

UNIVERSIDAD DE CÁDIZ

FACULTAD DE CIENCIAS

GRADO EN MATEMÁTICAS

SIMULACIÓN NUMÉRICA DE ECUACIONES DE ONDA.

Trabajo de fin de grado presentado por

Ana Cuevas de Cózar

Tutores: Dr. José Rafael Rodríguez Galvan

D. Daniel Acosta Soba

Firma de la alumna

Firma de los tutores

Puerto Real, Cádiz, Junio de 2.023

Abstract

In this work we will address the study of the partial differential equation for wave phenomena, from a numerical perspective. We will establish a theoretical physics foundation to understand the origin of the wave equation and its derivation.

To proceed further, we are going to tackle the conservation law of the wave equation and the traveling wave type solutions.

Afterwards, using the finite differences method, we aim to obtain a numerical solution, by doing a discretization in space and time to approximate it. We will reintroduce the former aspects back from this perspective. We will also be conducting an acoustic and graphical comparative, varying the parameters and conditions obtained to verify their effects on it.

To sum up, we are going to present the remarks obtained throughout the work, and the relevant results.

A mis padres.

Resumen

En este trabajo trataremos el estudio de la ecuación de onda en derivadas parciales, desde una perspectiva numérica, explorando diferentes enfoques. Comenzaremos estableciendo una base física para comprender el origen de la ecuación de onda y como se deriva.

A continuación, abordaremos aspectos como la ley de la conservación de la energía o las soluciones de tipo onda viajera.

Luego, empleando el método de las diferencias finitas obtendremos una solución numérica de la ecuación de onda, haciendo una discretización en espacio y tiempo para aproximarla. También retomaremos los aspectos anteriores desde este punto de vista. Seguiremos realizando una comparativa, gráfica y sonora de las aproximaciones obtenidas, cambiando los parámetros y condiciones obtenidas comprobar su efecto en la misma.

Por último, presentaremos las conclusiones obtenidas a lo largo de este trabajo, resumiendo las observaciones y los resultados relevantes.

Agradecimientos

Cuando comencé el grado en matemáticas con todos mis compañeros, creo que ninguno sabíamos como era esta carrera. A lo largo de los años me enorgullezco en decir que hemos construido amistades duraderas, pasado juntos momentos complicados y aprendido mucho más de lo que podíamos imaginar.

Por todo esto quiero agradecerles a ellos y sobre todo a mi compañera y amiga Marina por estar a mi lado en cada paso del camino, porque gracias a eso hemos conseguido avanzar juntas.

Creo que todos tenemos que agradecer a nuestros profesores de la carrera, por intentar que aprendamos y seamos críticos. Sobre todo a mis directores de este trabajo por su paciencia infinita y su ayuda en todo momento.

También debo agradecer a mis padres, por creer siempre en mi y escucharme cada día, aún cuando ellos no comprendían lo que les contaba. Gracias, por construirme como persona y hacerme llegar a donde estoy.

Por último quiero agradecer a mi novio, porque estos últimos años, tan duros a la par que gratificantes, ha sabido llenarme de confianza.

Gracias a todos y espero poder devolveros algún día todo lo que me habéis dado.

Ana Cuevas de Cózar

junio 2023

Índice general

1	Introducción	1
2	Conceptos previos	5
2.1	Espacios de Lebesgue	5
2.2	Producto interior y suma por partes en L^2	6
3	Definición de la ecuación de onda	13
3.1	La ecuación de onda unidimensional	14
3.2	Clasificación de la ecuación de onda	16
3.3	Soluciones de tipo onda viajera	17
3.4	La energía en la ecuación de onda	19
4	Resolución numérica de la ecuación de onda	21
4.1	Métodos principales de resolución	21
4.2	Método de diferencias finitas	21
4.3	Dominio de dependencia	24
4.4	Consistencia	26
4.5	Análisis de Von Neumann	28
4.6	Resolución numérica para las soluciones de tipo viajero	31
4.7	Ley de energía discreta	32
5	Modelado en Python y comparativa	35
5.1	Soluciones	35
5.1.1	$f(x) = \sin(\pi x)$, $g(x) = 0$	35
5.1.2	$f(x)$ tipo Struck, $g(x) = 0$	37
5.1.3	$f(x) = 0$, $g(x)$ tipo Struck	38

ÍNDICE GENERAL

5.2	Soluciones tipo onda viajera	40
5.3	Comparación variando el valor de μ	42
5.3.1	$\mu \ll 1$	42
5.3.2	$\mu > 1$	43
5.4	Archivos .wav y síntesis de sonido	45
5.4.1	$f(x) = \sin(2\pi x), g(x) = 0$	45
6	Conclusiones y proyectos futuros	47
A	Anexo	49
A.1	Resolución numérica	49
A.2	Wave	53
A.3	Imágenes	54
	Bibliografía	57

Introducción

Los grandes avances en la computación y la tecnología en los últimos años, han logrado que las matemáticas, y en particular el campo de las ecuaciones diferenciales, hayan cambiado notablemente. Las EDP son una poderosa herramienta que nos permiten modelar un gran número situaciones en muchas disciplinas, a destacar los fenómenos físicos. Sin embargo, la exactitud con la que lo hagan dependerá de diversos factores. Conforme los fenómenos a modelar se compliquen, también lo harán estas, añadiendo nuevos términos, por ello en ocasiones resolver de manera exacta dichos modelos puede presentar dificultades.

En la actualidad, los métodos numéricos nos brindan la oportunidad de obtener aproximaciones cada vez más precisas de los modelos en ocasiones en las que antes teníamos que conformarnos con soluciones parciales o marcos muy específicos.

En este trabajo nos centraremos en la EDP de onda. Esta fue modelada por Jean le Rond d'Alembert en 1752, de forma similar a como lo realizaremos en nuestro estudio [1]. Esto provocó que pocos años más tarde, tanto D.Bernoulli, como Leonhard Euler se interesasen en el estudio de las ondas acústicas. Este último utilizó lo que hoy conocemos como series de Fourier, para obtener una solución de la ecuación de onda, más tarde, se descubriría que cualquier solución de d'Alembert, se podría construir como suma de soluciones de tipo Euler y viceversa.

1. INTRODUCCIÓN

A partir de entonces, se han realizado un gran despliegue de modelos para describir diversos fenómenos observables en la realidad. Podríamos destacar su uso en los campos de la sísmica [2], la acústica [3] y la óptica [4], usando asiduamente los métodos numéricos para la aproximación de soluciones.

La belleza de estas aproximaciones numéricas, es que, conforme la potencia de los dispositivos va incrementando, a la par que los métodos numéricos, las gráficas y distintos tipos de archivos que podemos crear a partir de las mismas, mejorarán de forma significativa.

Retomando la ecuación de onda, nos encontramos con diversos métodos numéricos utilizados, como el método de las diferencias finitas [5], el de los elementos finitos [6], o los métodos espectrales [7].

Nosotros nos centraremos en dar una visión general de la resolución de la ecuación de onda, utilizando el método de las diferencias finitas anteriormente mencionado.

En el capítulo 2 nos adentraremos en los conceptos previos necesarios para la comprensión del trabajo, como los Espacios de Lebesgue, el concepto de producto interior, o el mallado que utilizaremos para definir nuestro esquema numérico.

En el capítulo siguiente, 3, introduciremos la ecuación de onda desde un marco físico y realizaremos una clasificación de la misma. Posteriormente nos centraremos en las soluciones de tipo onda viajera. Por último, analizaremos la energía de la ecuación.

Una vez tenemos todos los cimientos contruidos, en el capítulo 4, haremos uso de uno de los métodos más utilizados para la resolución numérica, el método de las diferencias finitas, para aproximar numéricamente la ecuación de ondas. Continuando el capítulo, analizaremos la consistencia y estabilidad de las aproximaciones obtenidas. Acabaremos trasladando tanto las soluciones de tipo onda viajera como la conservación de la energía al ámbito discreto.

Concluiremos el grueso del trabajo, llevando a la práctica la teoría de los capítulos anteriores elaborando un script de Python, que nos permita computar las aproximaciones propuestas. Este se puede encontrar en el Anexo A y en el repositorio de Github del trabajo **Simulación numérica de ecuaciones de onda**. En el capítulo 5, compararemos las aproximaciones obtenidas desde distintos puntos de vista. Comprobaremos cuanto se aproximan a la solución exacta, y como se relacionan con las soluciones de tipo onda viajera obtenidas, variaremos ciertos parámetros para observar su implicación en la estabilidad de las aproximaciones, y sintetizaremos la onda, obteniendo archivos de audio.

Como cierre, en el apartado de conclusiones 6, sintetizaremos los puntos mas importantes de este trabajo y presentaremos sugerencias para seguir explorando en esta área de estudio.

Conceptos previos

Para facilitar la comprensión del tema que vamos a tratar hay varios conceptos que debemos explicar con anterioridad, en ello nos centraremos en este capítulo.

2.1 Espacios de Lebesgue

Comenzaremos definiendo los espacios de Lebesgue. Un concepto clásico que se utiliza de manera recurrente en varios campos de las matemáticas [8].

DEFINICIÓN 2.1.1: Espacio de Lebesgue

Para $E \subset \mathbb{R}^n$, con $n \in \mathbb{N}$ medible y $p \in \mathbb{R}$ tal que $1 \leq p < \infty$ definimos, el Espacio de Lebesgue $L^p(E)$, como la colección de funciones medibles, en la que se realiza una identificación entre las funciones que son iguales en casi todo, que son p -integrables, es decir, las funciones f para las cuales

$$\|f\|_{L^p(E)} = \left(\int_E |f|^p \right)^{\frac{1}{p}} < \infty. \quad (2.1)$$

En lo que sigue del trabajo trabajaremos sobre este tipo de espacios. Sigamos viendo qué características tendrán nuestros espacios.

