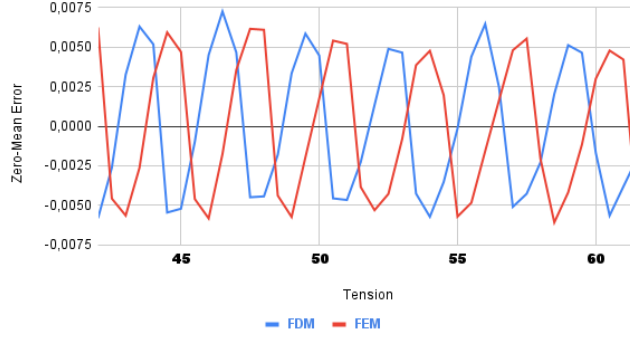


Figure 9: Zero-mean error of third harmonic frequency according to tension N

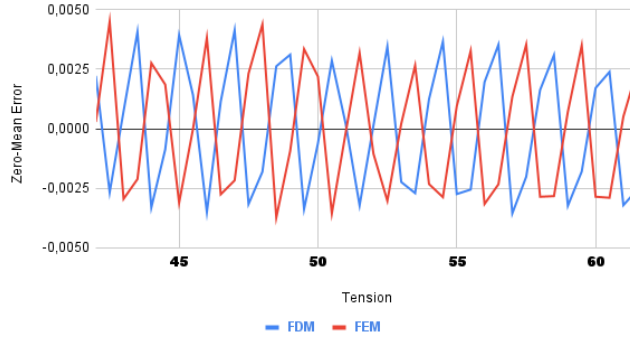


Figure 10: Zero-mean error of fifth harmonic frequency according to tension N

In Figure 8, the fundamental frequency f_1 has a tension periodicity of 9.5 N corresponding to 25 Hz. Examining the higher harmonics (Fig.9, Fig.10) we get:

- $3f_1$: Periodicity 3.10 N or 8 Hz.
- $5f_1$: Periodicity 1.9 N or 5 Hz.

By computing the periodicity ratios:

$$\frac{9.5}{3} = 3.17, \quad \frac{9.5}{5} = 1.9 \quad (46)$$

$$\frac{25}{3} = 8.3, \quad \frac{25}{5} = 5 \quad (47)$$

We observe a clear relationship between the harmonic order and the periodicity of the error. The periodicity of the error is the harmonic order times the periodicity of error of the fundamental.

These periodicity errors may arise due to the discretization in time. Since we sample the frequency domain at discrete intervals, numerical artifacts manifest at

harmonic positions. This behavior suggests that the discretization process influences the periodic detection of frequency variations in a structured manner. Moreover, the error margin is probably not only created by the simulation but also by the FFT that discretizes the signal in the frequency domain. The way the FFT is done impacts the result, but since the same FFT was done on each waveform, the tendency of the error of the simulation to be periodic should follow the same trend.

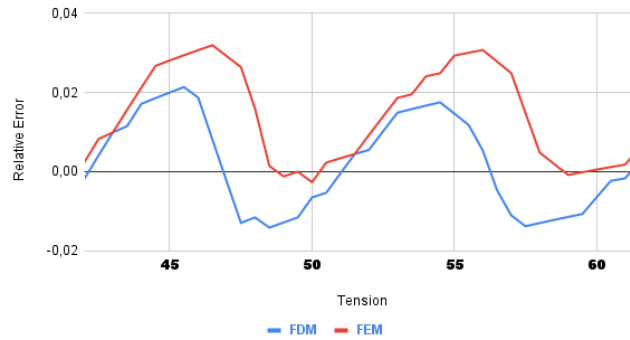


Figure 11: Error of fundamental frequency according to tension

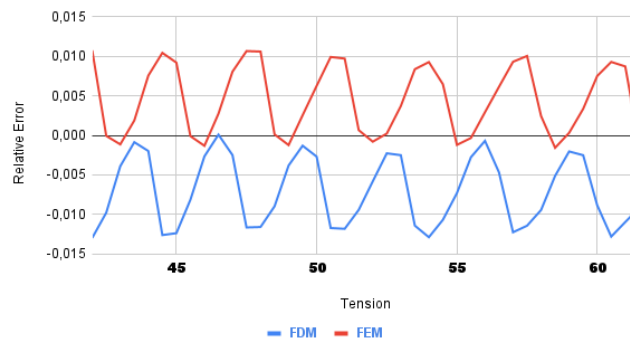


Figure 12: Error of third harmonic frequency according to tension

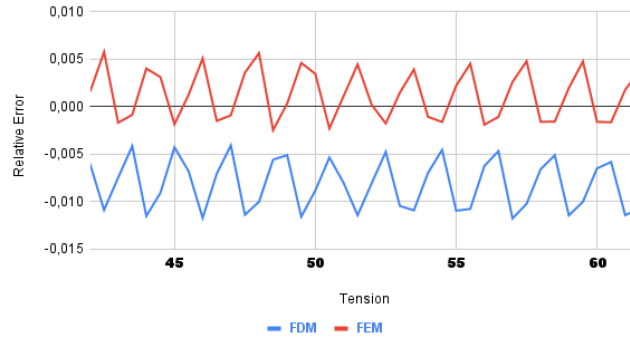


Figure 13: Error of fifth harmonic frequency according to tension

4.2 Variation of the time interval

In numerical simulations, the choice of time step precision significantly impacts the accuracy of results. During one second, we decreased the time interval and analyse how that affected the error. The parameters are fixed to:

- Tension: 42.86 N
- Length: 0.655 m
- Position of plucking: 0.18m
- Number of nodes the string is split into: 80
- Linear mass density: $4.30 \times 10^{-4} \text{ kg m}^{-1}$

A comparison of time-step values reveals that both methods have a minimum time-step to converge.

- FDM converges when the time-step is below $2.27 \times 10^{-5} \text{ s}$
- FEM converges when the time-step is below $1.52 \times 10^{-5} \text{ s}$

This discrepancy suggests that FEM requires a higher level of time precision than FDM to achieve comparable accuracy. The divergence of FEM at lower precision indicates its sensitivity to time discretization, emphasizing the need for finer resolution in FEM-based simulations. For non-NaN values, the data step of 6.7×10^{-4} was repeated 40 times, and the string was plucked at a location of 0.18 m. The plucking position was changed from the previous experience to see if the periodicity was due to the location. We thus have all the odds and even harmonics. Moreover, only the first three frequencies were measured.

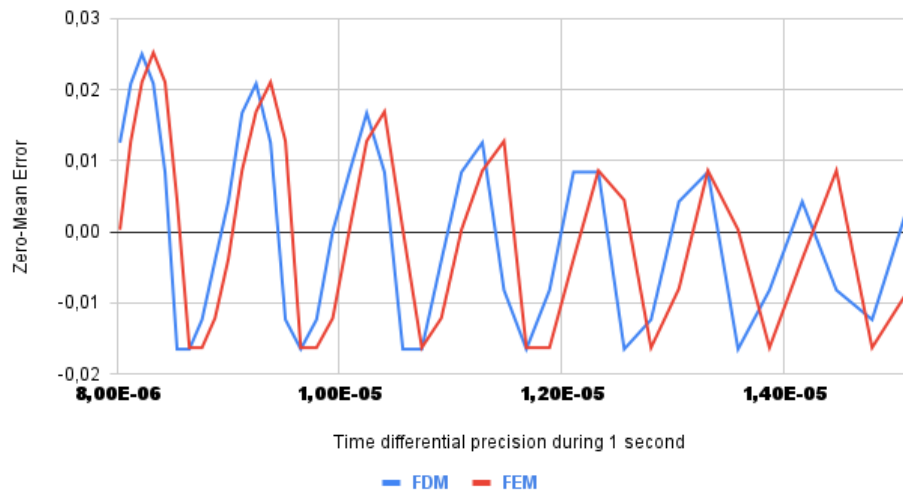


Figure 14: Zero-mean error of the fundamental frequency compared to analytical when changing the time step

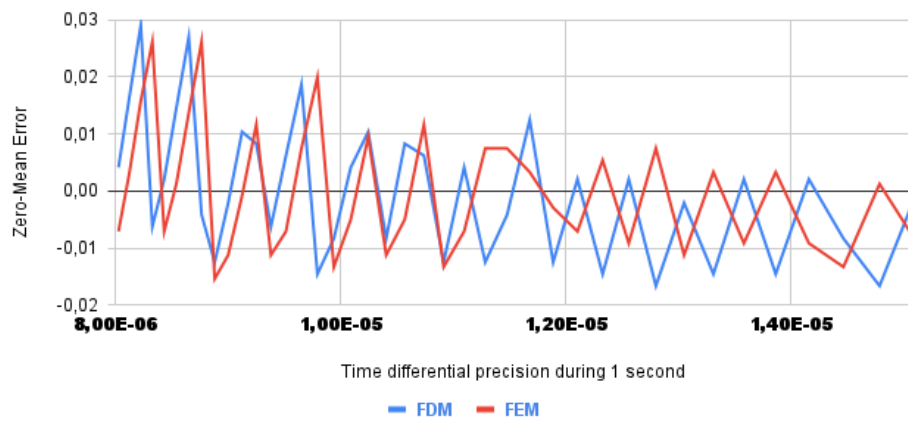


Figure 15: Zero-mean error of the second harmonic compared to analytical when changing the time step

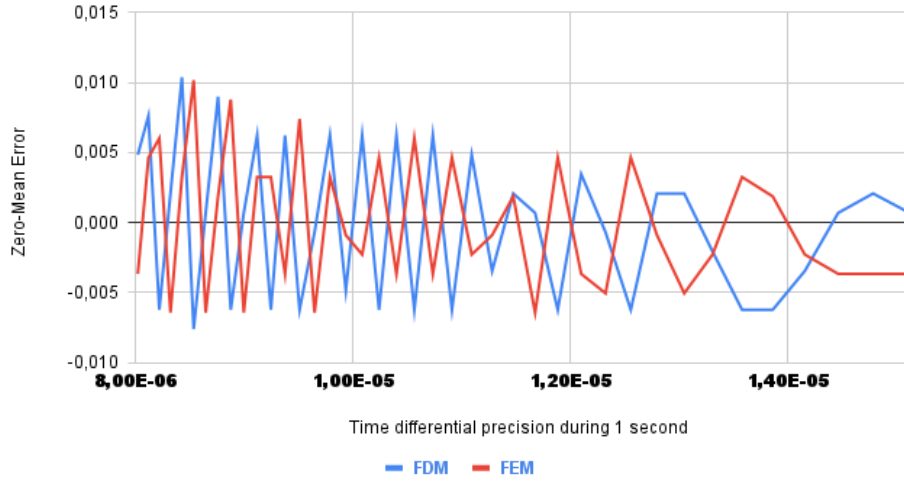


Figure 16: Zero-mean error of the third harmonic compared to analytical when changing the time step

In the zero-mean error frequency plots (Fig.14, Fig.15, Fig.16), once again, it can be seen that both FEM and FDM exhibit the same evolution in error precision. However, the smaller the time interval, the more points we have in time, and the larger the error becomes. On the first array, the data does not seem to follow the expected curve of decreasing error with increased precision of the time measurement. This phenomenon is actually due to the way the FFT is done. The window size (i.e. number of points) remained the same even as the sampling frequency, directly linked to the time interval, increased. As a result, the error in the FFT measurement also increased. Figure 17 represents the evolution of the fundamentals' error precision as the sampling frequency increases. The curves represent the evolution of the fundamentals' error precision due to the FFT, the measurement in the FEM, and the measurement in the FDM. All three curves follow the same trend. This explains the previous results. The FFT influences the peak of the error; with fixed window size, the error max increases as the interval decreases. Finally, in the FFT, it is necessary to use a larger window size when increasing the sampling frequency, corresponding to reducing the time interval to reduce the error and not increase its peak.

