



Movimiento de una onda en una cuerda considerando la rigidez del medio, modelamiento por medio de diferencias finitas en Python

Movement of a wave in a string considering the stiffness of the medium, finite difference modeling in Python

C. PERDOMO¹, M. REYES²

Recibido: 8 de septiembre de 2020 / Aceptado: 15 de diciembre de 2020

RESUMEN

El movimiento de una onda en una cuerda en la realidad se ve afectado por diversas características constructivas de la misma, su rigidez, amortiguación, entre otros parámetros que normalmente no se toman en cuenta en los cursos básicos de física. En este artículo se realiza la implementación de una simulación del movimiento de una onda en una cuerda tomando en cuenta la rigidez de la misma. Se utilizó el software Python, generando gráficos de señal y espectro de potencia en diferentes formas de excitación de la cuerda.

ABSTRACT

The movement of a wave in a string in reality is affected by various fabrication characteristics of the string, its stiffness, damping, among other parameters that are not normally taken into account in basic physics courses. In this article, the implementation of a simulation of the movement of a wave in a string is carried out, taking into account its stiffness. For this the Python software was used, generating signal graphics and power spectrum in different forms of excitation of the string.

PALABRAS CLAVES

python, onda en cuerda, rigidez, diferencias finitas

KEYWORDS

python, string wave, stiffness, finite differences

¹Universidad Nacional Autónoma de Honduras
en el Valle de Sula email:
carlos.perdomo@unah.edu.hn

²Universidad Nacional Autónoma de Honduras
en el Valle de Sula email:
marco.reyes@unah.edu.hn

* Esta obra está bajo una licencia Creative Commons Reconocimiento - NoComercial 4.0 Internacional

* This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

I | INTRODUCCIÓN

En los estudios básicos de ondas en cuerdas, generalmente se ha tratado solo el caso de una cuerda ideal. Por supuesto, las cuerdas reales tendrán algunas pérdidas por fricción (amortiguación), así como también cierta rigidez. Para tener una idea de cómo estos afectan las vibraciones, se considerará en este artículo el efecto de la rigidez. Los fenómenos de amortiguación y rigidez no se limitan a las ondas en una cuerda. Por ejemplo, el mismo fenómeno (y una ecuación similar) se encuentra cuando se trata de ondas electromagnéticas en un medio con pérdida o sonido en la atmósfera.

Para el modelado del comportamiento de un elemento de onda se utilizará Diferencias finitas en Python, igualmente, se desarrollarán las gráficas de espectros de potencia para los casos de una cuerda excitada por una onda senoidal, un pulso exponencial gaussiano, y en el caso más realista, por un tirón a la cuerda.

II | TEORÍA

1 | Caracterización del movimiento en una cuerda

Para entender la ecuación que define el movimiento de una onda, podemos valernos de la Figura 1, en la cual se toma un elemento diferencial de la cuerda sobre la cual viajará la onda.

En promedio, la onda viajará en la dirección x , por lo que los elementos de onda lo harán en el eje y . Consideremos ahora el movimiento a través del segmento de cuerda que está resaltado, y denotado por y_i .

La fuerza en el segmento i , vendrá de los segmentos adyacentes $i \pm 1$. La componente vertical de estas fuerzas, será la tensión de la cuerda multiplicada por el seno del ángulo que forman con la cuerda. Haciendo una sumatoria por la segunda Ley de Newton tenemos:

$$Dm \frac{d^2 y_i}{dt^2} = T \sin(q_{i+1}) - T \sin(q_i) \quad (1)$$

Donde la masa se puede reescribir como μDx , donde μ es la densidad lineal de masa. De igual forma podemos recordar que si Dx es lo suficientemente pequeño, entonces

$$Dx \approx dx \quad (2)$$

$$\sin(q) \approx \tan(q) \approx q = \frac{\Delta y}{\Delta x} \quad (3)$$

Utilizando la aproximación de diferencias finitas de la primera derivada para $\sin(q)$ y sustituyendo en (1) tenemos que:

$$\sin(q) \approx \frac{\Delta y}{\Delta x} \approx \frac{y_i - y_{(i-1)}}{Dx}$$

Tenemos que:

$$\frac{\Delta^2 y}{\Delta t^2} \approx \frac{T}{\mu} \frac{y_{(i+1)} - 2y_i + y_{(i-1)}}{(Dx)^2} \approx \frac{T}{\mu} \frac{\Delta^2 y}{\Delta x^2} \quad (4)$$