2. CONCEPTOS PREVIOS

DEFINICIÓN 2.1.2: Espacio de Banach

Un espacio de Banach $(E, \|\cdot\|)$ es un espacio vectorial normado, en el que E es completo bajo la métrica inducida por la norma $\|\cdot\|$. Es decir, cada serie de Cauchy en E es una serie convergente.

Por lo tanto, tomando $\|f\|_{L^p(E)}$ como la norma asociada a $L^p(E)$, este será un espacio de Banach.

DEFINICIÓN 2.1.3: Espacio de Hilbert

Un espacio de Banach E se llama de Hilbert si tiene un producto interior asociado y es completo bajo su métrica.

En nuestro caso trabajaremos sobre L^2 , que será un espacio de Hilbert y por lo tanto vamos a definir un producto interior en el mismo.

2.2 Producto interior y suma por partes en L^2

DEFINICIÓN 2.2.1: Producto interior en L^2

Sean f y g dos funciones definidas en E , definimos el producto interior en $L^2(E)$, de ellas como:

$$\langle f, g \rangle_{L^2(E)} = \int_E f(x)g(x)dx.$$

La norma inducida por este producto interior, será la que utilizaremos. Esta se corresponderá con la norma definida por 2.1. Para funciones que dependen de más de una variable, como por ejemplo del espacio y el tiempo, $u(x, t)$, fijando un instante t , quedará:

$$\langle u_1(\cdot, t), u_2(\cdot, t) \rangle_{L^2(E)} = \int_E u_1(x, t)u_2(x, t)dx, \quad \|u(\cdot, t)\|_{L^2(E)} = \sqrt{\langle u(x, t), u(x, t) \rangle_{L^2(E)}}. \quad (2.2)$$

Esto a partir de ahora lo denotaremos como $\langle u_1, u_2 \rangle_{L^2(E)}$ y $\|u\|_{L^2(E)}$ respectivamente.

En este punto nuestro objetivo será llegar a una generalización discreta de este concepto, ya que lo usaremos en la sección 4.7. Para ello vamos a definir un mallado tal y como

lo usaremos mas adelante.

DEFINICIÓN 2.2.2: Mallado

Sea $[a, b] \times [c, d]$ con $b \geq a, d \geq c$ y sean $n, m \in \mathbb{N}$. Definimos h como tamaño del mallado en espacio y el paso de tiempo k , de forma que los puntos de nuestro mallado serán:

$$(x, t)_{j,r} = (x_j, t_r), \quad \text{donde}$$

$$x_j = jh \quad \text{para } j = 0, 1, \dots, m,$$

$$t_r = rk \quad \text{para } r = 0, 1, \dots, n.$$

Vemos que nuestro mallado sería del tipo:

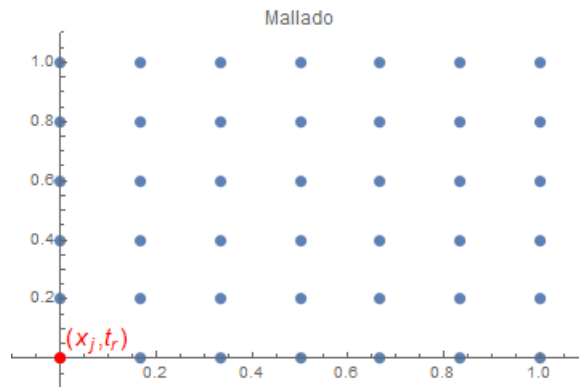


Figura 2.1: Mallado

Una vez ya definido este mallado retomemos la generalización de 2.2. Más información sobre como se aproxima al producto interior se puede encontrar en [9].

2. CONCEPTOS PREVIOS

DEFINICIÓN 2.2.3: Producto interior en dominios discretos

Sean u y w dos funciones definidas en nuestro mallado y tomemos D como la partición del interior del intervalo espacial $[a, b]$, es decir, $D = \{x_1, \dots, x_{m-1}\}$. Para ello debemos fijar un instante temporal t_r . En este contexto, definiremos el producto interior sobre D como:

$$(u(\cdot, t_r), w(\cdot, t_r))_D = \sum_{j=1}^{m-1} h u(x_j, t_r) w(x_j, t_r), \quad (2.3)$$

Para este tipo de casos usaremos la siguiente notación:

$$(u_{\cdot, r}, w_{\cdot, r})_D = \sum_{j=1}^{m-1} h u_{j, r} w_{j, r}. \quad (2.4)$$

La norma asociada a este producto escalar es: $\|u\|_D = \sqrt{(u, u)_D} \geq 0$.

El producto interior discreto, no tiene porque ser definido expresamente en un mallado, pero, ya que este será el contexto en el que lo utilizaremos a lo largo de el trabajo, es conveniente definirlo en el interior del mismo.

Una vez definidas las operaciones que nos harán falta a lo largo del trabajo, y para simplificar la notación que usaremos, necesitaremos los siguientes conceptos:

DEFINICIÓN 2.2.4: Desplazamiento progresivo y regresivo

Sean u una función discretizada en el mallado antes descrito. Definiremos el desplazamiento hacia delante (resp. desplazamiento hacia atrás) como:

$$e_{x+} u_{j, r} = u_{j+1, r}, \quad (e_{x-} u_{j, r} = u_{j-1, r} \text{ respectivamente.})$$

Continuando con este concepto podríamos aproximar la derivada de primer orden de la siguiente forma.

DEFINICIÓN 2.2.5: Aproximaciones progresivas, regresivas y centradas

Sea u una función discretizada en un mallado como el la definición 2.2.2. Definiremos las diferencias aproximadas hacia adelante, atrás o centradas (más conocidas como *Forward*, *backward* y *centered difference approximations*) respectivamente, como:

- $\delta_{x+} u_{j,r} = \frac{1}{h} (u_{j+1,r} - u_{j,r}),$
- $\delta_{x-} u_{j,r} = \frac{1}{h} (u_{j,r} - u_{j-1,r}),$
- $\delta_x u_{j,r} = \frac{1}{2h} (u_{j+1,r} - u_{j-1,r}).$

Podemos hacer lo mismo para el eje temporal, denotándolas como $\delta_{t+}, \delta_{t-}, \delta_t$

Para segundo orden definimos,

DEFINICIÓN 2.2.6: Aproximación centrada de la derivada de segundo orden

Sea u una función discretizada en un mallado como el de la definición 2.2.2. Definimos la aproximación centrada de la derivada de segundo orden como:

$$\begin{aligned} \delta_{xx} u_{j,r} &= \delta_{x+x-} (u_{j,r}) = \frac{1}{h^2} (e_{x+} u_{j,r} - 2u_{j,r} + e_{x-} u_{j,r}) = \\ &= \frac{1}{h^2} (u_{j+1,r} - 2u_{j,r} + u_{j-1,r}) \cong \frac{\partial^2}{\partial x^2} u_{j,r}. \end{aligned}$$

Al igual que la definición anterior esto será análogo en el eje temporal.

Veamos ahora la relación que guarda este concepto con 2.2.5.

$$\begin{aligned} \delta_{x+} (\delta_{x-} (u_{j,r})) &= \delta_{x+} \left(\frac{1}{h} (u_{j,r} - u_{j-1,r}) \right) \\ &= \frac{1}{h} (\delta_{x+} u_{j,r} - \delta_{x+} u_{j-1,r}) \\ &= \frac{1}{h} \left(\frac{1}{h} (u_{j+1,r} - u_{j,r}) - \frac{1}{h} (u_{j,r} - u_{j-1,r}) \right) \\ &= \frac{1}{h^2} (u_{j+1,r} - 2u_{j,r} + u_{j-1,r}) \\ &= \delta_{xx} u_{j,r}, \end{aligned}$$

2. CONCEPTOS PREVIOS

$$\begin{aligned}
\delta_{x-}(\delta_{x+}(u_{j,r})) &= \delta_{x-}\left(\frac{1}{h}(u_{j+1,r} - u_{j,r})\right) \\
&= \frac{1}{h}(\delta_{x-}u_{j+1,r} - \delta_{x-}u_{j,r}) \\
&= \frac{1}{h}\left(\frac{1}{h}(u_{j+1,r} - u_{j,r}) - \frac{1}{h}(u_{j,r} - u_{j-1,r})\right) \\
&= \frac{1}{h^2}(u_{j+1,r} - 2u_{j,r} + u_{j-1,r}) \\
&= \delta_{xx}u_{j,r}.
\end{aligned}$$

Por lo tanto, podemos concluir que la posición de δ_{x+} y δ_{x-} en este contexto es intercambiable, es decir, $\delta_{xx}u_{j,r} = \delta_{x+}(\delta_{x-}(u_{j,r})) = \delta_{x-}(\delta_{x+}(u_{j,r}))$.

Para finalizar la sección, hay otro concepto que tendremos que introducir. Este nos ayudará a hacer una representación discreta de la integración por partes.