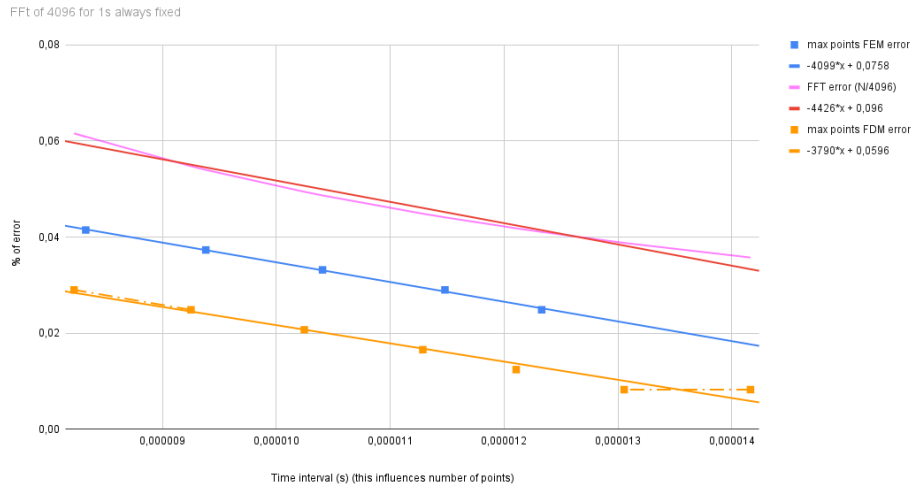


Figure 17: Fundamental error FFT frequency interval compared to Amplitude of errors when changing the time step

Moreover, comparing in a broader perspective, figure 18, 19 and 20 show the error with its mean. The error margin of the FFT is still visible. In the plot, both FEM and FDM follow the same error variations at the same rate. However, FDM converges for bigger time intervals than FEM (as explained in part 4.2), and the error max is smaller for FDM for the fundamental and second harmonic.

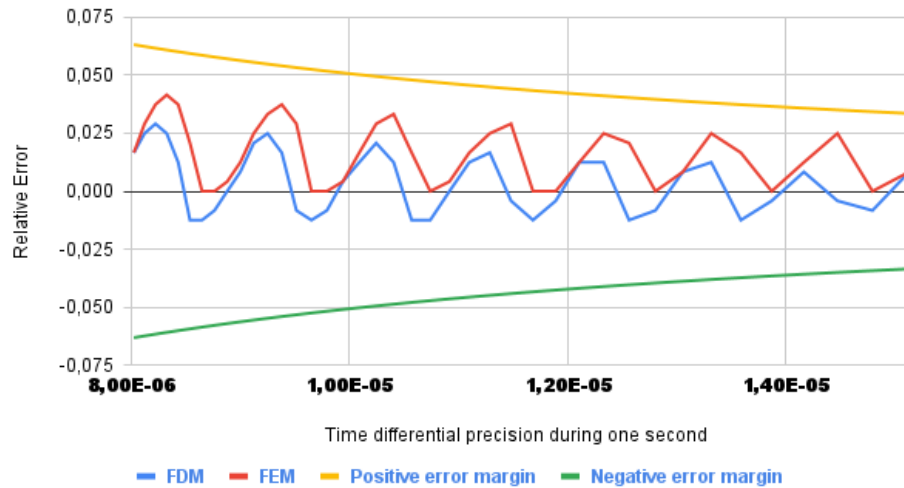


Figure 18: Error of the fundamental frequency compared to analytical when changing the time step

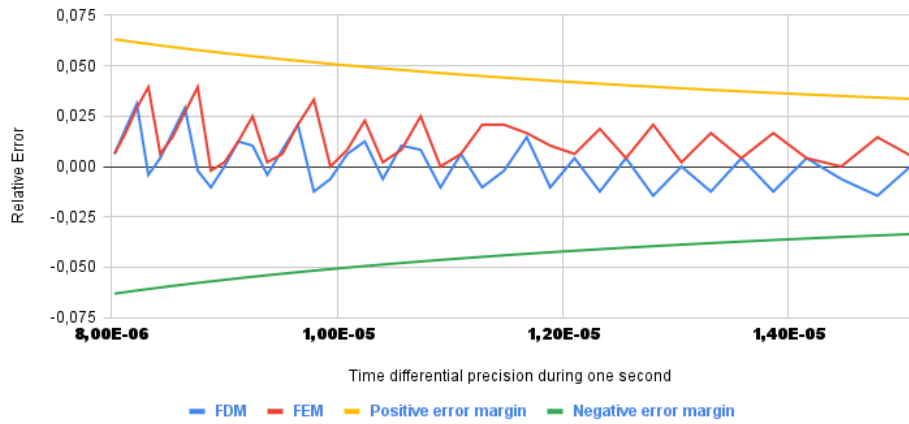


Figure 19: Error of the second harmonic compared to analytical when changing the time step

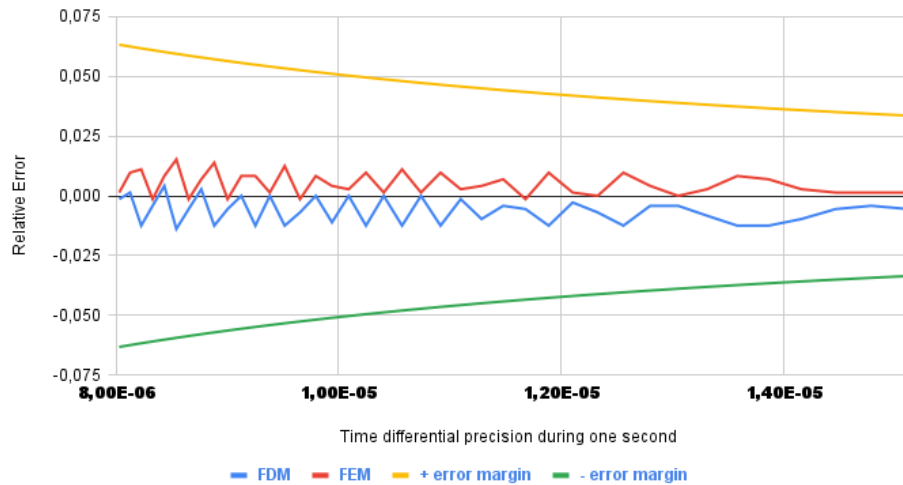


Figure 20: Error of the third harmonic compared to analytical when changing the time step

4.3 Variation in the number of nodes

When doing FEM or FDM, you have several nodes you can create; for FEM, you can have a minimum of 1 element (2 nodes), and for FDM, you can also have at least 2 nodes. However, one element or two nodes would not create any sound in our case since both extreme nodes are fixed. Thus, we can cut the guitar string

from three nodes to as many as we wish. Here, we see how the number of nodes affects the error. The fixed parameters are:

- **Tension:** 42.86 N
- **Length:** 0.655 m
- **time Interval:** 1.0×10^{-5} s
- **Position of plucked:** 0.18m
- **Number of points in time:** 1.0×10^5
- **time of wave simulated:** 1.0 s
- **Amplitude:** 3.0×10^{-4} m
- **Linear mass density:** 4.30×10^{-4} kg m⁻¹

The guitar string is plucked, and all the harmonics are present.

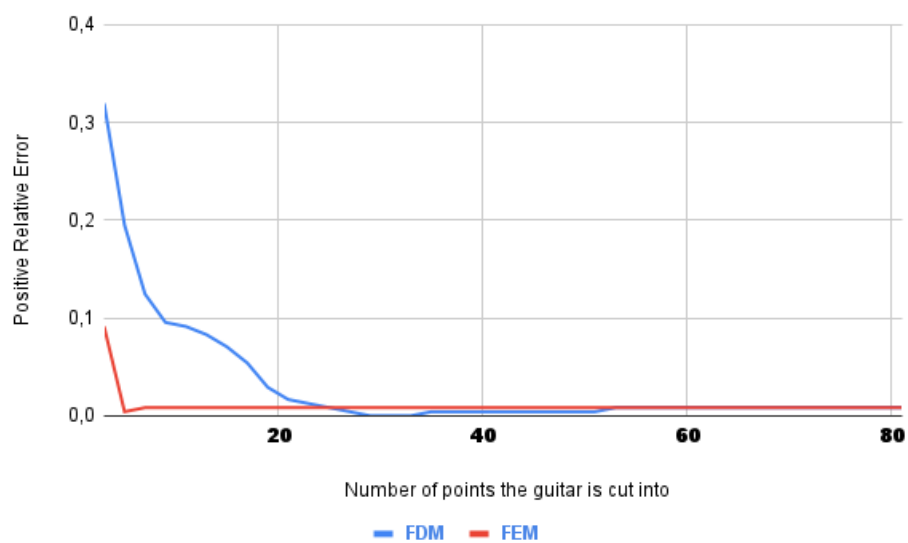


Figure 21: Positive relative error of the fundamental frequency compared to analytical according to the number of nodes

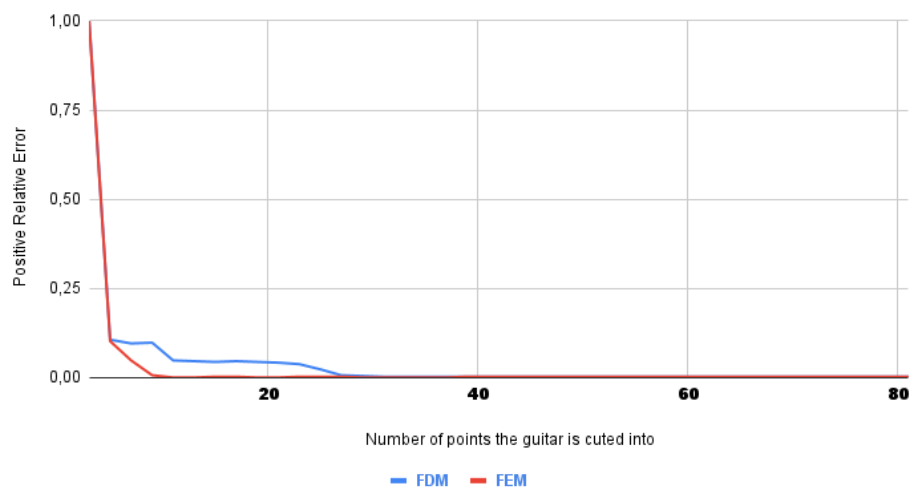


Figure 22: Positive relative error of the second harmonic compared to analytical according to the number of nodes