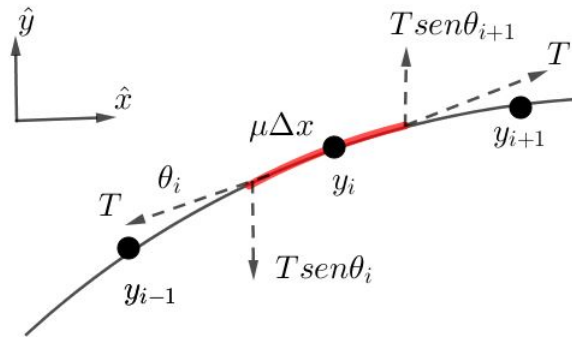


Figura 1: Fuerzas en un pequeño elemento de cuerda

Recordando que

$$c = \frac{\sqrt{T}}{\mu}$$

La ecuación (4) quedaría de la forma conocida para la ecuación de onda en un medio ideal

$$\frac{\partial^2 y}{\partial t^2} = c^2 \frac{\partial^2 y}{\partial x^2} \tag{5}$$

2 | Ecuación de onda para una cuerda considerando la rigidez

Si consideramos la ecuación (5), se asume una cuerda perfectamente flexible, esto significa, que la única fuerza en un segmento particular de cuerda es únicamente debido a la tensión, y que esa fuerza actúa directamente a lo largo de la cuerda. Aunque el modelamiento es bastante preciso para algunas cuerdas, siempre habrá una fuerza debido a la rigidez que se opone al doblamiento de la cuerda. Este es el caso, por ejemplo, de un hilo metálico, las vibraciones son muy diferentes a las de una cuerda completamente flexible.

Para un modelo realista, en el cual se considera la rigidez de la cuerda, debemos considerar que existirá otra fuerza, de naturaleza restauradora debido a la misma rigidez, por lo que la ecuación (1) debería ser expresada de la forma

$$Dm \frac{d^2 y_i}{dt^2} = T \sin(\theta_{i+1}) - T \sin(\theta_i) + F_{rigidez} \tag{6}$$

Para encontrar una expresión que modele la fuerza debido a la rigidez, se debe considerar las características constructivas de la cuerda (forma, área transversal, etc). Por lo que un modelo acertado es pensar en la cuerda como una varilla delgada (en este caso, de área trasversal circular y radio transversal a).

Una varilla es capaz de transmitir vibraciones transversales de la misma manera que una cuerda. Sin embargo, la dependencia de la frecuencia de la tensión es más complicada que en una cuerda. De hecho, una barra vibra bastante bien bajo tensión cero, las fuerzas elásticas dentro de la varilla suministran la fuerza de restauración necesaria en este caso. Cuando se dobla una varilla o barra, la parte exterior

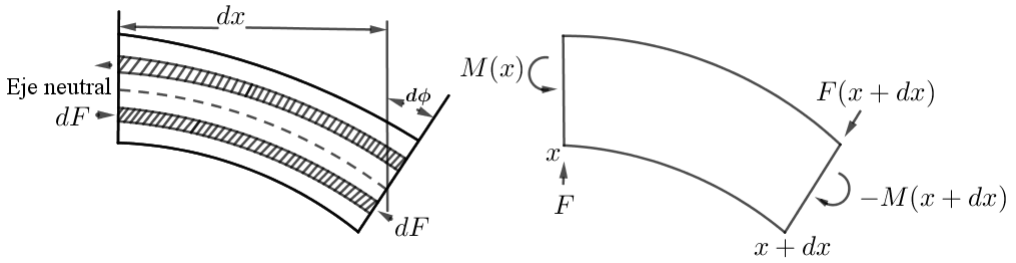


Figura 2: a). Flexión de deformaciones en una varilla b). Momentos flectores y fuerzas cortantes

se estira y la parte interior se comprime. En algún lugar intermedio hay un eje neutro cuya longitud permanece sin cambios, como se muestra en la figura (2). Un filamento ubicado a una distancia z por debajo del eje neutro se comprime en una cantidad df . La deformación es $zdf=dx$, y la cantidad de fuerza requerida para producir la deformación es

$$EdS z \frac{df}{dx}$$

Donde dS es el área de la sección transversal del filamento y E es el módulo de Young. El momento de esta fuerza alrededor de la línea central es

$$dM = EdS z \frac{df}{dx} z$$

Por lo que el momento total para comprimir todos los filamentos está dado por:

$$M = \int dM = E \frac{df}{dx} \int z^2 dS \tag{7}$$

Es común definir una constante k llamada radio de giro de la sección transversal tal que

$$k^2 = \frac{1}{S} \int z^2 dS \tag{8}$$

Donde $S = \int dS$ es el área transversal total. El radio de giro de algunas formas familiares se muestra en la figura (3). El momento flector es entonces:

$$M = E \frac{df}{dx} S k^2 \approx -E S k^2 \frac{d^2 y}{dx^2} \tag{9}$$

Dado que

$$df \approx - \frac{d^2 y}{dx^2} dx \quad \text{Para } df \text{ pequeño} \tag{10}$$

El momento flector no es el mismo para todas las partes de la barra. Para mantener la barra en equilibrio, debe haber una fuerza cortante F con un momento $F dx$, como se muestra en la figura (2b).

$$F dx = (M + dM) - M = dM \tag{11}$$

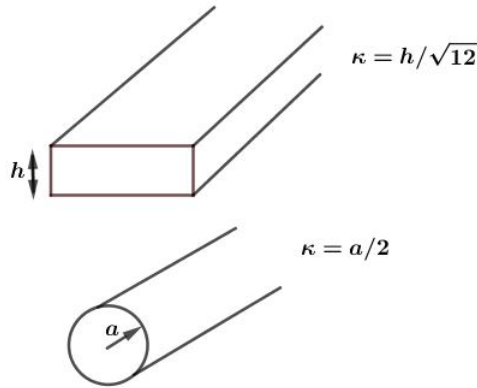


Figura 3: Radio de giro para algunas formas comunes

Por lo que, utilizando la ecuación (9)

$$F = \frac{\partial M}{\partial x} = -ESK^2 \frac{\partial^3 y}{\partial x^3} \tag{12}$$

Pero la fuerza de corte F tampoco es constante; la fuerza neta

$$dF = \frac{\partial F}{\partial x} dx \tag{13}$$

$$dF = -ESK^2 \frac{\partial^4 y}{\partial x^4} dx \tag{14}$$

produce una aceleración perpendicular al eje de la barra. La ecuación del movimiento entonces, se encuentra operando la ecuación (6) con las aproximaciones utilizadas en (2), (3) y (4) junto con (14) por lo que

$$\mu \frac{\partial^2 y}{\partial t^2} = T \frac{\partial^2 y}{\partial x^2} - ESK^2 \frac{\partial^4 y}{\partial x^4} \tag{15}$$

Dividiendo la ecuación (15) entre μ

$$\frac{\partial^2 y}{\partial t^2} = \frac{T}{\mu} \frac{\partial^2 y}{\partial x^2} - \frac{ESK^2}{\mu} \frac{\partial^4 y}{\partial x^4} \tag{16}$$

Sacando de factor común $T = \mu = c^2$

$$\frac{\partial^2 y}{\partial t^2} = c^2 \frac{\partial^2 y}{\partial x^2} - eL^2 \frac{\partial^4 y}{\partial x^4} \tag{17}$$

donde e es un parámetro adimensional de rigidez, dado por

$$e = k^2 \frac{ES}{TL^2} \tag{18}$$

donde r es el radio de giro de la cuerda, E es el módulo de Young, S es el área de sección transversal, es la tensión y L es la longitud de la cuerda.

El origen de la derivada de 4to orden, también está relacionada con la energía potencial de la cuerda durante el doblez, para una cuerda ideal y exible, la energía potencial está relacionada con la elasticidad de la cuerda, y es de la forma $\frac{1}{2}E(\frac{dy}{dx})^2$. La energía potencial asociada con el doblez es de la forma $\frac{1}{2}Y(\frac{d^2y}{dx^2})^2$, donde la constante de proporcionalidad es proporcional a una de las constantes de elasticidad del material (Giordano y Nakanishi, 2005).

El término agregado es el mismo que aparece en la ecuación de la viga (también llamada ecuación de Euler-Bernoulli), que modela el movimiento de una viga vibrante bajo la hipótesis que no hay esfuerzo cortante ni inercia rotacional. Se puede ver (17) como la generalización de una EDP para un material vibrante: el primer término se debe a la elasticidad del material (su capacidad de volver a la posición inicial después de una deformación) y el segundo, debido a la resistencia contra la flexión. Si el primer término es cero, el material no es elástico y obtenemos la ecuación de la viga. Por el contrario, si el segundo término es cero, el material no es rígido y obtenemos la ecuación de onda (Chaigne y Askenfelt, 1994).