LEMA 2.2.1: Representación de la suma por partes

Sean u, w funciones definidas en el mallado 2.2.2 y sea $D = \{x_1, \dots, x_m - 1\}$, una partición del interior del intervalo $[a, b]$, tal y como en la definición 2.2.3. La representación de la suma por partes utilizando el producto interior que usaremos en lo que sigue es, fijando un instante temporal r :

$$(u_{\cdot,r}, \delta_{x+}w_{\cdot,r})_D = -(\delta_{x-}u_{\cdot,r}, w_{\cdot,r})_D + u_{m-1,r}w_{m,r} - u_{0,r}w_{-1,r} \quad (2.5)$$

$$(u_{\cdot,r}, \delta_{x-}w_{\cdot,r})_D = -(\delta_{x+}u_{\cdot,r}, w_{\cdot,r})_D - u_{1,r}w_{0,r} + u_{m,r}w_{m-1,r}. \quad (2.6)$$

Al igual que en la definición 2.2.3, este lema se podría dar en $D' = \{x_0, \dots, x - m\}$, pero utilizando como ejemplo (2.5), nos quedaría

$$(u_{\cdot,r}, \delta_{x+}w_{\cdot,r})_{D'} = -(\delta_{x-}u_{\cdot,r}, w_{\cdot,r})_{D'} + u_{m,r}w_{m+1,r} - u_{-1,r}w_{0,r}, \quad (2.7)$$

podemos observar que en los últimos términos quedan fuera del mallado definido, estos se llaman puntos virtuales del mallado.

Este lema es fácilmente demostrable siguiendo las igualdades.

Demostración.

$$\begin{aligned}
 (u_{\cdot,r}, \delta_{x^+} w_{\cdot,r})_D &= \sum_{j=0}^m h u_{j,r} \frac{1}{h} (w_{j+1,r} - w_{j,r}) = \\
 &= - \sum_{j=0}^m h \frac{1}{h} (u_{j,r} - u_{j-1,r}) w_{j,r} + u_{m,r} w_{m+1,r} - u_{-1,r} w_{0,r} = \\
 &= - (\delta_{x^-} u_{\cdot,r}, w_{\cdot,r})_D + u_{m,r} w_{m+1,r} - u_{-1,r} w_{0,r}
 \end{aligned}$$

$$\begin{aligned}
 (u_{\cdot,r}, \delta_{x^-} w_{\cdot,r})_D &= \sum_{j=0}^m h u_{j,r} \frac{1}{h} (w_{j,r} - w_{j-1,r}) = \\
 &= - \sum_{j=0}^m h \frac{1}{h} (u_{j+1,r} - u_{j,r}) w_{j,r} - u_{0,r} w_{-1,r} + u_{m+1,r} w_{m,r} = \\
 &= - (\delta_{x^+} u_{\cdot,r}, w_{\cdot,r})_D - u_{0,r} w_{-1,r} + u_{m+1,r} w_{m,r}.
 \end{aligned}$$

□

Definición de la ecuación de onda

Comenzaremos dando una introducción a la ecuación de onda en \mathbb{R}^n , con $n \geq 1$, siguiendo [10, 11]. Para simplificar la notación denotaremos a la función $u(x, t)$, como u , ahora la ecuación general de la ecuación de onda será,

$$u_{tt} - c^2 \Delta u = f, \quad \Omega \times [0, +\infty), \quad (3.1)$$

donde $\Omega \subset \mathbb{R}^n$, f es una función de valores reales, $c > 0$ es una constante de propagación y el operador

$$\Delta u = \sum_{j=1}^N \frac{\partial^2 u}{\partial x_j^2}, \quad (3.2)$$

es conocido como el laplaciano. En esta ecuación la variable t denota el tiempo transcurrido, x_i , $i = 1, \dots, n$ denotan las variables espaciales en cada dirección de \mathbb{R}^n y $u(x, t)$ será la posición la posición de la onda en cada punto del dominio espacio-temporal.

Podemos observar que se trata de una ecuación lineal y, siempre que $f \not\equiv 0$, la ecuación será no homogénea, es decir, tendrá termino independiente no nulo.

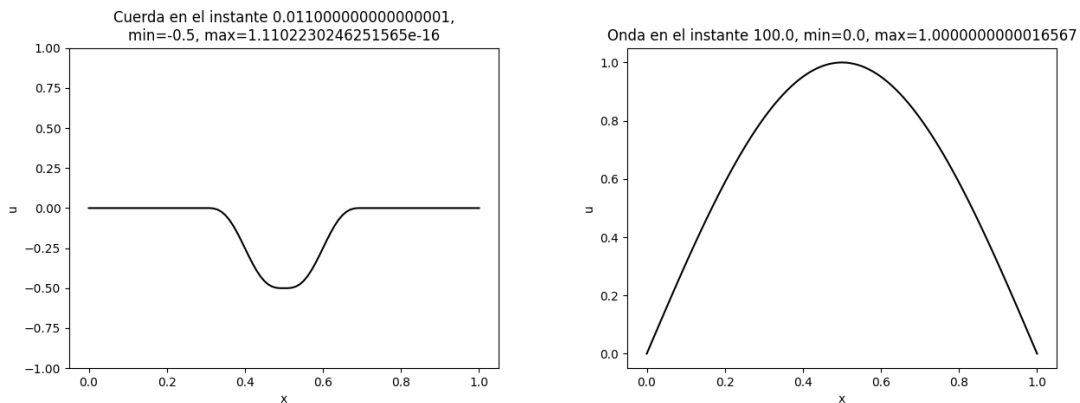
3. DEFINICIÓN DE LA ECUACIÓN DE ONDA

3.1 La ecuación de onda unidimensional

Ya que habiéndonos situado en la ecuación de onda, a partir de esta sección nos centraremos en el caso unidimensional, es decir en el que dominio $\Omega \subset \mathbb{R}$. Comenzaremos explicando el modelado de dicho fenómeno, apoyándonos en [1, 12].

Supongamos que tenemos una cuerda uniforme y tensa, sujeta por los extremos, tal como lo estaría, por ejemplo, en una guitarra. Esta cuerda estaría colocada en nuestro esquema en el eje X entre 0 y L , siendo L el largo de la misma. Así vemos que cada punto de la cuerda se desplazará en el tiempo. Por lo tanto, la función que describa el movimiento de la cuerda será del tipo $u(x, t)$.

Nuestra cuerda en un instante t podría tener una forma parecida a alguna de las siguientes:



Supongamos que la cuerda no está en reposo, habrá una fuerza tensora T , producida por los puntos en los que está sujeta. Nos apoyaremos en la siguiente figura para ver como se describiría la ecuación:

3.1 La ecuación de onda unidimensional

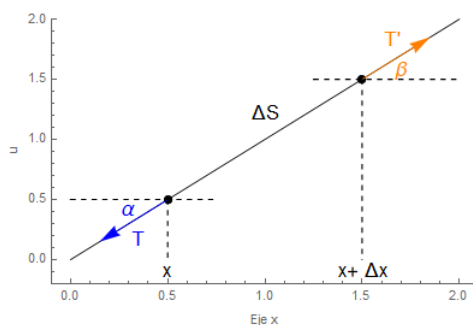


Figura 3.1: Representación de una sección diferencial de la cuerda y las fuerzas asociadas a la misma.

Elegido un segmento diferencial del arco de la cuerda ΔS , nos damos cuenta de que la cuerda se habrá desplazado perpendicularmente una cantidad u . Por lo tanto la fuerza tensora que produce cada punto en el que está fijado, no tendrán la misma dirección y sentido opuesto, sino que tendrán componente vertical. Las componentes verticales son:

$$T_y = -F_T \sin(\alpha),$$

$$T'_y = F_T \sin(\beta).$$

Luego la fuerza vertical quedaría $F = F_T(\sin(\beta) - \sin(\alpha))$, donde F_T es la fuerza generada por la tensión, por otro lado, al encontrarnos en una sección diferencial los ángulos son tan pequeños que podemos decir $\sin(\alpha) \approx \tan(\alpha)$ y $\sin(\beta) \approx \tan(\beta)$ quedando

$$F = F_T(\tan(\beta) - \tan(\alpha)).$$

Además $\tan(\alpha)$ y $\tan(\beta)$ son las pendientes de la curva en los puntos x y $x + \Delta x$ respectivamente, entonces podemos expresarlos como la primera derivada en los puntos correspondientes $\tan(\alpha) = u_x(x, t)$ y $\tan(\beta) = u_x(x + \Delta x, t)$. Obtenemos $F = F_T(u_x(x + \Delta x, t) - u_x(x, t))$.

Aplicando ahora la segunda Ley de Newton, la fuerza debe ser la masa por la aceleración. Luego será la masa $\rho_l \Delta S$, donde ρ_l es la densidad lineal de masa de la cuerda y como estamos en un ambiente diferencial podemos aproximar $\Delta S \approx \Delta x$, y la aceleración $u_{tt}(x, t)$. Reestructurando la ecuación queda:

3. DEFINICIÓN DE LA ECUACIÓN DE ONDA

$$u_{tt}(x, t) = \frac{F_T}{\rho_l} \frac{u_x(x + \Delta x, t) - u_x(x, t)}{\Delta x}.$$

Tomando limite cuando $\Delta x \rightarrow 0$ y $c^2 = \frac{F_T}{\rho_l}$ la ecuación queda de la siguiente forma:

$$u_{tt} = c^2 u_{xx}, \quad (0, L) \times [0, T]. \quad (3.3)$$

En este caso podemos tomamos el dominio como $\Omega\mathbb{R}$, donde T es un parámetro dado que representa el instante final de la observación. Añadiendo las condiciones de contorno usuales (valores conocidos de la solución sobre la frontera del dominio, $x = 0$ y $x = L$), para más información ver [13], obtendremos el siguiente modelo.

$$u_{tt} - c^2 u_{xx} = 0, \quad (x, t) \in (0, L) \times (0, T), \quad (3.4)$$

$$u(x, 0) = f(x), \quad x \in (0, L), \quad (3.5)$$

$$u_t(x, 0) = g(x), \quad x \in (0, L), \quad (3.6)$$

$$u(0, t) = u(L, t) = 0, \quad t \in (0, T). \quad (3.7)$$

Estas última condiciones de contorno, (3.7), se llaman de tipo Dirichlet homogéneas. En lo que sigue, supondremos que f y g son funciones suficientemente regulares para que el problema anterior esté bien planteado y tenga solución, es decir, exista una función u dos veces derivable con respecto a t y x que verifica (3.4) a (3.7).