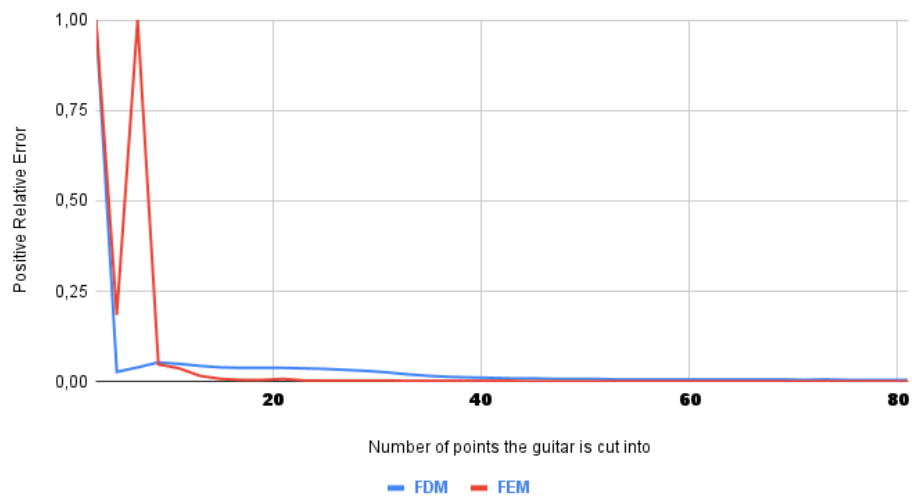


Figure 23: Positive relative error of the third harmonic compared to analytical according to the number of nodes

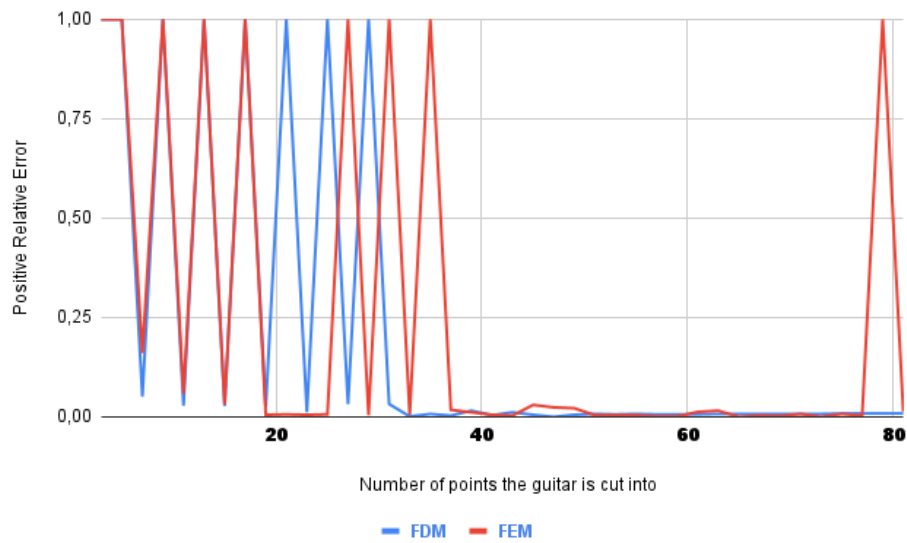


Figure 24: Positive relative error of the fourth harmonic compared to analytical according to the number of nodes

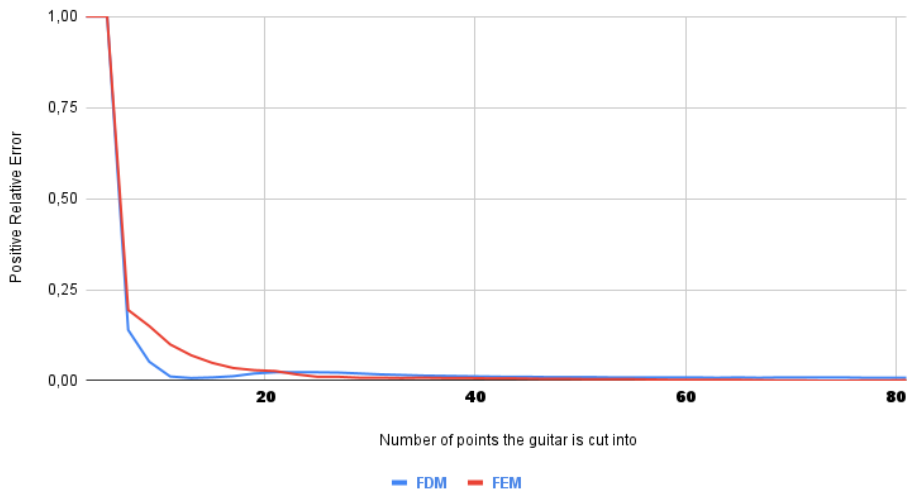


Figure 25: Positive relative error of the fifth harmonic compared to analytical according to the number of nodes

Figures 21, 22, 23, 24 and 25, show the relative error of the measured frequency according to the number of nodes we cut the string into. When the error is equal to one (Fig.24), no frequency is detected. The fundamental FEM converges faster

than FDM for the fundamental (Fig.21) and the second harmonic (Fig.22). However, after 40 nodes, the error is equivalent, and they both converge to the same error for most harmonics (Fig.21, Fig.22, Fig.23 and Fig.25). The fourth harmonic (Fig.24) was hard to detect because it had a small amplitude, but some convergence was still detectable after 40 nodes were simulated. Finally, FDM converges faster in terms of the number of nodes representing an object than FEM.

4.4 Analyzing the effects of modified parameters on the time of simulation

Two simulation codes, based on the FEM and the FDM, were developed in Python. Both codes were executed on the same computer, and the simulation times were recorded for each run corresponding to the experimental cases previously described. It should be emphasized that these performance results (Fig.26, Fig.27, Fig.28) are inherently dependent on the specific computing environment as well as the implementation details of the Python code. Consequently, the findings presented here should not be considered universally representative but rather indicative within the context of the conditions under which the analysis was conducted.

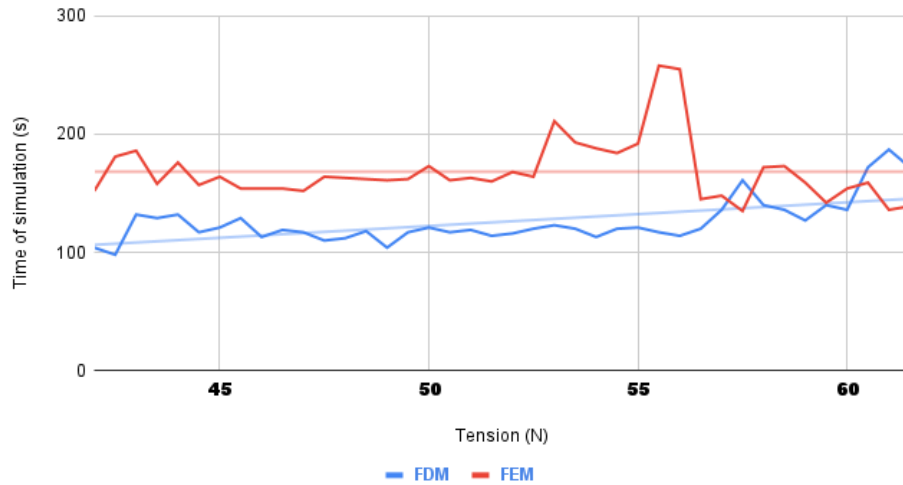


Figure 26: Time of FEM and FDM simulations of a string according to tension



Figure 27: Time of FEM and FDM simulations according to the time interval chosen in the discretizations

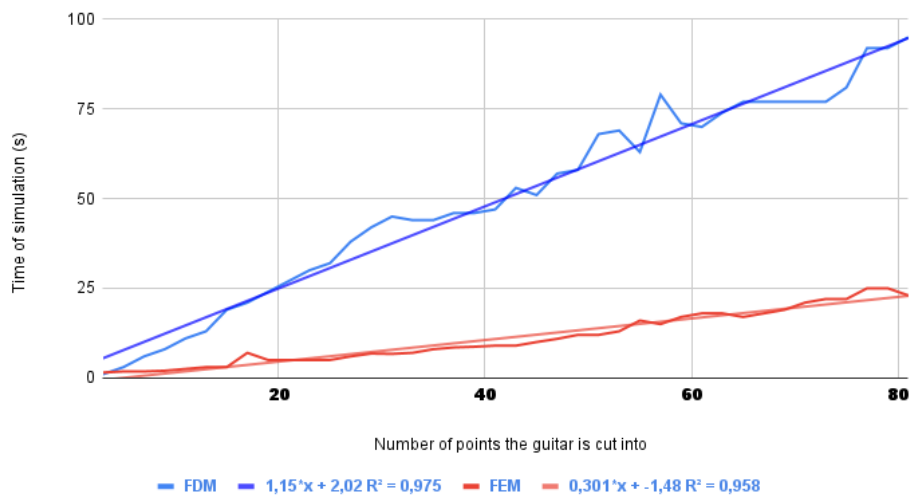


Figure 28: Time of FEM and FDM simulations of a string according to the number of cuts

Figure 26 underlines how an evolution in the tension affects the simulation time. The FEM curve is approximately following a horizontal line. Thus, the tension (Fig.26), so the frequency, does not influence the time. However, we can see that the FDM line has a small positive slope compared to FEM. However, FDM

is still faster than FEM in these conditions. Following this trend, FDM would be slower than FEM after a certain frequency.

Figure 27 illustrates the influence of a change of the time interval on the time of a simulation. Nevertheless, the smaller the time interval the slower the simulation since it means there are more data to calculate. FDM, compared to FEM, is quicker for the given time interval.

Figure 28 illustrates the impact of varying the number of nodes composing the string on simulation time for both the FEM and the FDM. As the number of nodes increases, FEM demonstrates significantly shorter simulation times compared to FDM. This discrepancy can be attributed to differences in code implementation: the FDM approach relies on an explicit Python loop that iterates over all nodes, while the FEM implementation utilizes matrix operations applied directly to the entire set of elements. As a result, FEM is expected to offer greater computational efficiency. It is important to note that this analysis reflects the specific coding strategies employed and may vary under different implementations.

5 To compare to reality: guitar with damping and without the modulus of elasticity

The D'Alembert equation for transverse vibrations of a string with fluid damping is given by [13] [10]:

$$\mu \frac{\partial^2 u}{\partial t^2} - T \frac{\partial^2 u}{\partial x^2} + \sigma \frac{\partial u}{\partial t} = 0, \quad (48)$$

where:

- $u(x, t)$: Transverse displacement of the string at position x and time t
- $c = \sqrt{\frac{T}{\mu}}$: Wave velocity in m s^{-1} , dependent on the tension T in N and the linear mass density μ in kg m^{-1}
- σ : Fluid damping coefficient kg s^{-1}

This equation includes a damping term $\sigma \frac{\partial u}{\partial t}$, which represents the resistance due to friction. We discretize the string in space and time:

- The spatial domain $[0, L]$ is divided into N nodes with spacing $\Delta x = \frac{L}{N}$.
- The temporal domain $[0, \tau]$ is divided into K points with spacing $\Delta t = \frac{\tau}{K}$.