La cuerda con rigidez basada en el modelo de Euler-Bernoulli es el modelo más utilizado. Sin embargo, existen otras ecuaciones que pueden modelar la vibración de materiales rígidos (y en particular cuerdas de piano). Entre ellos destaca el modelo de vigas de Timoshenko, que toma en cuenta el esfuerzo cortante y la inercia rotacional (Chabassier y Duru é, 2012).

3 | Espectro de Potencia

Una forma de analizar la dinámica de sistemas no lineales es mediante su espectro de potencia. Este es una función matemática que nos informa de cómo está distribuida la potencia o la energía (según el caso) de una señal sobre las distintas frecuencias de las que está formada.

El espectro de potencia de una variable dinámica está definida como el cuadrado del módulo de la transformada normalizada de Fourier de la variable dinámica, es decir:

$$S(w) = |g(w)|^2 \quad (19)$$

donde

$$g(w) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} y(t) e^{j\omega t} dt \quad (20)$$

donde $y(t)$ es la variable dinámica dependiente del tiempo. En nuestro caso, haremos uso de la Transformada Discreta de Fourier (DFT), esta es un método numérico utilizado para definir muestreos referentes a las frecuencias espectrales que componen una señal. Estudia funciones periódicas en parámetros cerrados, arrojando como resultado otra señal discreta. Haciendo uso de la transformada rápida de Fourier (FFT), la evaluación del espectro de potencia de una variable dependiente del tiempo se vuelve sencilla. Cabe señalar que la FFT no es una nueva transformada, sino que se trata de un algoritmo para el cálculo de DFT. Su importancia radica en el hecho que elimina gran parte de los cálculos repetitivos a los que está sometida la DFT, además, la FFT generalmente permite mayor precisión en el cálculo de

la DFT, disminuyendo los errores de redondeo (Pang, 2006).

A diferencia de la transformada de Fourier, el espectro de potencia no contiene información de fase, por lo que el espectro de potencia no es una transformación invertible: no es posible reconstruir la señal a partir del espectro de potencia. Sin embargo, el espectro de potencia tiene un rango más amplio de aplicabilidad (audio (Bernal, Gomez, y Bobadilla, 2009), sismología (Hernandez, 2009), Densidad MER en pacientes con Parkinson (Echeverry, Guarnizo Lemus, y Orozco, 2007), entre otros) y se puede utilizar para algunas señales que no tienen una transformada de Fourier signi cativa (como las que resultan de procesos aleatorios). Dado que el espectro de potencia no contiene información de fase se aplica en situaciones donde la fase no se considera útil o para datos que contienen mucho ruido, ya que la información de fase se corrompe fácilmente por el ruido.

4 | Aproximación de la ecuación de onda mediante diferencias finitas

La cuarta derivada parcial, se puede escribir utilizando diferencias finitas de la forma

$$\frac{\partial^4 y}{\partial x^4} \approx \frac{y(i+2;n) - 4y(i+1;n) + 6y(i;n) - 4y(i-1;n) + y(i-2;n)}{(Dx)^4} \quad (21)$$

Por lo que la ecuación (17) se puede escribir mediante aproximaciones de diferencias finitas como:

$$c^2 \frac{y(i;n+1) - 2y(i;n) + y(i;n-1)}{(Dt)^2} = \frac{y(i+2;n) - 4y(i+1;n) + 6y(i;n) - 4y(i-1;n) + y(i-2;n)}{(Dx)^4} \quad (22)$$

Reordenando la ecuación con la intención de dejar $y(i;n+1)$ en término de los pasos previos, obtenemos:

$$y(i;n+1) = (2 - 2r^2 - 6er^2M^2)y(i;n) - y(i;n-1) + r^2(1 + 4eM^2)(y(i+1;n) + y(i-1;n)) - er^2M^2[y(i+2;n) - y(i-2;n)] \quad (23)$$

donde $r = cDt/Dx$, y $M = L/Dx$. Visto en un mallado, el comportamiento sería como en la Figura 4. La ecuación (22) produce el desplazamiento de la cuerda en el paso de tiempo $n + 1$ en términos de desplazamiento en los pasos de tiempo anteriores.