La solución de este esquema será única, la demostración de la unicidad se puede encontrar en [14].

3.2 Clasificación de la ecuación de onda

En esta sección clasificaremos las EDP lineales, estas se clasifican en: elípticas, hiperbólicas y parabólicas. Para poder realizar la clasificación primero tendremos que estandarizarla. Aunque las ecuaciones se pueden clasificar y estandarizar siempre, nosotros por simplicidad lo haremos solo para las de segundo orden.

Una ecuación lineal de segundo orden en forma estándar sería como sigue:

$$Au_{tt} + Bu_{tx} + Cu_{xx} + Du_t + Eu_x + Fu = G, \quad (3.8)$$

donde A, B, C, D, E, F, G son funciones de variables x y t .

Clasificaremos (3.8) dependiendo de las ecuaciones características relativas a la ecuación en derivadas parciales (EDP):

DEFINICIÓN 3.2.1: Clasificación de las EDP de segundo orden

Si en todos los puntos (x, t) de una región $W \subset \mathbb{R}^2$ se cumple que:

- $B^2(x, t) - 4A(x, t)C(x, t) > 0$, entonces la EDP (3.8) se dice hiperbólica.
- $B^2(x, t) - 4A(x, t)C(x, t) = 0$, entonces la EDP (3.8) se dice parabólica.
- $B^2(x, t) - 4A(x, t)C(x, t) < 0$, entonces la EDP (3.8) se dice elíptica.

Vemos que siguiendo la notación de (3.3), $A = 1, C = -c^2, G = f$ y cada una de las demás funciones es 0. En el caso de la ecuación de onda, tenemos $B^2 - 4AC = 4c^2 > 0$ y por lo tanto nuestra ecuación es hiperbólica.

Este tipo de clasificación de las ecuaciones la usaremos para separar los distintos métodos numéricos que se pueden usar, ya que estos suelen diferir dependiendo de la misma.

3.3 Soluciones de tipo onda viajera

Una vez definida nuestra ecuación de onda, es interesante ver que esta admite dos soluciones de tipo onda viajera para cada uno de los dos sentidos del eje X . Para ello comenzaremos suponiendo que la solución $u(x, t)$ se puede descomponer en dos funciones de propagación, donde c sabemos que es la velocidad de la onda,

$$u(x, t) = p_+(x - ct) + p_-(x + ct), \quad (3.9)$$

aquí llamamos p_+ a la propagación hacia la derecha y p_- hacia la izquierda. Ahora siguiendo [15], veamos como sería la resolución del problema de valores iniciales, (3.4) a (3.7) teniendo en cuenta (3.9). Para ello necesitaremos suponer la derivabilidad de f .

Comenzamos utilizando (3.5), para $t = 0$ tendremos,

$$f(x) = u(x, 0) = p_+(x) + p_-(x), \quad (3.10)$$

3. DEFINICIÓN DE LA ECUACIÓN DE ONDA

derivando (3.10) en x ,

$$f'(x) = p_+(x) + p_-(x). \quad (3.11)$$

Ahora queremos sustituir el valor de $u_t(x, 0)$ en la condición (3.6), para ello derivemos antes (3.9) respecto a t ,

$$u_t = \frac{\partial}{\partial t}(p_+(x - ct) + p_-(x + ct)) = cp'_+(x - ct) - cp'_-(x + ct), \quad (3.12)$$

entonces,

$$g(x) = u_t(x, 0) = -cp_+(x) + cp_-(x), \quad (3.13)$$

dividiendo la igualdad por c ,

$$\frac{1}{c}g(x) = -p_+(x) + p_-(x). \quad (3.14)$$

Haciendo (3.10) + (3.13) y (3.10) - (3.13):

$$f'(x) + \frac{1}{c}g(x) = 2p'_-(x), \quad \text{y} \quad f'(x) - \frac{1}{c}g(x) = 2p'_+(x).$$

Luego despejando,

$$p'_-(x) = \frac{1}{2}f'(x) + \frac{1}{2c}g(x), \quad \text{y} \quad p'_+(x) = \frac{1}{2}f'(x) - \frac{1}{2c}g(x).$$

Como todo este proceso podría hacerse para todo $x \in \mathbb{R}$, integrando respecto a x y sustituyéndolo por $x + ct$ y $x - ct$, respectivamente, obtendremos las representaciones siguientes de p_- y p_+ .

$$p_-(x + ct) = \frac{1}{2}f(x + ct) + \frac{1}{2c} \int_0^{x+ct} g(s)ds + A, \quad (3.15)$$

$$p_+(x + ct) = \frac{1}{2}f(x - ct) - \frac{1}{2c} \int_0^{x-ct} g(s)ds + B, \quad (3.16)$$

donde A, B son constantes que tomaremos como 0 al saber que $A + B = 0$ gracias a (3.9). Nuestras integrales comienzan en 0 por el dominio de (3.4).

Finalmente llegamos a que la solución del problema de contorno, será:

$$u(x, t) = \frac{1}{2}(f(x + ct) + f(x - ct)) + \frac{1}{2c} \int_{x-ct}^{x+ct} g(s)ds. \quad (3.17)$$

3.4 La energía en la ecuación de onda

En esta sección nos centraremos en analizar la ley de energía en la ecuación de onda. Para poder obtener la energía de la ecuación haremos para todo $t \in [0, T]$ el producto interior 2.2.1 de la misma con u_t :

$$\langle u_t(\cdot, t), u_{tt}(\cdot, t) \rangle_{L^2(\Omega)} = c^2 \langle u_t(\cdot, t), u_{xx}(\cdot, t) \rangle_{L^2(\Omega)}.$$

Para facilitar el entendimiento de la sección simplificaremos la notación de la siguiente forma:

$$\langle u_t, u_{tt} \rangle_{L^2(\Omega)} = c^2 \langle u_t, u_{xx} \rangle_{L^2(\Omega)}. \quad (3.18)$$

Cada uno de los términos de la energía tendremos que ponerlo de la forma $\frac{\partial}{\partial t}(\cdot)$, comenzaremos por el término de la derecha. Para desarrollarlo necesitaremos la siguiente igualdad, derivada de la integración por partes,

$$\langle u_t, u_{tt} \rangle_{L^2(\Omega)} = \int_0^L u_t u_{xx} dx = u_t u_x \Big|_0^L - \int_0^L u_{tx} u_x dx. \quad (3.19)$$

Introduciendolo, en la ecuación (3.18),

$$\begin{aligned} 0 &= \langle u_t, u_{tt} \rangle_{L^2(\Omega)} - c^2 \langle u_t, u_{xx} \rangle_{L^2(\Omega)} \\ &= \int_0^L u_{tt} u_t dx - c^2 \left(u_t u_x \Big|_0^L - \int_0^L u_{tx} u_x dx \right) \\ &= \int_0^L u_{tt} u_t dx + c^2 \left(\int_0^L u_{tx} u_x dx - u_t u_x \Big|_0^L \right) \end{aligned} \quad (3.20)$$

El término $u_t u_x \Big|_0^L$ se anula para nuestro esquema, gracias a la condición (3.7). Esto es:

$$u(0, t) = u(L, t) = 0, \text{ para } t \in (0, T).$$

Por lo tanto derivando en t ,

$$u_t(0, t) = u_t(L, t) = 0, \text{ para } t \in (0, T).$$

Luego,

$$u_t u_x \Big|_0^L = u_t(\cdot, L) u_x(\cdot, L) - u_t(\cdot, 0) u_x(\cdot, 0) = 0$$

Continuando con (3.20), utilizando la regla de la cadena obtenemos:

3. DEFINICIÓN DE LA ECUACIÓN DE ONDA

$$\begin{aligned} 0 &= \int_0^L u_t u_{tt} dx + c^2 \int_0^L u_x u_{tx} dx \\ &= \int_0^L \frac{1}{2} \frac{\partial}{\partial t} u_t^2 dx + c^2 \int_0^L \frac{1}{2} \frac{\partial}{\partial t} u_x^2 dx \\ &= \frac{1}{2} \frac{\partial}{\partial t} \int_0^L u_t^2 dx + c^2 \frac{1}{2} \frac{\partial}{\partial t} \int_0^L u_x^2 dx \\ &= \frac{\partial}{\partial t} \left(\frac{1}{2} \|u_t\|_{L^2(\Omega)}^2 + \frac{c^2}{2} \|u_x\|_{L^2(\Omega)}^2 \right) \end{aligned}$$

Definiendo de la siguiente forma,

$$E(u) = \frac{1}{2} \|u_t\|_{L^2(\Omega)}^2 + \frac{c^2}{2} \|u_x\|_{L^2(\Omega)}^2,$$

donde E es la energía, que depende de la solución u , vemos que,

$$\frac{\partial}{\partial t} E(u) = 0.$$

Por lo tanto para nuestro esquema se da la ley de la conservación de la energía.

Cabe recalcar que si en lugar de las condiciones de tipo Dirichlet impuestas en (3.7) se diesen unas condiciones de tipo Neumann homogéneas en dichos puntos, es decir

$$u_x(0, t) = u_x(L, t) = 0, \text{ para } t \in (0, T),$$

tambien se cumpliría dicha afirmación.