Using finite differences, we approximate:

$$\frac{\partial^2 u}{\partial x^2} \approx \frac{u_{n+1,k} - 2u_{n,k} + u_{n-1,k}}{\Delta x^2}, \quad (49)$$

$$\frac{\partial^2 u}{\partial t^2} \approx \frac{u_{n,k+1} - 2u_{n,k} + u_{n,k-1}}{\Delta t^2}, \quad (50)$$

$$\frac{\partial u}{\partial t} \approx \frac{u_{n,k+1} - u_{n,k-1}}{2\Delta t}. \quad (51)$$

Substituting these into the D'Alembert equation yields:

$$0 = \mu \frac{u_{n,k+1} - 2u_{n,k} + u_{n,k-1}}{\Delta t^2} - T \frac{u_{n+1,k} - 2u_{n,k} + u_{n-1,k}}{\Delta x^2} + \sigma \frac{u_{n,k+1} - u_{n,k-1}}{2\Delta t} \quad (52)$$

Rearranging terms, and starting with the equation:

$$\begin{aligned} u_{n,k+1} \left(\mu \frac{1}{\Delta t^2} + \frac{\sigma}{2\Delta t} \right) &= \frac{T}{\Delta x^2} (u_{n+1,k} + u_{n-1,k}) \\ &+ \left(2\mu \frac{1}{\Delta t^2} - 2\frac{T}{\Delta x^2} \right) u_{n,k} \\ &+ u_{n,k-1} \left(-\mu \frac{1}{\Delta t^2} + \frac{\sigma}{2\Delta t} \right) \end{aligned} \quad (53)$$

To isolate $u_{n,k+1}$, divide through by $\mu \frac{1}{\Delta t^2} + \frac{\sigma}{2\Delta t}$:

$$u_{n,k+1} = \frac{\frac{T}{\Delta x^2} (u_{n+1,k} + u_{n-1,k}) + \left(2\mu \frac{1}{\Delta t^2} - 2\frac{T}{\Delta x^2} \right) u_{n,k} + u_{n,k-1} \left(-\mu \frac{1}{\Delta t^2} + \frac{\sigma}{2\Delta t} \right)}{\mu \frac{1}{\Delta t^2} + \frac{\sigma}{2\Delta t}} \quad (54)$$

We can define the coefficients to have a easier notation:

$$\gamma = \frac{T^2 \Delta t^2}{\Delta x^2}, \quad (55)$$

$$\alpha = \mu + \frac{\sigma \Delta t}{2}, \quad (56)$$

$$\theta = -\mu + \frac{\sigma \Delta t}{2}. \quad (57)$$

The equation becomes:

$$u_{n,k+1} = \frac{\gamma \frac{1}{\Delta t^2} (u_{n+1,k} + u_{n-1,k}) + \left(2\frac{\mu}{\Delta t^2} - 2\gamma \frac{1}{\Delta t^2} \right) u_{n,k} + u_{n,k-1} \left(\theta \frac{1}{\Delta t^2} \right)}{\alpha \frac{1}{\Delta t^2}}. \quad (58)$$

Thus, the final equation is:

$$u_{n,k+1} = \frac{\gamma}{\alpha} (u_{n-1,k} + u_{n+1,k}) + \frac{2(\mu - \gamma)}{\alpha} u_{n,k} + \frac{\theta}{\alpha} u_{n,k-1} \quad (59)$$

The boundary conditions are the same as in the case without sampling seen in equation 34. The initial conditions are also the same as previously in equations 31 and 32.

5.1 Comparison to Reality

This study compares our FDM simulation results to real-world conditions (Fig.29). The following physical parameters characterize the experimental setup, which we will use as the basis for our simulations:

- Fundamental frequency: $f = 246.9\text{Hz}$ (corresponding to the musical note B3), determined using a FFT with a window size of 4096.
- String dimensions:
 - Length: $L = 65.5\text{ cm} = 0.655\text{m}$
 - Diameter: $d = 0.69\text{ mm} = 6.9 \times 10^{-4}\text{m}$
- Linear mass density: $\mu = 1150 \times \left(\frac{d}{2}\right)^2 \pi = 43 \times 10^{-5}\text{kg m}^{-1} = 4.30 \times 10^{-4}\text{kg m}^{-1}$
- Playing position: The string is plucked at 0.180m .
- Theoretical tension: $T = 45.02\text{ N}$

These parameters will be the reference for validating our numerical simulations against experimental data (Fig.29) [8] [9].



Figure 29: Real guitar of string length 0.655 m played on the fifth string at 0.18 m

The simulation was performed under the following conditions:

- Number of nodes: 80 nodes
- Playing position: 22nd point, 0.18 m
- Initial Amplitude: 3.0×10^{-4} m
- Simulation duration: 3 s
- Time step: $\Delta t = 9.65 \times 10^{-6}$ s
- Damping $\sigma = 0.0013 \text{ kg s}^{-1}$ (Chosen to match the time of damping of the real guitar)

To determine the appropriate values for our simulation for string time step (Δt) in the simulation, we analyzed the error introduced by the different parameters:

- At $T = 45.02 \text{ N}$, the frequency error is +2% as shown in Figure 11.
- At $\Delta t = 9.653 \times 10^{-6} \text{ s}$, the frequency error is -1.2% as shown in Figure 18.

Thus, knowing the error generated by FDM at that tension, a time differential is chosen to balance the error created by the FDM in the numerical simulation (Fig.33).

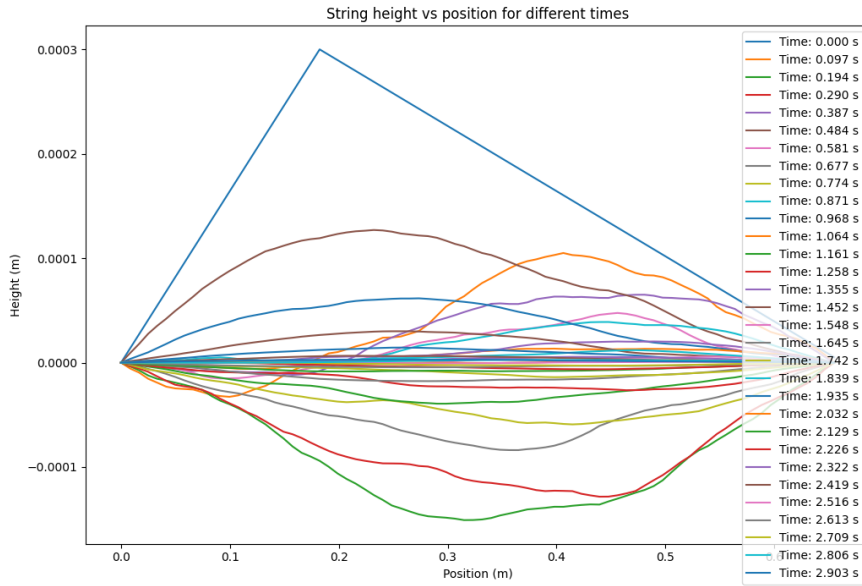


Figure 30: Simulation result of this particular string

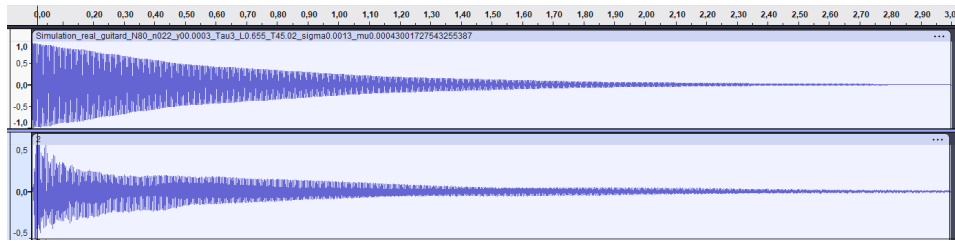


Figure 31: Damping looked at on *Audacity*, the upper waveform is the simulation and the bottom one is the real signal