Para efectos de estabilidad en la ecuación (22) se consideran los pasos tales que cDt/Dx sea igual a 1, esto es a propósito. Resulta que cuando los términos de orden superior se cancelan al deducir (22). Si se repitiesen los cálculos con cDt/Dx un poco más pequeño y el mismo valor Dx , los resultados sería menos precisos. Por otro lado, reducir el paso espacial dejando fijo el paso temporal, por ejemplo si reducimos Dx a la mitad, dejando fijo Dt , nos daría un valor de $r = 2$; en este caso, los pulsos de onda crecerían rápidamente con el tiempo, y el cálculo de desplazamiento a lo largo de la cuerda podría divergir.

En principio, solo se requiere una pequeña modificación al resultado obtenido si fuese una cuerda ideal, sin embargo, hay una ligera complicación. La ecuación (20) implica el desplazamiento en sí mismo a

Figura 4: Dependencia del desplazamiento $y(i, n+1)$ (marcado con círculo blanco) de los valores previos (marcado con círculos rellenos) en el muestreo para una cuerda rígida

Figura 5: Condiciones de frontera articuladas en los extremos de la cuerda. La cuerda real termina en el desplazamiento $y(i=0) = 0$ y toma el valor $y(i=+1)$ en el punto fantasma.

unidades del punto en análisis. Ahora, cuando la rutina de propagación se acerca al final de la cuerda es decir, la última sección móvil de la cuerda, necesitamos saber el desplazamiento al final y también el desplazamiento una unidad más allá de los extremos. La forma más sencilla de lidiar con este problema es asumir que los extremos de la cadena están "articulados". Es decir, tomamos el desplazamiento en cada extremo como cero y suponemos que hay ubicaciones fantasma una unidad más allá de los extremos que tienen desplazamientos opuestos a los desplazamientos en ubicaciones una unidad dentro de los extremos. Esto se ilustra en la figura 5.

Aquí la cadena real termina en $x=0$ (correspondiente al paso espacial $i=0$ y un extremo izquierdo), pero el programa le permite extenderse efectivamente a $x=1$. El desplazamiento de este punto fantasma es $y(i=-1) = -y(i=+1)$. Este tratamiento simple de las condiciones de frontera será adecuado para nuestro trabajo y en realidad es el procedimiento utilizado en el modelado detallado de cuerdas de piano.

III | IMPLEMENTACIÓN DEL ALGORITMO

Se utilizará el método de diferencias finitas, combinado por el método iterativo de Gauss Seidel para resolver el sistema de ecuaciones lineales producido, utilizando el lenguaje Python.

Aunque este método puede aplicarse a cualquier sistema de ecuaciones lineales que produzca una matriz (cuadrada, naturalmente pues para que exista solución única, el sistema debe tener tantas ecuaciones como incógnitas) de coeficientes con los elementos de su diagonal no-nulos, la convergencia del método solo se garantiza si la matriz es diagonalmente dominante o si es simétrica y, a la vez, de tanda positiva.

Aun cuando, por simplicidad, se ha incorporado el procedimiento iterativo de Gauss-Seidel en el algoritmo, es recomendable en general usar una técnica directa como la eliminación Gaussiana cuando el sistema es pequeño, del orden de 100 o menor, ya que la simetría y la propiedad de ser positiva de tanda aseguran la estabilidad con respecto a errores de redondeo (Díaz Ferrer y Cruz, 2019).

1. Librerías

Listing 1: Se cargan las librerías a utilizar

```

1 import matplotlib.pyplot as plt
  from matplotlib import animation
3 import numpy as np
  from scipy.fftpack import rfft, irfft, fftfreq

```

2. Condiciones iniciales

Listing 2: Definición de condiciones de inicialización del problema

```

class Guass_seidel:
2   def __init__(self,L,dx,T,e):
      self.L = L ##Valor se ingresa al llamar la clase, longitud de la cuerda
4      self.dx = dx #paso en x, Valor se ingresa al llamar la clase
      self.r = 0.25 #Valor de r utilizado para asegurar estabilidad
6      self.c = 300 #velocidad de la onda
      self.T = T #Valor se ingresa al llamar la clase
8      self.dt = self.dx/(4*self.c) #Se calcula un dt estable
      self.e = e #epsilon

```

3. Inicialización

Listing 3: Se define la inicialización del movimiento

```

1   def initialization(self):
      self.string = []
3      for i in range(int(self.T/self.dt)):
          self.string.append([])
5          for j in range(int(self.L/self.dx)):
              self.string[i].append(0)