Resolución numérica de la ecuación de onda

4.1 Métodos principales de resolución

Las ecuaciones en derivadas parciales, en general, no se pueden resolver con métodos exactos, como separación de variables, transformada de Fourier, etc. Con la excepción de ecuaciones muy sencillas sobre dominios espaciales no acotados o muy estructurados, como intervalos, rectángulos o esferas, que en la práctica pueden no resultar útiles para capturar todos los matices de la realidad. Esto nos lleva a la creación de métodos numéricos que consiguen aproximar sus soluciones, pero con errores tan pequeños como sea preciso siempre que podamos asumir el coste computacional.

En este apartado nos centraremos en el método de diferencias finitas, y veremos las características principales que nos atañen del mismo. Más tarde, en el capítulo 5, implementaremos este método en Python.

4.2 Método de diferencias finitas

Este método consiste principalmente en el uso de cocientes incrementales para la aproximación en tiempo y en espacio de nuestro problema de contorno con valores iniciales. En nuestro caso, usaremos dicho método para aproximar la solución de la ecuación de onda

4. RESOLUCIÓN NUMÉRICA DE LA ECUACIÓN DE ONDA

(3.4)-(3.7).

Primero, vamos a crear un mallado tal y como en 2.2.2, para hacer una discretización del problema. Definimos un entero m y el tamaño de paso en espacio correspondiente $h = \frac{L}{m}$, tales que:

$$x_j = jh \quad \text{para } j = 0, \dots, m,$$

y $x_0 = 0, x_m = L$, es decir es una partición de nuestro dominio espacial $(0, L)$.

Por otro lado, vamos a crear una partición para el dominio temporal. En este caso, a partir de los parámetros que obtendremos en la sección 4.5, decidiremos el valor de k , el tamaño de paso temporal. A partir de este definiremos $n \in \mathbb{N}$ tal que,

$$t_r = rk \quad \text{para } r = 0, 1, \dots, n,$$

con $t_0 = 0, t_n = T$, partición del dominio temporal $(0, T)$.

Con las particiones ya definidas, haremos un mallado del tipo (x_j, t_r) .

La ecuación evaluada en los puntos del mallado (x_j, t_r) , que se encuentran en el interior del dominio espacio temporal podrían escribirse como:

$$u_{tt}(x_j, t_r) = c^2 u_{xx}(x_j, t_r). \quad (4.1)$$

En lo que sigue, nos propondremos calcular un conjunto de valores, denotados $u_{j,r} \in \mathbb{R}$ que aproximen la solución de la igualdad anterior de forma que, $u(x_j, t_r) \approx u_{j,r}$. Para ello, el método se construirá optando por un esquema del tipo explícito en tiempo, utilizando una aproximación centrada tanto en espacio como en tiempo.

$$u_{xx}(x_j, t_r) \approx \frac{u_{j+1,r} - 2u_{j,r} + u_{j-1,r}}{h^2}, \quad (4.2)$$

$$u_{tt}(x_j, t_r) \approx \frac{u_{j,r+1} - 2u_{j,r} + u_{j,r-1}}{k^2}. \quad (4.3)$$

Sería posible utilizar diversos esquemas diferentes, se puede obtener más información en [16].

Ahora sustituyendo las expresiones (4.2),(4.3) en la ecuación de onda (4.1) nos queda el esquema.

$$\frac{u_{j,r+1} - 2u_{j,r} + u_{j,r-1}}{k^2} = c^2 \frac{u_{j+1,r} - 2u_{j,r} + u_{j-1,r}}{h^2}. \quad (4.4)$$

Es usual definir $\mu = \frac{ck}{h}$, con lo que se tendría:

$$u_{j,r+1} - 2u_{j,r} + u_{j,r-1} = \mu^2 u_{j+1,r} - 2\mu^2 u_{j,r} + \mu u_{j-1,r},$$

$$\text{para } r = 1, 2, \dots, n, \quad j = 1, 2, \dots, m-1.$$

Despejando la aproximación con mayor nivel en tiempo, obtenemos:

$$u_{j,r+1} = \mu^2(u_{j+1,r} + u_{j-1,r}) + 2(1 - \mu^2)u_{j,r} - u_{j,r-1}. \quad (4.5)$$

Esta expresión define el método que utilizaremos para aproximar la solución de la EDP en el interior de nuestro dominio, omitiendo, por ahora, el tratamiento de las condiciones iniciales (3.5) y (3.6) y de contorno (3.7).

A continuación nos centraremos en dichas condiciones, siguiendo [15]. En primer lugar la condición inicial (3.5) nos dará los valores de los nodos

$$u_{j,0} = f(x_j), \quad \text{para } j = 0, 1, \dots, m.$$

Por otra parte, las condiciones de contorno de tipo Dirichlet (3.7) quedarán como $u_{0,r} = u_{m,r} = 0$ para $r = 0, 1, \dots, n$.

Para la condición que falta (3.6) utilizaremos, haciendo una aproximación por cociente incremental en los dos primeros instantes, $u_{j,1} = u_{j,0} + kg(x_j)$. Para esta podría haber muchas mas opciones. En nuestro caso no ahondaremos mas en este tema y utilizaremos la expresión mas sencilla descrita anteriormente.

Nuestro esquema al completo quedaría:

$$u_{j,r+1} = \mu^2(u_{j+1,r} + u_{j-1,r}) + 2(1 - \mu^2)u_{j,r} - u_{j,r-1}, \quad (4.6)$$

$$\text{para } j = 1, \dots, m-1 \text{ y } r = 1, \dots, n-1,$$

$$u_{j,0} = f(x_j), \quad j = 0, 1, \dots, m, \quad (4.7)$$

$$u_{j,1} = u_{j,0} + kg(x_j), \quad j = 0, 1, \dots, m, \quad (4.8)$$

$$u_{0,r} = u_{m,r} = 0, \quad r = 0, 1, \dots, n. \quad (4.9)$$

4. RESOLUCIÓN NUMÉRICA DE LA ECUACIÓN DE ONDA

4.3 Dominio de dependencia

Para el esquema que acabamos de definir intentaremos encontrar las condiciones, en las que nuestra solución se aproxime más fielmente a la realidad. El pensamiento inicial puede ser que a más pequeñoselijamos h y k mejores resultados obtendremos, pero esto no es cierto en general. Por ello, se utilizan diversos métodos de análisis de estabilidad y convergencia. En esta sección nos centraremos en un concepto altamente relacionado con la estabilidad y energía del modelo (3.4)-(3.7), tal y como se puede ver en [17], el dominio de dependencia.

DEFINICIÓN 4.3.1: Dominio de dependencia

Sea una función $u(x, t)$ y un conjunto $[a, b] \times [c, d]$. El dominio de dependencia de u en un punto arbitrario (x_1, t_1) , es el conjunto de puntos de $[a, b]$ necesarios para dar la solución de u en (x_1, t_1) .

Como vemos, este concepto no solo estará definido para entornos discretos como lo que hemos estado tratando hasta ahora. Véase que para cada punto del mallado ?? (x_j, t_r) tendremos un dominio de dependencia asociado.

Es importante remarcar que el dominio de dependencia no es fácilmente calculable, de hecho en muchos casos no se obtiene en la práctica, pero es una herramienta muy útil para comprender las situaciones que se pueden dar en nuestro esquema. Nuestro objetivo será que los puntos del mallado que hemos utilizado para aproximar la solución, contengan íntegramente al dominio de dependencia.

Lo ilustramos con las siguiente figuras, utilizando como ejemplo nuestro esquema concreto.

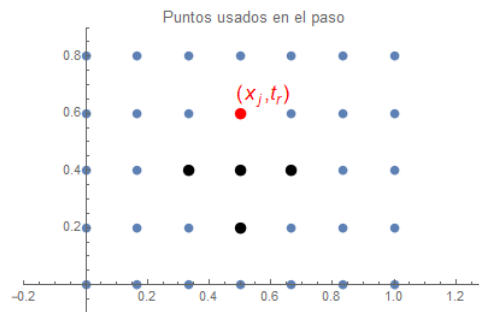


Figura 4.1: Puntos que usamos para aproximar $u(x_j, t_r)$.

4.3 Dominio de dependencia

A su vez para aproximar la solución de cada uno de los puntos coloreados de negro, necesitaremos de los anteriores, por lo tanto:

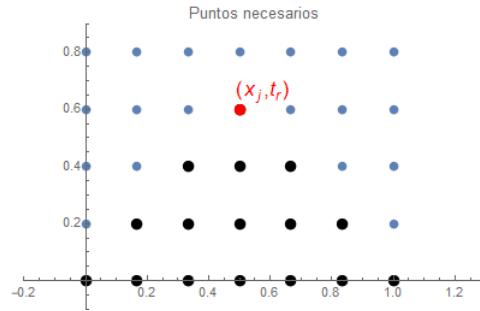


Figura 4.2: Todos los puntos que se usan para aproximar la solución de (x_j, t_r) .

Siendo la recta coloreada e azul, \overline{QR} , se podrían dar dos casos
Que el dominio de dependencia quedase en el interior de los puntos de $[a, b]$ utilizados en el esquema en diferencias finitas (Coloreados de verde):

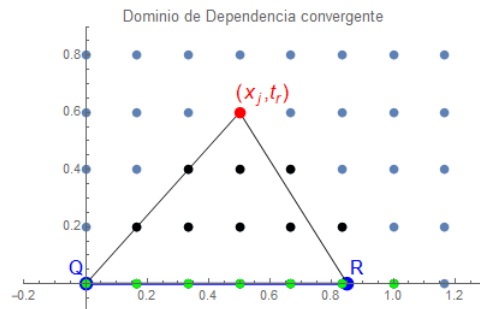


Figura 4.3: En este caso si hay un cambio en cualquier punto de la recta \overline{QR} , los puntos verdes lo captarían, al contener íntegramente a la misma.