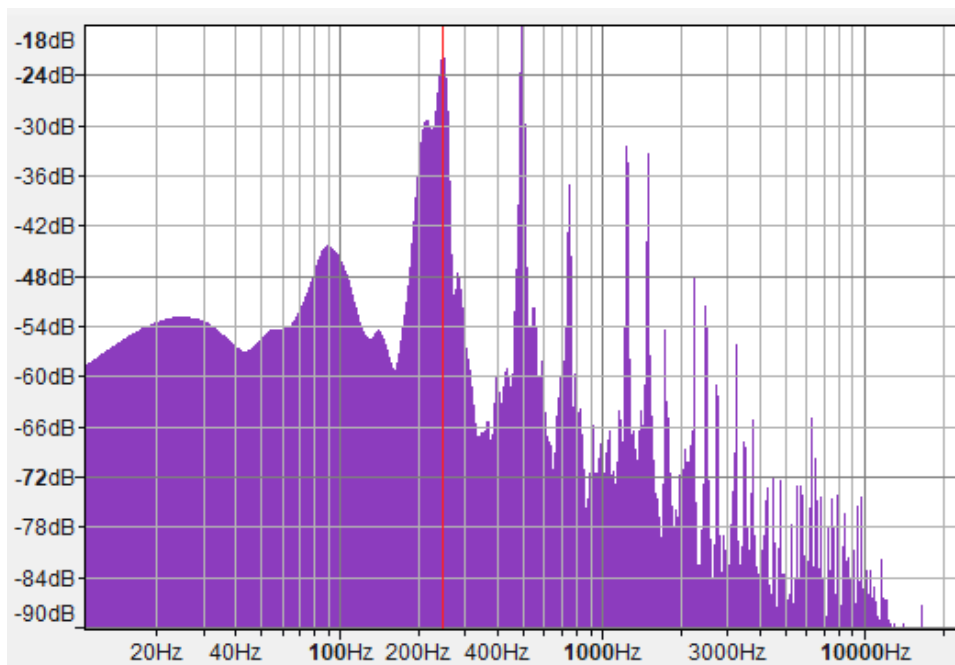


Figure 32: FFT through *Audacity* of the real signal, peak at 252 Hz

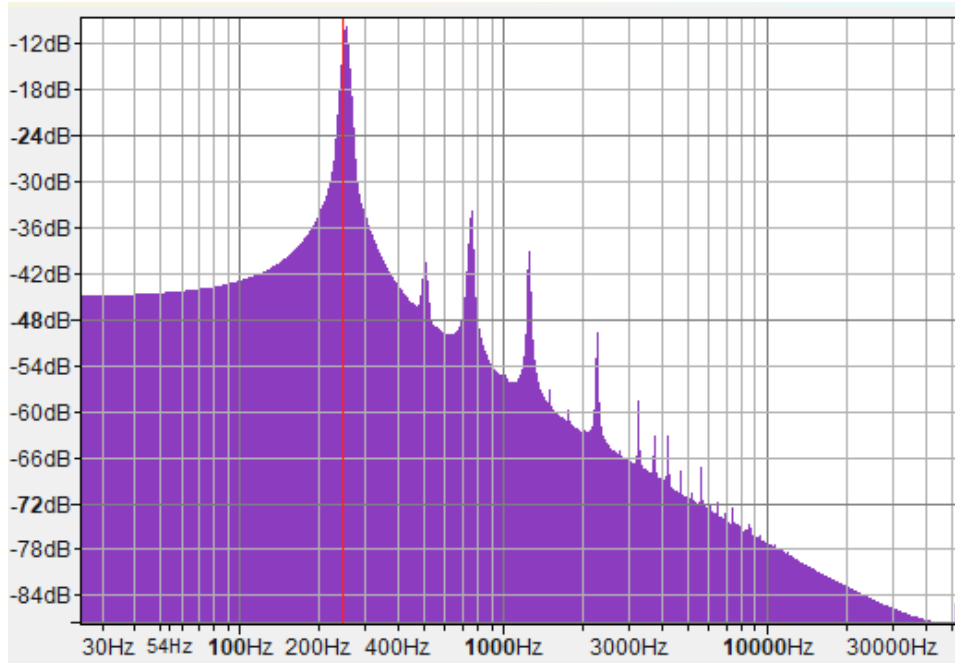


Figure 33: FFT through *Audacity* on the simulated waveform, peak at 244 Hz

Figure 30 shows the string simulated in time. Figure 31 shows the real and simulated signal damping. And finally, figure 32, and figure 33 show the FFT on both the real and the simulated waveform. First, the damping is well represented, as shown in figure 31. Then, in the frequency domain, the simulated signal (Fig.33) closely matches the real signal (Fig.32), demonstrating the accuracy of the numerical model. However, it is essential to note that perceptually, the simulated sound still exhibits noticeable differences from the real instrument. Specifically, the simulation accurately represents the vibrating string (Fig.30). The same frequencies, are seen through the simulation and the experimental data. Finally, even if the spectrum has the same frequencies the simulation does not account for the reflections from surrounding surfaces or the effects of the guitar body, which significantly contribute to the instrument's characteristic timbre. The timbre is not represented in the simulation. However, if the goal is solely to identify the string type, the simulation can still achieve this.

5.2 Important note on the simulation

When performing an FFT, it is crucial to clearly define the sampling rate and the sampling window, as errors can arise from these parameters. The accuracy of the FFT is inherently limited by the resolution of the discrete data and the chosen windowing function (section 4). Moreover, achieving higher numerical precision in the simulation does not necessarily translate to increased accuracy in practi-

cal measurements. In reality, experimental data acquisition is subject to various sources of uncertainty, including instrument limitations and environmental noise. Additionally, the FFT introduces a loss of precision due to spectral leakage and finite resolution. Therefore, while an exact numerical approach may be beneficial for theoretical analysis, it is essential to acknowledge the inherent limitations of real-world measurements.

6 The cycling bell with damping and modulus of elasticity

6.1 Model

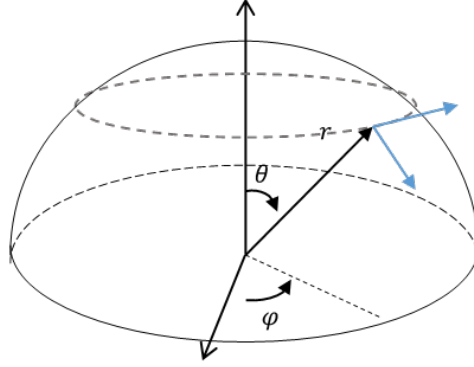


Figure 34: The model of cycling bell

The bell is modeled as the shell of a hemisphere with radius R , positioned relative to its center in spherical coordinates (r, θ, φ) as shown in the figure. We have a fixed point at the top. Assuming the shell of the bell is very thin (thickness $h \ll R$), for every point on the shell, it is possible to write: $r \approx R$.

6.2 Theory

As mentioned previously, we did not consider the flexion in the d'Alembert equation. It was a negligible term. However, that term is helpful for the cycling bell as it is mainly made of metal [14]. The fundamental equation of dynamics with damping with elasticity is expressed as follows:

$$D\nabla^4 u + \sigma \frac{\partial u}{\partial t} + \rho h \frac{\partial^2 u}{\partial t^2} = 0 \quad (60)$$

where:

- $D = \frac{Eh^3}{12(1-\nu^2)}$: the bending rigidity in N m

- h : thickness in m
- ρ : density in kg m^{-3}
- E : Young's modulus in N m^{-2}
- ν : Poisson's ratio
- ∇^4 : the biharmonic operator (double Laplacian)
- σ : damping in $\text{N s}^{-1} \text{m}^{-2}$

6.3 Analytical resolution

We seek solutions of the form:

$$u(\theta, \varphi, t) = u(\theta, \varphi)e^{i\omega t} \quad (61)$$

Substituting into the equation gives:

$$\nabla^4 u - \left(\omega^2 \frac{\rho h}{D} - i\omega \frac{\sigma}{D} \right) u(\theta, \varphi) = 0 \quad (62)$$

Or equivalently:

$$(\nabla^4 - k_\omega^4)u(\theta, \varphi) = 0 \quad (63)$$

where:

$$k_\omega^4 = \omega^2 \frac{\rho h}{D} - i\omega \frac{\sigma}{D} \quad (64)$$

We can write:

$$(\nabla^4 - k_\omega^4) = (\nabla^2 + k_\omega^2)(\nabla^2 - k_\omega^2) \quad (65)$$

The solution can be split into two components:

$$u(\theta, \varphi) = u_+(\theta, \varphi) + u_-(\theta, \varphi) \quad (66)$$

With:

$$(\nabla^2 + k_\omega^2)u_+(\theta, \varphi) = 0 \quad (67)$$

$$(\nabla^2 - k_\omega^2)u_-(\theta, \varphi) = 0 \quad (68)$$

In the spherical coordinate system, the Laplacian is expressed as:

$$\nabla^2 u = \Delta u = \frac{1}{r^2} \frac{\partial}{\partial r} \left(r^2 \frac{\partial u}{\partial r} \right) + \frac{1}{r^2 \sin \theta} \frac{\partial}{\partial \theta} \left(\sin \theta \frac{\partial u}{\partial \theta} \right) + \frac{1}{r^2 (\sin \theta)^2} \frac{\partial^2 u}{\partial \varphi^2} \quad (69)$$

Since the shell is very thin, we can assume that r is constant and equal to R . Thus, the waves described by equation (22) propagate mainly along the bell's surface, with variations in θ and φ : $u = u(\theta, \varphi, t)$

$$\frac{1}{r^2 \sin \theta} \frac{\partial}{\partial \theta} \left(\sin \theta \frac{\partial u}{\partial \theta} \right) = \frac{1}{r^2 \sin \theta} \left(\cos \theta \frac{\partial u}{\partial \theta} + \sin \theta \frac{\partial^2 u}{\partial \theta^2} \right) \quad (70)$$

$$= \frac{1}{r^2} \left(\cot \theta \frac{\partial u}{\partial \theta} + \frac{\partial^2 u}{\partial \theta^2} \right) \quad (71)$$

Having $r = R = \text{constant}$, the Laplacian becomes:

$$\nabla^2 u = \frac{1}{R^2} \left(\cot \theta \frac{\partial u}{\partial \theta} + \frac{\partial^2 u}{\partial \theta^2} + \frac{1}{(\sin \theta)^2} \frac{\partial^2 u}{\partial \varphi^2} \right) \quad (72)$$

We seek the solutions by separating variables:

$$u(\theta, \varphi) = w(\theta) \cdot h(\varphi) \quad (73)$$

We write:

$$(\nabla^2 + k_\omega^2) w(\theta) \cdot h(\varphi) = 0 \quad (74)$$

Taking:

$$\left(\sin \theta \cos \theta \frac{\partial w}{\partial \theta} + (\sin \theta)^2 \frac{\partial^2 w}{\partial \theta^2} \right) \frac{1}{w(\theta)} + (\sin \theta)^2 R^2 k_\omega^2 = -\frac{\partial^2 h}{\partial \varphi^2} \frac{1}{h(\varphi)} \quad (75)$$

In the equation 75, the term on the right-hand side depends only on φ , and the term on the left-hand side depends only on θ . This implies that each term must equal a constant λ . We thus obtain:

$$\begin{cases} \left(\sin \theta \cos \theta \frac{\partial w_+}{\partial \theta} + (\sin \theta)^2 \frac{\partial^2 w_+}{\partial \theta^2} \right) \frac{1}{w_+} + (\sin \theta)^2 R^2 k_\omega^2 = \lambda \\ -\partial^2 \frac{h_+}{\partial \varphi^2} \frac{1}{h_+} = \lambda \end{cases} \quad (76)$$

We can rewrite it:

$$\begin{cases} (\sin \theta)^2 \frac{\partial^2 w_+}{\partial \theta^2} + \sin \theta \cos \theta \frac{\partial w_+}{\partial \theta} + \left((\sin \theta)^2 R^2 k_\omega^2 - \lambda \right) \cdot w_+ = 0 \\ \lambda h_+ + \partial^2 \frac{h_+}{\partial \varphi^2} = 0 \end{cases} \quad (77)$$

Or:

$$\begin{cases} \partial^2 \frac{w_+}{\partial \theta^2} + \cot \theta \frac{\partial w_+}{\partial \theta} + \left(R^2 k_\omega^2 - \frac{\lambda}{(\sin \theta)^2} \right) \cdot w_+(\theta) = 0 \\ \lambda h_+ + \partial^2 \frac{h_+}{\partial \varphi^2} = 0 \end{cases} \quad (78)$$

We would also get:

$$\begin{cases} \partial^2 \frac{w_-}{\partial \theta^2} + \cot \theta \frac{\partial w_-}{\partial \theta} + \left(-R^2 k_\omega^2 - \frac{\lambda}{(\sin \theta)^2} \right) \cdot w_-(\theta) = 0 \\ \lambda h_- + \partial^2 \frac{h_-}{\partial \varphi^2} = 0 \end{cases} \quad (79)$$

Let us specify the ranges of the variables in question:

$$\varphi \in [0; \pi 2[(+2k\pi) \quad (80)$$

$$\theta \in \left[0; \frac{\pi}{2}\right] \quad (81)$$

The solutions to the second equality in 78 and 79 are exponential functions with:

$$h_+(\varphi) = h_{0+} e^{\pm i\sqrt{\lambda}\varphi} \quad (82)$$

$$h_-(\varphi) = h_{0-} e^{\pm i\sqrt{\lambda}\varphi} \quad (83)$$

From a physical standpoint, the functions h_+ and h_- are 2π periodic. This means that $\sqrt{\lambda}$ must be an integer, thus, $\sqrt{\lambda} = m \in \mathbf{N}$. By setting: $\Omega^2 = R^2 k_\omega^2$ we can now solve the equation 84 below (for both w_+ and w_-).