```

1. Inicialización por un tirón a la cuerda

```

x_0 = int((self.L/self.dx)/8)
2 self.string[1][x_0] = 0.1
k1 = 0.1/(x_0 - 1)
4 k2 = 0.1/(int(self.L/self.dx) - 1 - x_0)
for l in range(1,x_0):
6 self.string[1][l] = k1*(l - 1)
for m in range(x_0,int(self.L/self.dx) - 1):
8 self.string[1][m] = 0.1 - k2*(m - x_0)
self.string[1][0] = -self.string[1][2]
10 self.string[1][-1] = -self.string[1][-3]
#plt.plot(self.string[1])
12 self.string[0] = self.string[1][:]

```

2. Inicialización por excitación senoidal

```

for i in range(1,int(int(self.L/self.dx) - 1):
2 self.string[0][i] = np.sin(10*np.pi/(48.5)*(i - 1))
self.string[1][i] = np.sin(10*np.pi/(48.5)*(i - 1))
4 self.string[0][0] = - self.string[0][2]
self.string[0][-1] = - self.string[0][-3]
6 self.string[1][0] = - self.string[1][2]
self.string[1][-1] = - self.string[1][-3]

```

3. Inicialización por un "golpe gaussiano"

```

for i in range(1,int(int(self.L/self.dx) - 1):
2 self.string[0][i] = np.e**(-0.1*(i - 30)**2)
self.string[0][1] = 0
4 self.string[0][-2] = 0
self.string[0][0] = - self.string[0][2]
6 self.string[0][-1] = - self.string[0][-3]
self.string[1] = self.string[0][:]
8
return 0

```

4. Implementación de esquemas por diferencias nitas

```

1 def calculation(self):
print (1)
3 M = int(self.L/self.dx)
for i in range(2,int(self.T/self.dt)):
5 for j in range(2,M-2):
self.string[i][j] = (2 - 2*self.r**2 -
6*self.e*self.r**2*M**2)*self.string[i - 1][j] - self.string[i -
2][j] + self.r**2*(1 +
4*self.e*M**2)*(self.string[i-1][j+1]+self.string[i-1][j-1]) -
self.e*self.r**2*M**2*(self.string[i-1][j+2]+self.string[i-1][j-2])

```

```

7         self.string[i][0] = -self.string[i][2]
          self.string[i][-1] = -self.string[i][-3]
9     return self.string

```

5. Generación de grá cos

```

1     def fft(self):
3         t_plot = np.arange(0,self.T,self.dt)
          print (len(t_plot))
          amplitude_record = []
          for i in range(int(self.T/self.dt)):
7             amplitude_record.append(self.string[i][20])
          print (len(amplitude_record))

```

6. Grá co de señal

```

2     plt.subplot(121)
          plt.title('Señal de la cuerda vs tiempo')
          plt.ylabel('Señal (Unidades arbitrarias)')
          plt.xlabel('Tiempo (s)')
          plt.plot(t_plot,amplitude_record, label = 'epsilon = '+str(self.e))

```

7. Grá co de Espectro de Potencia

```

2     freq = fftfreq(len(amplitude_record), d=self.dt)
          freq = np.array(abs(freq))
          f_signal = rfft(amplitude_record)
          f_signal = np.array(f_signal**2)
          plt.title('Espectro de Potencia')
          plt.ylabel('Potencia (unidades arbitrarias)')
          plt.xlabel('Frecuencia (Hz)')
          plt.xlim(0.0,5000.0)
          plt.plot(freq,f_signal,label = 'epsilon = '+str(self.e))
10
          return 0

```

8. Generación de Grá cos

```

1     plt.figure(figsize = (8,8))
          A = Guass_seidel(1,0.01,0.05,0)
          A.initialization()
          data_record = A.calculation()
          A.fft()
5
          B = Guass_seidel(1,0.01,0.05,1e-5)
          B.initialization()
          data_record = B.calculation()
9

```

```

B.fft()
11
C = Guass_seidel(1,0.01,0.05,2e-5)
13 C.initialization()
data_record = C.calculation()
15 C.fft()

17
plt.legend()
19 plt.savefig('señal_cuerda_senoidal.png',dpi = 144)
plt.show()