Que el dominio de dependencia quedase fuera de los puntos $[a, b]$ utilizados (verdes):

4. RESOLUCIÓN NUMÉRICA DE LA ECUACIÓN DE ONDA

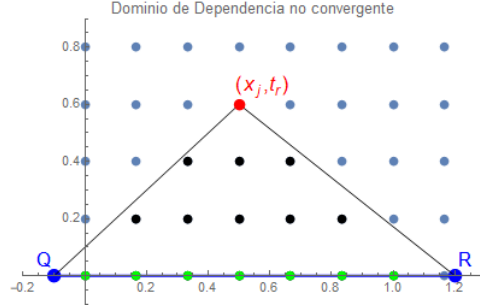


Figura 4.4: En este caso, si se da un cambio en la zona de la recta azul exterior a los puntos verdes, estos no lo captarían y por lo tanto la solución no variaría.

La situación más favorable, y por tanto la que busquemos, para nuestro esquema será la primera. En ese caso, al menos no tendremos el tipo de errores comentados en la Figura 4.4.

4.4 Consistencia

Para poder comprobar el orden del esquema que hemos elegido (4.5) vamos a realizar un análisis de la consistencia. Comenzaremos sustituyendo en la EDP nuestro esquema, y separando la ecuación (4.4) de la siguiente forma,

$$\underbrace{\frac{u(j, r+1) - 2u(j, r) + u(j, r-1))}{k^2}}_{(1)} - c^2 \underbrace{\frac{u(j+1, r) - 2u(j, r) + u(j-1, r))}{h^2}}_{(2)} = 0. \quad (4.10)$$

Haremos ahora el desarrollo de Taylor para cada una de las componentes anteriores, denotando para las derivadas a partir de orden dos $u_{xxx} = u_{3x}$ y $u_{ttt} = u_{3t}$, respectivamente. Suponiendo que nuestra solución u sea suficientemente regular, es decir, que se pueda derivar tantas veces como sea necesario, y denotando por brevedad $u = u(x_j, t-r)$, obtenemos.

$$\begin{aligned} u(x_{j+1}, t_r) &= u + hu_x + h^2 \frac{u_{xx}}{2} + h^3 \frac{u_{3x}}{3!} + h^4 \frac{u_{4x}}{4!} + h^5 \frac{u_{5x}}{5!} + O(h^6), \\ u(x_{j-1}, t_r) &= u - hu_x + h^2 \frac{u_{xx}}{2} - h^3 \frac{u_{3x}}{3!} + h^4 \frac{u_{4x}}{4!} - h^5 \frac{u_{5x}}{5!} + O(h^6), \\ u(x_j, t_{r+1}) &= u + ku_t + k^2 \frac{u_{tt}}{2} + k^3 \frac{u_{3t}}{3!} + k^4 \frac{u_{4t}}{4!} + k^5 \frac{u_{5t}}{5!} + O(k^6), \\ u(x_j, t_{r-1}) &= u - ku_t + k^2 \frac{u_{tt}}{2} - k^3 \frac{u_{3t}}{3!} + k^4 \frac{u_{4t}}{4!} - k^5 \frac{u_{5t}}{5!} + O(k^6), \\ u(x_j, t_r) &= u. \end{aligned}$$

Sustituyendo en (1) y (2), obtendremos:

$$(1) = \frac{u(j, r+1) - 2u(j, r) + u(j, r-1)}{k^2} = u_{tt} + 2\frac{k^2 u_{4t}}{4!} + 2\frac{k^4 u_{6t}}{6!} + O(h^6),$$

$$(2) = \frac{u(j+1, r) - 2u(j, r) + u(j-1, r)}{h^2} = u_{xx} + 2\frac{h^2 u_{4x}}{4!} + 2\frac{h^4 u_{6x}}{6!} + O(k^6).$$

Volviendo a (4.10), obtenemos:

$$\begin{aligned} T_{h,k}u &:= u_{tt} + 2\frac{k^2 u_{4t}}{4!} + 2\frac{k^4 u_{6t}}{6!} - c^2 u_{xx} - 2c^2 \frac{h^2 u_{4x}}{4!} - 2c^2 \frac{h^4 u_{6x}}{6!} \dots = \\ &= u_{tt} + 2\frac{k^2 u_{4t}}{4!} - c^2 u_{xx} - 2c^2 \frac{h^2 u_{4x}}{4!} + O(k^4) + O(h^4). \end{aligned}$$

Como estamos evaluando la solución exacta en cada punto tendremos $u_{tt} = c^2 u_{xx}$, entonces

$$\begin{aligned} u_{3t} &= c^2 u_{xxt}, \\ u_{4t} &= c^2 u_{xxtt}. \end{aligned}$$

Por otro lado,

$$\begin{aligned} c^2 u_{3x} &= u_{ttx}, \\ c^2 u_{4x} &= u_{ttxx}. \end{aligned}$$

De aquí deducimos, $u_{4x} = \frac{u_{4t}}{c^4}$ y por tanto:

$$\begin{aligned} T_{h,k}u &= u_{tt} + 2\frac{k^2 u_{4t}}{4!} - c^2 u_{xx} - 2c^2 \frac{h^2 u_{4x}}{4!} + O(k^4) + O(h^4) = \\ &= 2\frac{k^2 u_{4t}}{4!} - 2c^2 \frac{h^2 u_{4t}}{4! c^4} + O(k^4) + O(h^4) = \\ &= \frac{2}{4!} (k^2 - \frac{h^2}{c^2}) u_{4t} + O(k^4) + O(h^4) = \\ &= O(k^2) + O(h^2). \end{aligned}$$

Así hemos comprobado que el esquema tiene orden de consistencia dos en espacio y en tiempo.

4. RESOLUCIÓN NUMÉRICA DE LA ECUACIÓN DE ONDA

4.5 Análisis de Von Neumann

En la utilización de métodos multipaso, como el esquema en el que nos centramos en este trabajo, los errores globales finalmente observados pueden ser mucho mayores de lo que podría esperarse a partir del orden de consistencia local estudiado anteriormente. Esto es debido a que en el error global interviene un aumento del error que se propaga y que, a menudo, incluso se incrementa al disminuir la longitud de paso temporal. Dicho error que se propaga, puede estar relacionado con varios factores, como el método de inicialización, el truncamiento local y errores de redondeo. Esto se conoce como **inestabilidad**.

En esta sección nos centraremos en el análisis de Estabilidad de Fourier o Von Neumann. Este aunque no garantizará la estabilidad total del esquema, suele dar muy buenos resultados en la práctica, y es el más empleado.

Podemos encontrar diversas notaciones y formas de hacer este análisis en libros clásicos y modernos, véase [16, 18, 19].

Comenzamos relacionando cada solución numérica del esquema con la solución exacta y el error del mismo, de la forma:

$$u_{j,r} = u(x_j, t_r) + \varepsilon_{j,r} \quad (4.11)$$

donde $\varepsilon_{j,r}$ denotará el error en cada uno de los puntos del retículo que hemos formado.

Ahora vamos a aplicarlo a la ecuación de onda, para profundizar en la estabilidad del esquema (4.6). Haciendo uso del término del error como hemos visto en (4.11), lo sustituimos en el esquema, esta vez utilizando (4.4).

$$\left(\frac{u(x_j, t_{r+1}) - 2u(x_j, t_r) + u(x_j, t_{r-1}))}{k^2} - c^2 \frac{u(x_{j+1}, t_r) - 2u(x_j, t_r) + u(x_{j-1}, t_r)}{h^2} \right) + \left(\frac{\varepsilon_{j,r+1} - 2\varepsilon_{j,r} + \varepsilon_{j,r-1}}{k^2} - c^2 \frac{\varepsilon_{j+1,r} - 2\varepsilon_{j,r} + \varepsilon_{j-1,r}}{h^2} \right) = 0.$$

Como hemos observado que se da la convergencia del esquema para la solución exacta, la ecuación de propagación de los errores será:

$$\frac{\varepsilon_{j,r+1} - 2\varepsilon_{j,r} + \varepsilon_{j,r-1}}{k^2} - c^2 \frac{\varepsilon_{j+1,r} - 2\varepsilon_{j,r} + \varepsilon_{j-1,r}}{h^2} = 0.$$

Tomando la solución de tipo onda $\varepsilon_{j,r} = g^r e^{ij\beta h}$ donde i es la unidad imaginaria, g denota el factor de amplitud y β se corresponderá con el número de onda de la ecuación de Fourier. Vamos a sustituirlo en la ecuación anterior, quedando:

$$g^{r+1} e^{ij\beta h} = \mu^2 (g^r e^{i(j+1)\beta h} + g^r e^{i(j-1)\beta h}) + 2(1 - \mu^2) g^r e^{ij\beta h} - g^{r-1} e^{ij\beta h}.$$

Dividiendo entre $g^{r-1} e^{ij\beta h}$:

$$\begin{aligned} g^2 &= \mu^2 (g e^{i\beta h} + g e^{-i\beta h}) + 2(1 - \mu^2) g - 1, \\ 0 &= g^2 - \mu^2 (g e^{i\beta h} + g e^{-i\beta h}) - 2(1 - \mu^2) g + 1. \end{aligned}$$

Usando las identidades trigonométricas $e^{ix} + e^{-ix} = 2\cos(x)$ y $1 - \cos(x) = 2\sin^2(\frac{x}{2})$ obtenemos,

$$\begin{aligned} 0 &= g^2 - 2\mu^2 \cos(\beta h) g - 2(1 - \mu^2) g + 1, \\ &= g^2 - 2\mu^2 \cos(\beta h) g + 2\mu^2 g - 2g + 1, \\ &= g^2 + 2g(1 + \mu^2(1 - \cos(\beta h))) + 1, \\ &= g^2 + 2 \left(2\mu^2 \sin^2 \left(\frac{\beta h}{2} \right) - 1 \right) g + 1. \end{aligned}$$

Podemos ahora tratarlo como una ecuación de segundo grado, es decir, tomando $A = \left(1 - 2\mu^2 \sin^2 \left(\frac{\beta h}{2} \right) \right)$ vemos que nos quedaría,

$$\begin{aligned} g &= \frac{2A \pm \sqrt{(-2A)^2 - 4}}{2} = A \pm \sqrt{A^2 - 1} = \\ &= 1 - 2\mu^2 \sin^2 \left(\frac{\beta h}{2} \right) \pm \sqrt{\left(1 - 2\mu^2 \sin^2 \left(\frac{\beta h}{2} \right) \right)^2 - 1}. \end{aligned}$$

Se puede comprobar que si $r \rightarrow \infty$ entonces $|\varepsilon_{j,r}| = |g|^r$ tiende a infinito si y solo si $|g| > 1$, por lo tanto vamos a estudiar cuando se dará $|g| \leq 1$

$$\left| A \pm \sqrt{A^2 - 1} \right| \leq 1.$$

Vamos a ver que pasará dependiendo de los valores posibles de A . Como $2\mu^2 \sin^2 \left(\frac{\beta h}{2} \right) \geq 0$ tendremos que $A \leq 1$.