$$\frac{\partial^2 w(\theta)}{\partial \theta^2} + \cot \theta \frac{\partial w(\theta)}{\partial \theta} + \left(\pm \Omega^2 - \frac{\lambda}{(\sin \theta)^2} \right) w(\theta) = 0 \quad (84)$$

We can change the variable to find solutions to the equation 84. By setting $x = \cos \theta$ with θ respecting its interval 81, we can show that the equation can be rewritten with $x \in [0; 1]$ as:

$$(1 - x^2) \frac{\partial^2 w(x)}{\partial x^2} - 2x \frac{\partial w(x)}{\partial x} + \left(\pm \Omega^2 - \frac{m^2}{1 - x^2} \right) w(x) = 0 \quad (85)$$

The equation 85 is called the "associated Legendre equation". We can rewrite the equation 85 by seeking a solution, with m an integer, in the form of:

$$w(x) = (1 - x^2)^{\frac{m}{2}} p(x) \quad (86)$$

In this case, the equation is rewritten with $x \in [0; 1]$ as:

$$(1 - x^2) \frac{\partial^2 p(x)}{\partial x^2} - 2x(m+1) \frac{\partial p(x)}{\partial x} + (\pm \Omega^2 - m(m+1)) p(x) = 0 \quad (87)$$

We seek the solution $p(x)$ in the form of a convergent power series on $x \in [0; 1]$:

$$p(x) = \sum_{n \geq 0} a_n x^n \quad (88)$$

$$\frac{\partial p(x)}{\partial x} = \sum_{n \geq 0} a_{n+1} (n+1) x^n \quad (89)$$

$$\frac{\partial^2 p(x)}{\partial x^2} = \sum_{n \geq 0} a_{n+2} (n+2)(n+1) x^n \quad (90)$$

Substituting equation 90, 88, and 89 into equation 87, we get:

$$0 = \sum_{n \geq 0} a_{n+2} (n+2) (n+1) x^n - \sum_{n \geq 0} a_{n+2} (n+2) (n+1) x^{n+2} - \sum_{n \geq 0} a_{n+1} (n+1) (m+1) x^{n+1} + \sum_{n \geq 0} a_n (\pm \Omega^2 - m(m+1)) x^n \quad (91)$$

This simplifies to:

$$0 = \sum_{n \geq 0} a_{n+2} (n+2) (n+1) x^n - \sum_{n \geq 2} a_n (n) (n-1) x^n - \sum_{n \geq 1} a_n (n) (m+1) x^n + \sum_{n \geq 0} a_n (\pm \Omega^2 - m(m+1)) x^n \quad (92)$$

This gives us that for $n \geq 0$, the coefficients of x^n are:

$$a_{n+2} (n+2) (n+1) - a_n \cdot (n(n+m) + m(m+1) \mp \Omega^2) \quad (93)$$

By substituting the derivatives of $p(x)$ into the equation 87 and by thinking through recurrence, we can prove that to have all the coefficients at zero, we must have:

$$a_{n+2} = a_n \frac{n^2 + n(2m+1) + m(m+1) \mp \Omega^2}{(n+2)(n+1)} \quad (94)$$

6.3.1 First case

To avoid divergence of the series (even when $x = 1$), it is required that all the coefficients are zero from a certain rank onward. This means that the suitable values of Ω must satisfy for some n_0 :

$$\Omega = \pm \sqrt{n_0^2 + n_0(2m+1) + m(m+1)} \quad (95)$$

We can also see that there is a complex solution:

$$\Omega = \pm i \sqrt{n_0^2 + n_0(2m+1) + m(m+1)} \quad (96)$$

It should be noted that the recurrence relation between the coefficients of the power series is of order 2. This means that for all the coefficients to be zero from a certain rank onward, all coefficients must have the same parity as n_0 . By setting $k = n_0 + m$, in equation 96 and equation 95, Ω can be either:

$$\Omega = \pm \sqrt{k(k+1)} \quad (97)$$

$$\Omega = \pm i \sqrt{k(k+1)} \quad (98)$$

6.3.2 Second case

We seek convergence for all $x \leq 1$. D'Alembert's rule says convergence for all $x < 1$. But at $x = 1$ we need to use: Raabe-Duhamel test for $x = 1$

$$\left| \frac{a_2(k+1)}{a_2k} \right| = 1 - \frac{2-m}{k} + o\left(\frac{1}{k}\right) \quad (99)$$

Thus, to avoid divergence, we will have convergence if $m = 0$; however, when that is the case, $w(x) = p(x)$ (equation 86). With the condition at the limit, the top is fixed $p(1) = 0$. Figure 35 shows the value of $p(1)$ according to different Ω for $m = 0$. It highlights all the Ω where $p(1) = 0$. It has been verified that all the values of Ω where the plot passed by zero are the cases where there is a finite sum; thus, Ω follows the first case seen before where all the terms are null after a rank n_0 . In conclusion, the only case where that condition is verified is when all the terms are null after a certain rank.

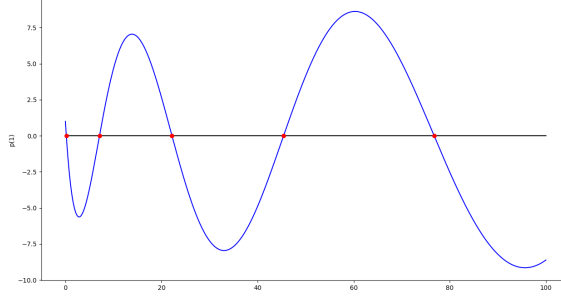


Figure 35: $p(1)$ according to Ω in $m = 0$

Taking the equations 97 and 98, the frequency solutions are either way, because $i^4 = 1$ so $\Omega^4 = (Rk_\omega)^4$:

$$\omega^2 \frac{\rho h R^4}{D} - i\omega \frac{\sigma R^4}{D} - k^2 (k+1)^2 = 0 \quad (100)$$

This means:

$$\omega = \left(i \frac{\sigma}{2\rho h} \pm \sqrt{-\left(\frac{\sigma}{2\rho h}\right)^2 + \frac{D}{\rho h R^4} k^2 (k+1)^2} \right) \quad (101)$$

Finally we have an attenuation term $e^{\frac{-\sigma}{2\rho h}t}$ and a harmonic term $e^{2i\pi f_k t}$ with:

$$f_k = \frac{1}{2\pi} \sqrt{-\left(\frac{\sigma}{2\rho h}\right)^2 + \frac{D}{\rho h R^4} k^2 (k+1)^2} \quad (102)$$

We calculate two characteristic coefficients of the material and the geometry of the shell:

- The attenuation coefficient $\gamma = \frac{\sigma}{2\rho h}$
- The stiffness coefficient $\alpha = \frac{D}{\rho h R^4}$

Next, we choose two integers m_0 and n_0 that determine the vibration mode. For each pair (m_0, n_0) chosen, we determine the coefficient $k_0 = n_0 + m_0$ which allows the calculation of the vibration frequency:

$$f_{k_0} = \frac{1}{2\pi} \sqrt{-\gamma^2 + \alpha k_0^2 (k_0 + 1)^2} \quad (103)$$

The vibration mode is written as follows:

$$u(\theta, \varphi, t) = u_{(m_0, n_0)}(\theta, \varphi) e^{\frac{-\sigma}{2\rho h} t} e^{2i\pi f_{k_0} t} \quad (104)$$

The elongations are written for the chosen m_0 and n_0 as:

$$u_{(m_0, n_0)} = w_{(m_0, n_0)}(\theta) \cdot h_{m_0}(\varphi) \quad (105)$$

With:

$$h_{m_0}(\varphi) = h_0 e^{im_0 \varphi} \quad (106)$$

$$w_{(m_0, n_0)}(\theta) = (\sin \theta)^{\frac{m_0}{2}} \cdot p_{n_0}(\cos \theta) \quad (107)$$

$$p_{n_0}(x) = a_0 + a_1 x^1 + a_2 x^2 + \dots a_{n_0} x^{n_0} \quad (108)$$

To calculate the coefficients of $p_{n_0}(x)$, we first calculate:

$$\Omega_0 = \sqrt{k_0 (k_0 + 1)} \quad (109)$$

If n_0 is even, all coefficients a_n with odd n are zero. If n_0 is odd, all coefficients a_n with even n are zero. Next, for n varying from 0 to n_0 , we calculate:

$$a_{n+2} = a_n \frac{n^2 + n(2m_0 + 1) + m_0(m_0 + 1) - \Omega_0^2}{(n+2)(n+1)} \quad (110)$$

We start with a_0 for the even coefficients and with a_1 for the odd coefficients. Due to the recurrence relation between the coefficients, we notice that all even coefficients are proportional to a_0 and all odd coefficients are proportional to a_1 . Thus, the coefficients can be adjusted based on the boundary conditions. Parameters for numerical application:

$$\sigma = 10 \text{ kg m}^{-2}$$

$$R = 0.05 \text{ m}$$

$$h = 0.001 \text{ m}$$

Table 1: Material properties: density, Young's modulus E , and Poisson's ratio ν

Material	Density (kg m ⁻³)	E (GPa)	ν
Steel	7850	210	0.24–0.30
Aluminum	2700	62	0.24–0.33
Copper	8920	128	0.33
Brass	8470	80–100	0.37

6.3.3 Numerical test

Using Matlab, we could compute the sound the bell would do in theory by superposing all the harmonics with the proper amplitudes (Fig.36).



Figure 36: Time signal of the audio file generated viewed on *Audacity*

6.4 Comparing analytical to reality

Important information Before comparing our mathematical model to reality, we can see the numerical frequency is related to the damping term:

$$f_k = \frac{1}{2\pi} \sqrt{-\left(\frac{\sigma}{2\rho h}\right)^2 + \frac{D}{\rho h R^4} k^2 (k+1)^2} \quad (111)$$

In the equation 111 showing the frequencies of a bell there is a damping term. Yet there are no known explicit damping terms for the cycling bell. Figure 37 shows the effect of the term damping on the frequency in the logarithmic scale. The light orange and light blue lines coincide with the orange and blue lines. They represent linear regression, and both slop terms are negligible compared to the initialization terms. In consequence, the damping term has little effect on the frequency.

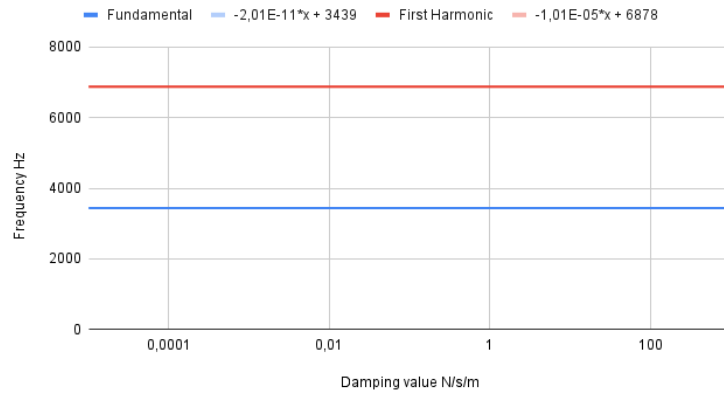


Figure 37: Damping effect on the fundamental frequency and first harmonic in logarithmic scale with linear regression

Fixing the damping at $10 \text{ kg m}^2 \text{ s}^{-1}$ allows us to match the damping speed of a real-life cycling bell that was tested knowing that it will not effect the analysis comparing the experimental and the calculated frequencies. Figure 38 shoes a big aluminum bell used in the experiment.