```

9. Gráfico Animado

```

1 fig = plt.figure(figsize = (8,6))
ax = plt.axes(xlim=(0, 1),ylim = (-0.5,0.5))
3 line, = ax.plot([], [], 'k')
def init():
5     line.set_data([], [])
    return line,
7
def animate(i):
9     x_plot = np.arange(0.01,0.99,0.01)
    y_plot = []
11     for j in range(1, int(A.L/A.dx) - 1):
        y_plot.append(data_record[i][j])
13     line.set_data(x_plot,y_plot)
    return line,
15     #camera.snap()
anim = animation.FuncAnimation(fig, animate, init_func=init, frames=200, interval=50,
    blit=True)
17 anim.save('cuerda_animada_tiron.gif', fps=20, writer='Pillow')
plt.show()

```

IV | RESULTADOS GRÁFICOS

Se corrió la simulación con cada tipo de inicialización, obteniendo un gráfico animado del movimiento del pulso en la cuerda, el gráfico de señal de la cuerda y el espectro de potencia (ver siguientes gráficos).

V | CONCLUSIONES

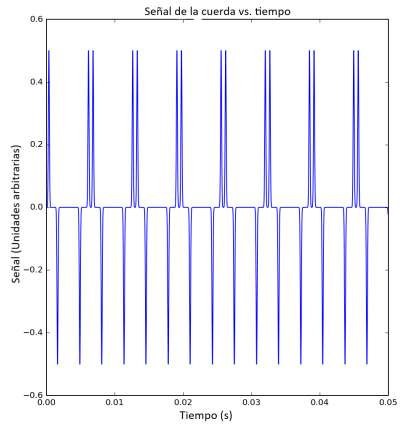
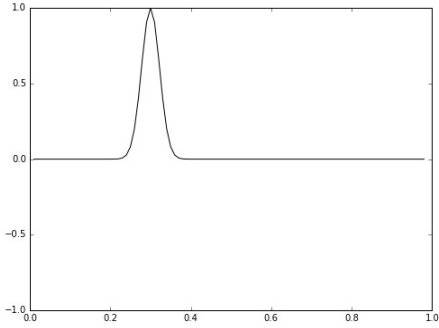
1. Se realizó la simulación del movimiento de una cuerda considerando la rigidez que esta presenta obteniendo los gráficos de señales de la onda con distintos tipos de excitación.
2. Se obtuvo el diagrama de Espectro de Potencia para excitaciones senoidales, gaussianas y de pulso mostrando desplazamientos de picos de potencias sobre todo en altas frecuencias a medida que aumenta la rigidez. De igual manera, se observan los picos de potencia en frecuencias más altas cuando la excitación de la cuerda era senoidal.

(a) Movimiento de una onda en cuerda excitada senoidalmente (b) Señal de una cuerda excitada senoidalmente

Figura 6

(a) Espectro de frecuencia de una cuerda ideal sometido a una excitación de alta frecuencia. (b) Espectro de frecuencia de una cuerda real

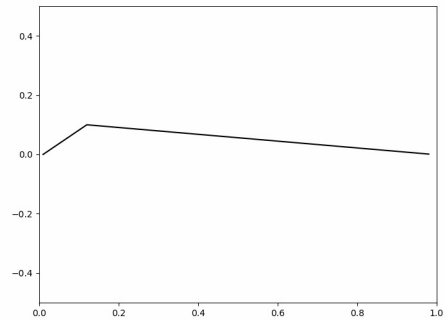
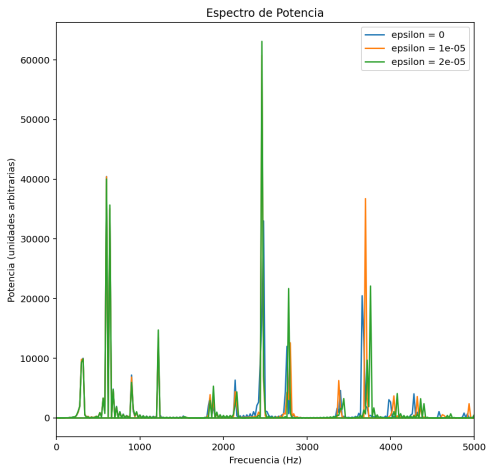
Figura 7



(a) Movimiento de una onda en cuerda excitada por un pulso Gaussiano.

(b) Señal de una cuerda excitada por pulso gaussiano.

Figura 8



(a) Espectro de frecuencia de una cuerda sometido a una excitación Gaussiana.

(b) Movimiento de una onda en cuerda excitada por un tirón.

Figura 9