4. RESOLUCIÓN NUMÉRICA DE LA ECUACIÓN DE ONDA

- Si $A < -1$:

La solución $g = A - \sqrt{A^2 - 1} < -1$ y por lo tanto siempre se da $|g| > 1$ y el esquema será en cualquier caso inestable.

- Si $-1 \leq A \leq 1$:

En este caso se tendrá que g es un número complejo, ya que $-1 \leq A^2 - 1 \leq 0$.

$$g = A \pm \sqrt{A^2 - 1} = A \pm \sqrt{1 - A^2}i.$$

Por lo tanto, tomando módulo, ambas soluciones quedarían,

$$|g| = \sqrt{A^2 + \sqrt{1 - A^2}^2} = A^2 + 1 - A^2 = 1,$$

luego nuestro esquema es condicionalmente estable.

Veamos que condición se debe imponer en μ para que se tenga

$$-1 \leq A \leq 1$$

, y por tanto el esquema sea estable. Definimos $s^2 \triangleq \sin^2\left(\frac{\beta h}{2}\right)$, con esto tenemos

$$0 \leq s^2 \leq 1,$$

luego,

$$0 \leq 2\mu^2 s^2 \leq 2\mu^2,$$

$$0 \geq -2\mu^2 s^2 \geq -2\mu^2,$$

$$1 \geq 1 - 2\mu^2 s^2 \geq 1 - 2\mu^2$$

Ahora solo necesitamos que $1 - 2\mu^2 \geq -1$, para lo cual debe ser

$$2\mu^2 \leq 2,$$

$$\mu \leq 1, \tag{4.12}$$

llegamos a que para que el esquema sea estable necesitamos que se de esta última condición $\mu \leq 1$, la condición CFL.

Volviendo al concepto de dominio de dependencia 4.3.1, tal y como se extrae de [11], cuanto más nos acerquemos a la igualdad ($\mu = 1$), cubriremos de manera mas óptima el

4.6 Resolución numérica para las soluciones de tipo viajero

dominio de dependencia con nuestra aproximación. Siempre respetando la condición CFL.

En el capítulo 5 mostraremos una comparación entre las diferentes aproximaciones obtenidas al variar μ , para ilustrar como afecta dicho valor a la estabilidad de las aproximaciones tal y como indica el análisis anteriormente realizado.

4.6 Resolución numérica para las soluciones de tipo viajero

Tal y como comentamos en la sección 3.3 de la ecuación de onda se pueden obtener dos soluciones de tipo onda viajera (3.9). Ahora nos centraremos en implementar numéricamente dichas soluciones, esta técnica podría ser necesaria en caso de que no sea sencillo integrar g de forma exacta, tal y como haríamos en (3.17)

Querremos ver dichas soluciones en cada uno de nuestro puntos del mallado, para ello llamaremos a las aproximaciones de las mismas en cada uno como:

$$\begin{aligned}w^+(x_j, t_r) &= p_+(x_j - ct_r), \\w^-(x_j, t_r) &= p_-(x_j + ct_r).\end{aligned}$$

Ahora tendremos que la aproximación de la solución de u será,

$$u_{j,r} = w_{j,r}^+ + w_{j,r}^-. \quad (4.13)$$

Partiendo de las representaciones de las soluciones (3.15) y (3.16), aproximaremos $w_{j,r}^+$ y $w_{j,r}^-$. Esto lo haremos primeramente separando en dos la integral, trabajemos con $p_-(x_j + ct_r)$:

$$\begin{aligned}p_-(x_j + ct_r) &= \frac{1}{2}f(x_j + ct_r) + \frac{1}{2c} \int_0^{x_j + ct_r} g(s) ds = \\&= \frac{1}{2}f(x_j + ct_r) + \frac{1}{2c} \int_0^{x_j} g(s) ds + \frac{1}{2c} \int_{x_j}^{x_j + ct_r} g(s) ds.\end{aligned}$$

Realizamos una aproximación de cada una de las integrales utilizando el método del trapecio y los puntos ya definidos en el mallado. El tamaño de paso en la primera integral

4. RESOLUCIÓN NUMÉRICA DE LA ECUACIÓN DE ONDA

será h , y en la segunda será ck , por tanto la aproximación es.

$$w_{j,r}^- = \frac{1}{2}f(x_j + ct_r) + \frac{h}{4c} \left(g(x_0) + 2 \sum_{i=1}^{j-1} g(x_i) + g(x_j) \right) \\ + \frac{ck}{4c} \left(g(x_j) + 2 \sum_{i=1}^{r-1} g(x_j + ct_i) + g(x_j + ct_r) \right).$$

Por otro lado trabajando con $p_+(x_j + ct_r)$, y dividiendo la integral ahora de la siguiente forma,

$$p_+(x_j - ct_r) = \frac{1}{2}f(x_j + ct_r) + \frac{1}{2c} \int_0^{x_j} g(s)ds - \frac{1}{2c} \int_{x_j - ct_r}^{x_j} g(s)ds.$$

Análogamente, utilizando el método del trapecio quedará,

$$w_{j,r}^+ = \frac{1}{2}f(x_j + ct_r) + \frac{h}{4c} \left(g(x_0) + 2 \sum_{i=1}^{j-1} g(x_i) + g(x_j) \right) \\ - \frac{ck}{4c} \left(g(x_j - ct_r) + 2 \sum_{i=r-1}^1 g(x_j - ct_i) + g(x_j) \right).$$

Con esto ya podríamos implementar las soluciones en caso de que g no sea integrable. Más adelante, en la sección 5.2, ilustraremos la relación de estas soluciones, con la aproximación numérica.

4.7 Ley de energía discreta

En este punto nos preguntamos como se traduciría la ley de energía a nuestro esquema. Al ser el esquema discreto, tendremos que usar la definición de producto interior discreto 2.2.3, en lugar de la del producto interior anteriormente utilizada en la sección 3.4.

Como queremos comprobar que en nuestra aproximación se sigue dando la conservación de la energía, vamos a comenzar tomando como dominio el mallado 2.2.2. Sabiendo que vamos a utilizar el concepto de suma por partes, parece conveniente definir D como la partición del interior del eje espacial, ya que, en el interior es donde tenemos definido nuestro esquema (4.6).

Sin embargo, en nuestro caso nos beneficiaremos enormemente de utilizar $D = \{x_0, x_1, \dots, x_{m-1}\}$. Por tanto, vamos a analizar de que forma podría definirse nuestro esquema para los puntos de la forma (x_0, t_r) , con $r = 1, \dots, n-1$.

Evaluemos la ecuación 3.4, en el punto (x_0, t_r)

$$u_{tt}(x_0, t_r) = c^2 u_{xx}(x_0, t_r).$$

Utilizando la condición (3.7), obtenemos,

$$0 = c^2 u_{xx}(x_0, t_r),$$

y por lo tanto para que se verifique nuestro esquema, la aproximación en dicho punto, tendrá que ser,

$$0 = \frac{u_{1,r} - 2u_{0,r} + u_{-1,r}}{h^2}.$$

Vemos que al igual que (2.7), obtenemos un nodo virtual, el cual, si definimos de la siguiente forma

$$u_{-1,r} \triangleq -u_{1,r},$$

y volviendo a utilizar (4.9), hará que se verifique nuestro esquema en el punto (x_0, t_r) .

Retomando el tema que nos atañe, la energía en nuestro esquema, ya con $D = \{x_0, x_1, \dots, x_{m-1}\}$, y utilizando en este caso la función δ definida en 2.2.5, para hacer el producto interior discreto con $\delta_t u$, tendremos.

$$(\delta_{tt} u, \delta_t u)_D = c^2 (\delta_{xx} u, \delta_t u)_D.$$

Siguiendo la notación que tomábamos en 2.2.1, para cada $r = 1, \dots, n-1$ tendremos:

$$(\delta_{tt} u_{\cdot,r}, \delta_t u_{\cdot,r})_D - c^2 (\delta_{xx} u_{\cdot,r}, \delta_t u_{\cdot,r})_D = 0. \quad (4.14)$$

Ahora para desarrollar el segundo termino de la igualdad usaremos la observación 2.2.6 y posteriormente el lema 2.2.1.