Figure 38: Aluminium bell

The parameters of the bell (Fig.38) are:

- Damping coefficient (fixed arbitrarily): $\sigma = 10 \text{ kg m}^{-2} \text{ s}^{-1}$
- Density: $\rho = 2.70 \times 10^3 \text{ kg m}^{-3}$
- Radius: $R = 4.00 \times 10^{-2} \text{ m}$
- Thickness: $h = 8.00 \times 10^{-4} \text{ m}$
- Young's modulus: $E = 6.20 \times 10^{10} \text{ N} \cdot \text{m}^{-2}$
- Poisson's ratio: $\nu = 3.00 \times 10^{-1}$

Table 2: Comparison of fundamental and harmonic frequencies (f_n) between Matlab simulations, experimental data, and mathematical calculations for aluminum.

Cycling Bell	f_1 (Hz)	f_2 (Hz)	f_3 (Hz)	f_4 (Hz)
Model (Matlab)	689	1385	2305	3462
Reality	688	1742	3080	3502
Analytical analysis	692	1385	2308	3462

Table 2 shows the different frequencies calculated through different methods. The analytical analysis of the physical properties of the bell is one method. Another one is measuring frequencies through *Audacity* of the experimental data. Finally, the frequencies were measured through *Audacity* after simulating the analytical result through Matlab (summing all the vibration modes of equation 104). The code of Matlab directly corresponds to the mathematical frequency; thus, it should give the same result for the frequencies. However, there are a few Hertz differences for the fundamental frequency and the third harmonic. Thus, the FFT applied to the Matlab simulation greatly influences the results. The window is still at 4096; we are losing information and precision. Furthermore, the same FFT is applied to experimental data. In consequence, there is the same error margin. We also find almost the same frequency result as in the analytical case.

Moreover, a few Hertz of difference is not detectable for an average human ear; this shows that the analytical result can fully be used to model the bell. The same experience was conducted with a supposedly brass bell. However, the resulting frequencies from the experimental data and the analytical resolution were very different (more than 1000 Hz difference). Knowing that the model was correct showed that the cycling bell was not made of brass. After verifying the texture and the weight, the bell did not seem to have brass. This model could be used to check the material of a given object.

6.5 Finite Difference Method

We have analyzed the theory and how it reflects reality but we have not yet tested the FDM to simulate numerically the frequency heard. We study the bell without damping. Thus, a bell obeys the plate equation (equation 112).

$$\frac{\partial^2 u}{\partial t^2} = -\frac{D}{\rho h} \nabla^4 u \quad (112)$$

where:

- ∇^4 : biharmonic operator (double Laplacian).

Boundary conditions are defined as follows:

- At the point $\theta = 0$, vibration is always zero:

$$\forall t \forall \varphi u(0, \varphi, t) = 0 \quad (113)$$

- Along the great circle $\theta = \frac{\pi}{2}$, vibration is free along θ :

$$\forall t \forall \varphi \frac{\partial u}{\partial \theta}(0, \varphi, t) = 0 \quad (114)$$

In spherical coordinates, the Laplacian is expressed as:

$$\Delta u = \frac{1}{r^2} \frac{\partial}{\partial r} \left(r^2 \frac{\partial u}{\partial r} \right) + \frac{1}{r^2 \sin \theta} \frac{\partial}{\partial \theta} \left(\sin \theta \frac{\partial u}{\partial \theta} \right) + \frac{1}{r^2 \sin^2 \theta} \frac{\partial^2 u}{\partial \varphi^2} \quad (115)$$

Since the shell is very thin, we can consider r as constant and equal to R . Thus, the waves described by equation 112 primarily propagate along the surface of the bell, with variations in θ and φ : $u = u(\theta, \varphi, t)$. The Laplacian (equation 115) becomes:

$$\frac{1}{R^2} \left(\cot \theta \frac{\partial u}{\partial \theta} + \frac{\partial^2 u}{\partial \theta^2} + \frac{1}{(\sin \theta)^2} \frac{\partial^2 u}{\partial \varphi^2} \right) \quad (116)$$

For the double Laplacian, we can factor by $\frac{1}{R^4}$ in the same way and have 9 terms:

$$\cot \theta \frac{\partial}{\partial \theta} \left(\cot \theta \frac{\partial u}{\partial \theta} \right) = \frac{-\cot \theta}{(\sin \theta)^2} \frac{\partial u}{\partial \theta} + (\cot \theta)^2 \frac{\partial^2 u}{\partial \theta^2} \quad (117)$$

$$\cot \theta \frac{\partial}{\partial \theta} \left(\frac{\partial^2 u}{\partial \theta^2} \right) = \cot \theta \frac{\partial^3 u}{\partial \theta^3} \quad (118)$$

$$\cot \theta \frac{\partial}{\partial \theta} \left(\frac{1}{(\sin \theta)^2} \frac{\partial^2 u}{\partial \varphi^2} \right) = -2 \frac{(\cot \theta)^2}{(\sin \theta)^2} \frac{\partial^2 u}{\partial \varphi^2} + \frac{\cot \theta}{(\sin \theta)^2} \frac{\partial^3 u}{\partial \theta \partial \varphi^2} \quad (119)$$

$$\frac{\partial^2}{\partial \theta^2} \left(\cot \theta \frac{\partial u}{\partial \theta} \right) = \frac{2 \cot \theta}{(\sin \theta)^2} \frac{\partial u}{\partial \theta} + 2 \frac{-1}{(\sin \theta)^2} \frac{\partial^2 u}{\partial \theta^2} + \cot \theta \frac{\partial^3 u}{\partial \theta^3} \quad (120)$$

$$\frac{\partial^2}{\partial \theta^2} \left(\frac{\partial^2 u}{\partial \theta^2} \right) = \frac{\partial^4 u}{\partial \theta^4} \quad (121)$$

$$\begin{aligned} \frac{\partial^2}{\partial \theta^2} \left(\frac{1}{(\sin \theta)^2} \frac{\partial^2 u}{\partial \varphi^2} \right) &= \left(\frac{2}{(\sin \theta)^4} + 4 \frac{(\cot \theta)^2}{(\sin \theta)^2} \right) \frac{\partial^2 u}{\partial \varphi^2} \\ &+ \frac{-4 \cot \theta}{(\sin \theta)^2} \frac{\partial^3 u}{\partial \theta \partial \varphi^2} + \frac{1}{(\sin \theta)^2} \frac{\partial^4 u}{\partial \varphi^2 \partial \theta^2} \end{aligned} \quad (122)$$

$$\frac{1}{(\sin \theta)^2} \frac{\partial^2}{\partial \varphi^2} \left(\cot \theta \frac{\partial u}{\partial \theta} \right) = \frac{\cot \theta}{(\sin \theta)^2} \frac{\partial^3 u}{\partial \theta \partial \varphi^2} \quad (123)$$

$$\frac{1}{(\sin \theta)^2} \frac{\partial^2}{\partial \varphi^2} \left(\frac{\partial^2 u}{\partial \theta^2} \right) = \frac{1}{(\sin \theta)^2} \frac{\partial^4 u}{\partial \varphi^2 \partial \theta^2} \quad (124)$$

$$\frac{1}{(\sin \theta)^2} \frac{\partial^2}{\partial \varphi^2} \left(\frac{1}{(\sin \theta)^2} \frac{\partial^2 u}{\partial \varphi^2} \right) = \frac{1}{(\sin \theta)^4} \frac{\partial^4 u}{\partial \varphi^4} \quad (125)$$

We thus obtain with equation 117 to 125 the following term:

$$\begin{aligned}
\nabla^4 u = & \frac{\partial^4 u}{\partial \theta^4} + \frac{1}{(\sin \theta)^4} \frac{\partial^4 u}{\partial \varphi^4} + \left(\frac{2}{(\sin \theta)^2} \right) \frac{\partial^4 u}{\partial \varphi^2 \partial \theta^2} - 2 \frac{\cot \theta}{(\sin \theta)^2} \frac{\partial^3 u}{\partial \theta \partial \varphi^2} \\
& + 2 \cot \theta \frac{\partial^3 u}{\partial \theta^3} + \left((\cot \theta)^2 + \frac{-2}{(\sin \theta)^2} \right) \frac{\partial^2 u}{\partial \theta^2} \\
& + \left(\frac{2}{(\sin \theta)^4} + 2 \frac{(\cot \theta)^2}{(\sin \theta)^2} \right) \frac{\partial^2 u}{\partial \varphi^2} + \frac{\cot \theta}{(\sin \theta)^2} \frac{\partial u}{\partial \theta}
\end{aligned} \tag{126}$$

This means:

$$\begin{aligned}
\nabla^4 u = & \frac{\partial^4 u}{\partial \theta^4} + \frac{\partial^4 u}{(\sin \theta \partial \varphi)^4} + 2 \frac{\partial^4 u}{(\sin \theta \partial \varphi)^2 \partial \theta^2} \\
& - 2 \cot \theta \frac{\partial^3 u}{\partial \theta (\sin \theta \partial \varphi)^2} + 2 \cot \theta \frac{\partial^3 u}{\partial \theta^3} + \left(\frac{(\cos \theta)^2 - 2}{(\sin \theta)^2} \right) \frac{\partial^2 u}{\partial \theta^2} \\
& + 2 \left(\frac{1 + (\cos \theta)^2}{(\sin \theta)^2} \right) \frac{\partial^2 u}{(\sin \theta \partial \varphi)^2} + \frac{\cot \theta}{(\sin \theta)^2} \frac{\partial u}{\partial \theta}
\end{aligned} \tag{127}$$

Even:

$$\begin{aligned}
\nabla^4 u = & \frac{\partial^4 u}{\partial \theta^4} + 2 \cot \theta \frac{\partial^3 u}{\partial \theta^3} + \left(\frac{(\cos \theta)^2 - 2}{(\sin \theta)^2} \right) \frac{\partial^2 u}{\partial \theta^2} + \frac{\cot \theta}{(\sin \theta)^2} \frac{\partial u}{\partial \theta} \\
& + \frac{\partial^4 u}{(\sin \theta \partial \varphi)^4} + 2 \left(\frac{1 + (\cos \theta)^2}{(\sin \theta)^2} \right) \frac{\partial^2 u}{(\sin \theta \partial \varphi)^2} \\
& + 2 \frac{\partial^4 u}{(\sin \theta \partial \varphi)^2 \partial \theta^2} - 2 \cot \theta \frac{\partial^3 u}{\partial \theta (\sin \theta \partial \varphi)^2}
\end{aligned} \tag{128}$$

Remember that for $u(\theta, \varphi)$:

$$\begin{aligned}
\varphi & \in [0; 2\pi[\quad (+2k\pi) \\
\theta & \in \left[0; \frac{\pi}{2} \right]
\end{aligned}$$

We can cut in small pieces and get $u(\theta, \varphi) = u(n, k)$, this approximates the terms of equation 128 :

$$\frac{\partial u}{\partial \theta}(n, k) \cong \frac{1}{2\delta\theta} (u(n+1, k) - u(n-1, k)) \quad (129)$$

$$\frac{\partial^2 u}{\partial \theta^2}(n, k) \cong \frac{1}{\delta\theta^2} (u(n+1, k) - 2u(n, k) + u(n-1, k)) \quad (130)$$

$$\frac{\partial^2 u}{\partial \varphi^2}(n, k) \cong \frac{1}{\delta\varphi^2} (u(n, k+1) - 2u(n, k) + u(n, k-1)) \quad (131)$$

$$\begin{aligned} \frac{\partial^3 u}{\partial \theta^3}(n, k) \cong & \frac{3}{8\delta\theta^3} \left(u(n+2, k) - u(n-2, k) \right. \\ & \left. - 2(u(n+1, k) - u(n-1, k)) \right) \end{aligned} \quad (132)$$

$$\begin{aligned} \frac{\partial^4 u}{\partial \theta^4}(n, k) \cong & \frac{1}{\delta\theta^4} \left(u(n+2, k) + u(n-2, k) - 4u(n+1, k) \right. \\ & \left. - 4u(n-1, k) + 6u(n, k) \right) \end{aligned} \quad (133)$$

$$\begin{aligned} \frac{\partial^4 u}{\partial \varphi^4}(n, k) \cong & \frac{1}{\delta\varphi^4} \left(u(n, k+2) + u(n, k-2) - 4u(n, k+1) \right. \\ & \left. - 4u(n, k-1) + 6u(n, k) \right) \end{aligned} \quad (134)$$

$$\begin{aligned} \left(\frac{\partial^3 u}{\partial \varphi \partial \theta^2} \right) (n, k) \cong & \frac{1}{2\delta\varphi \delta\theta^2} \left(u(n+1, k+1) - 2u(n, k+1) + u(n-1, k+1) \right. \\ & \left. - u(n+1, k-1) + 2u(n, k-1) - u(n-1, k-1) \right) \end{aligned} \quad (135)$$

$$\begin{aligned} \left(\frac{\partial^3 u}{\partial \theta \partial \varphi^2} \right) (n, k) \cong & \frac{1}{2\delta\varphi^2 \delta\theta} \left(u(n+1, k+1) - 2u(n+1, k) + u(n+1, k-1) \right. \\ & \left. - u(n-1, k+1) + 2u(n-1, k) - u(n-1, k-1) \right) \end{aligned} \quad (136)$$

$$\begin{aligned} \left(\frac{\partial^4 u}{\partial \theta^2 \partial \varphi^2} \right) (n, k) \cong & \frac{1}{\delta\varphi^2 \delta\theta^2} \left(u(n+1, k+1) - 2u(n, k+1) + u(n-1, k+1) \right. \\ & - 2(u(n+1, k) - 2u(n, k) + u(n-1, k)) \\ & \left. + u(n+1, k-1) - 2u(n, k-1) + u(n-1, k-1) \right) \end{aligned} \quad (137)$$

However, to consider the time, we need to add the damping term:

$$\rho h \frac{\partial^2 u}{\partial t^2} = -\frac{D}{R^4} \nabla^4 u - \sigma \frac{\partial u}{\partial t} \quad (138)$$

We also know that:

$$\frac{\partial^2 u}{\partial t^2}(n, k, t) \cong \frac{1}{\delta t^2} (u(n, k, t+1) - 2u(n, k, t) + u(n, k, t-1)) \quad (139)$$

So using equation 139 in 112 we get:

$$\begin{aligned} \left(\frac{\rho h}{\delta t^2} + \frac{\sigma}{2\delta t} \right) u(n, k, t+1) = & -\frac{D}{R^4} \nabla^4 u(n, k, t) \\ & + 2 \frac{\rho h}{\delta t^2} u(n, k, t) + \left(-\frac{\rho h}{\delta t^2} + \frac{\sigma}{2\delta t} \right) u(n, k, t-1) \end{aligned} \quad (140)$$

For the condition at the limits equation 113, and 114 we get:

- $\forall t \quad \forall \varphi \quad u(0, \varphi, t) = 0$
- $\forall t \quad \forall \varphi \quad \frac{\partial u}{\partial \theta}(\frac{\pi}{2}, \varphi, t) = 0$

The first implemented code in Python was the FDM for the bell for the equation 141 that does not give audible frequencies. We assumed that the mechanical vibrations in the material obey D'Alembert's wave equation. This is not an unreasonable assumption since it is known to hold for waves in fluids, i.e., longitudinal waves.

$$\Delta u - \frac{1}{c^2} \frac{\partial^2 u}{\partial t^2} = 0 \quad (141)$$

The time of the simulation could go up to four hours. The simulation time of the code for the equation 112 developed previously would be even greater. In Matlab, the calculations are time-consuming for equation 112 and require very small steps to converge. The steps were so small that, because of the lack of time, no conclusion could be made. To conclude, in Python, further implementation is required. A draft was started, but computation time is still concerning. All in all, it was demonstrated that equation 112 is a reliable model.

7 Conclusion

7.1 General conclusion

When comparing FEM and FDM relative errors in frequency measurement when changing the tension, the time interval or the number of nodes, there was a relationship between the periodicity of the error and the degree of the harmonic. This relationship is probably due to the discretizations done in each method. Moreover, in the case of the tension change, between the periodicity of the error in FEM and FDM at a given harmonic or at the fundamental, a phase shift of $\pi/2$ is visible. In the two other cases of changing the time interval or the number of nodes, both FEM and FDM have the same phase.

The error associated with frequency measurements is negligible in most practical applications. The models discussed are primarily intended for comparison with real-world conditions, where measurement instruments impose inherent limitations. For example, the fundamental frequency of a waveform is typically captured using a standard microphone with a sampling rate of 44100 Hz. This sampling rate directly influences the resolution of the FFT, thereby constraining the

accuracy of the extracted frequency. Similarly, when simulating a waveform, the choice of sampling rate introduces a corresponding measurement uncertainty. As demonstrated in this study, the discrepancy between frequencies obtained through the FFT, whether from simulated or experimental data and those derived analytically, is limited to a few hertz. Notably, despite this deviation from the theoretical solution, the measurement error observed in simulations remains representative of real-world behavior, as similar margins of error are consistently observed in experimental contexts.

Computation time depends significantly on the implementation. However, for this study, the computational time for both methods is approximately equivalent. While the FDM is generally faster, the FEM tends to be more efficient when handling a higher number of nodes. Additionally, FDM converges for larger time intervals compared to FEM, indicating that FDM reaches a stable solution "more quickly" than FEM.

From a mathematical and implementation perspective, FDM is more straightforward but requires knowledge of the object's shape, necessitating recalculations for different geometries. In contrast, while FEM is conceptually more complex to understand and implement, once the initial derivations are established, the object type and boundary conditions are incorporated into the stiffness (K) and mass (M) matrices. This allows for a more generalized approach, where a single FEM algorithm can be adapted for different objects simply by modifying these matrices.

Furthermore, given the availability of open-source FEM software, it is evident that in an industrial context, FEM would be more cost-effective and flexible. Developing a single general FEM algorithm and adapting it by modifying the K and M matrices would be more efficient for large-scale applications requiring frequent modifications. Conversely, FDM remains the more practical choice for smaller and more specialized applications due to its relative simplicity and ease of implementation. However, as experience with the bell, coding FDM can quickly become a hassle because of the long equations.

7.2 Additional possible work

Further implementation and investigation are recommended. A formal comparison of the algorithmic complexity of FEM and FDM would provide clearer insights into computational efficiency. Implementing FEM for the bell model using Python or an open-source FEM simulator would yield practical information regarding implementation challenges [2]. Additionally, examining the frequencies of guitar strings by incorporating the modulus of elasticity could demonstrate its negligible impact on frequency results. Finally, exploring alternative numerical methods such as the Boundary Element Method (BEM) would highlight the existence of methods beyond FEM and FDM.

References

- [1] E. Kaselouris, M. Bakarezos, M. Tatarakis, N. A. Papadogiannis, and V. Dimitriou, “A review of finite element studies in string musical instruments,” *Acoustics*, 2022, a comprehensive review of FEM modeling applied to string musical instruments, covering soundboards, resonance boxes, and fluid-structure interactions.
- [2] Y. Konaté, “Calcul par éléments finis des structures en coque,” École Polytechnique de Thiès, Tech. Rep., 1986, a study on shell structure modeling using the finite element method, analyzing theoretical aspects and computational implementation.
- [3] M. Raibaud, “Modélisation et simulation de systèmes discrétisés par la méthode des éléments finis dans le formalisme des systèmes hamiltoniens portuels : Application à la synthèse sonore,” IRCAM, Tech. Rep., 2018.
- [4] V. Manet, *La Méthode des Éléments Finis: vulgarisation des aspects mathématiques, illustration des capacités de la méthode*. CEL — Centre pour l’Édition Électronique Ouverte, 2012, version 6, HAL Id: cel-00763690v6; ORCID: 0000-0002-5493-5122; IdHAL: vincent-manet. [Online]. Available: <https://cel.hal.science/cel-00763690v6>
- [5] O. Granier, “L’équation des ondes de d’alembert et ses applications,” *Physique des Ondes, Ondes mécaniques - 2ème année de CPGE scientifique, voie PC*, unisciel*, 2012.
- [6] A. Joussen, “De la guitare...” *Bulletin de l’Union des Physiciens*, 1990.
- [7] F. Martin, “Étude d’un instrument à cordes,” *Sciences et Vie Junior*, 1989.
- [8] K. Polisano and M. Jedouaa, *TP Méthodes Numériques : Modélisation d’une corde de guitare et du tympan*, ENSIMAG, 2016.
- [9] J. BASSET, “Physique des ondes,” *Polycopié de cours Peip2 S3, Université Paris Saclay*, Année universitaire 2021-2022.
- [10] G. Derveaux, “Modélisation numérique de la guitare acoustique,” Ph.D. thesis, École Polytechnique, 2002, mathématiques [math]. PASTEL ID: pastel-00002585.
- [11] S. Deghboudj, *Méthode des éléments finis*. Université Larbi Tébessi - Tébessa, 2018.
- [12] D. Samir, *Méthode des éléments finis*. Research Gate, 09 11 September 2018.

- [13] Y. Rocard, “Sur l’amortissement des ondes sonores dans un milieu gazeux homogène,” *Journal de Physique et le Radium*, vol. 1, no. 12, pp. 426–437, 1930.
- [14] Projet ONDES, INRIA, “Modélisation et simulation de phénomènes de propagation d’ondes,” INRIA – Institut National de Recherche en Informatique et en Automatique, Rocquencourt, France, Tech. Rep., 2002, rapport d’activité 2002 du projet ONDES. Applications en acoustique musicale, géophysique, électromagnétisme, etc. [Online]. Available: <https://radar.inria.fr/rapportsactivite/RA2002/ondes/ondes.pdf>