4. RESOLUCIÓN NUMÉRICA DE LA ECUACIÓN DE ONDA

$$\begin{aligned}
 (\delta_{xx}u_{\cdot,r}, \delta_t u_{\cdot,r})_D &\stackrel{2,2,6}{=} (\delta_{x^+x^-}u_{\cdot,r}, \delta_t u_{\cdot,r})_D, \\
 &\stackrel{(2.6)}{=} -(\delta_{x^+}u_{\cdot,r}, \delta_{x^+}\delta_t u_{\cdot,r})_D - (\delta_t u_{0,r})(\delta_{x^-}u_{-1,r}) + (\delta_t u_{m,r})(\delta_{x^-}u_{m-1,r}).
 \end{aligned} \tag{4.15}$$

(REVISAR) Insertando esto en la ecuación (4.14) nos quedaría:

$$(\delta_{tt}u_{\cdot,r}, \delta_t u_{\cdot,r})_D + c^2 (\delta_{x^+}u_{\cdot,r}, \delta_{x^+}\delta_t u_{\cdot,r})_D = c^2 (\delta_t u_{0,r})(\delta_{x^-}u_{-1,r}) - c^2 (\delta_t u_{m,r})(\delta_{x^-}u_{m-1,r}).$$

Denotando al termino de la derecha de la igualdad como \mathfrak{b} tendremos que

$$\delta_t E(u) = \mathfrak{b}.$$

Por lo tanto, para que se conserve la energía tendrá que ser $\mathfrak{b} = 0$. Veamos que es así.

Utilizando la condición de contorno para (4.9), para $r = 1, \dots, n-1$,

$$\delta_t u_{0,r} = \frac{1}{k} (u_{0,r+1} - u_{0,r-1}) = 0,$$

$$\delta_t u_{m,r} = \frac{1}{k} (u_{m,r+1} - u_{m,r-1}) = 0.$$

Luego nuestro esquema conserva la energía.

Modelado en Python y comparativa

En este capítulo nos centraremos en ver el modelado en Python de nuestro esquema resuelto por el método de diferencias finitas, (4.6). Veremos como varía la solución en varios instantes de tiempo, la forma que presenta la onda cambiando las condiciones iniciales (4.7) y (4.8), y haremos una comparativa de como afectan las condiciones de estabilidad (4.12).

A lo largo del capítulo se irán discutiendo varios archivos de audio realizados con el código de python del Anexo A, estos se pueden encontrar en el repositorio de Github del trabajo, **Simulación numérica de ecuaciones de onda**.

5.1 Soluciones

En esta sección comenzaremos señalando varias soluciones dependiendo de las condiciones iniciales. En toda la sección establecemos $c = 440$, y $\mu = 1$. Por lo tanto los demás parámetros serán

5.1.1 $f(x) = \sin(\pi x), g(x) = 0$

Cuando la posición inicial depende del seno y la velocidad inicial es 0, se puede observar como nuestra solución oscila sin llegar a parar, al no tener en cuenta nuestro sistema (4.6), fuerzas externas como el rozamiento. Utilizaremos como número de puntos en el espacio

5. MODELADO EN PYTHON Y COMPARATIVA

$m = 100$, por lo tanto los demás parámetros serán $h = \frac{L-0}{m} = \frac{1}{100}$ y como $\mu = \frac{ck}{h} = 1$, $k = \frac{1}{44000}$.

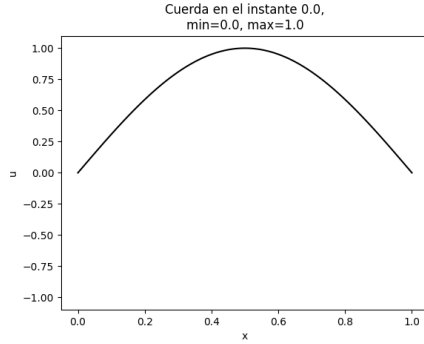


Figura 5.1: Posición inicial de la cuerda.

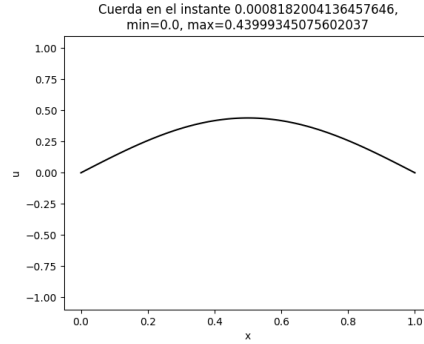


Figura 5.2: La cuerda comienza a bajar.

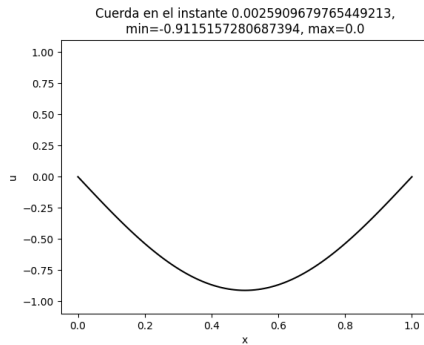


Figura 5.3: Sigue bajando hasta llegar a la posición de máxima amplitud.

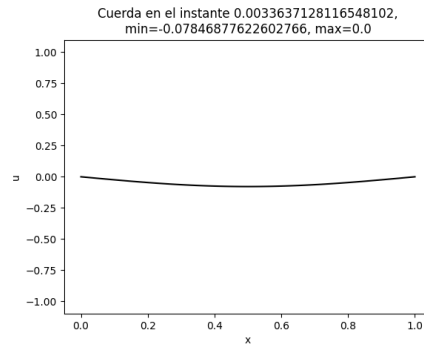


Figura 5.4: La cuerda comienza a subir

Este tipo de movimiento es muy regular, podemos observarlo, por ejemplo, en el movimiento del punto el punto $x = 0,5$ respecto al tiempo. Esta cadena de datos será la que guardemos en el archivo de audio. En este caso por la elección de $c = 440$, se escuchará un La central, y por la elección del punto, estará a máxima amplitud y por tanto volumen.

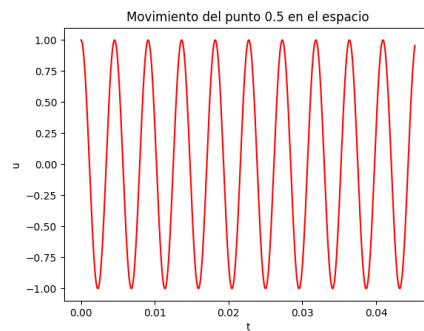


Figura 5.5: Se puede escuchar este sonido en el archivo figura5-5.wav adjunto.

5.1.2 $f(x)$ tipo Struck, $g(x) = 0$

Cuando la posición inicial es de tipo Struck, es decir, un "pico" en la zona central de la cuerda, y la velocidad inicial es 0, la solución no es tan suave, al ser f no derivable en un punto.

La expresión de f es la siguiente,

$$f(x) = \begin{cases} 0 & \text{si } 0 \leq x \leq \frac{1}{4} \\ 4x - 1 & \text{si } \frac{1}{4} < x \leq \frac{1}{2} \\ 3 - 4x & \text{si } \frac{1}{2} < x \leq \frac{3}{4} \\ 0 & \text{si } \frac{3}{4} < x \leq 1 \end{cases}$$

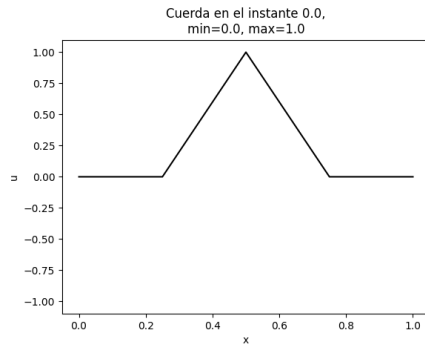


Figura 5.6: Posición inicial de la cuerda.

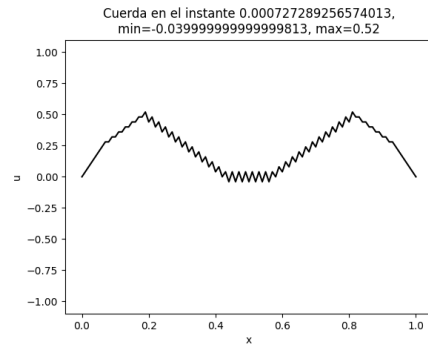


Figura 5.7: Se crean picos en la cuerda.

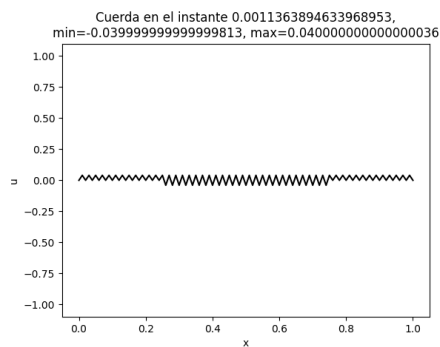


Figura 5.8

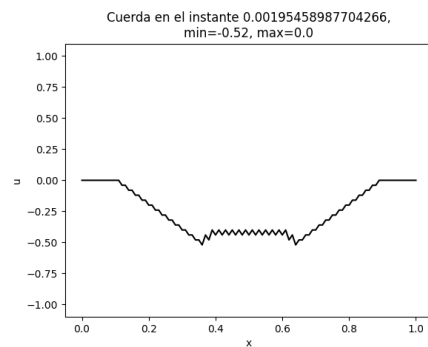


Figura 5.9

En este caso, si aumentamos el número de puntos del dominio espacial, los picos propios del punto no derivable irán desapareciendo. Lo ilustramos con tres imágenes en instantes similares, para $m = 100$, $m = 300$ y $m = 400$, siempre con $\mu = 1$ y $c = 440$.

5. MODELADO EN PYTHON Y COMPARATIVA

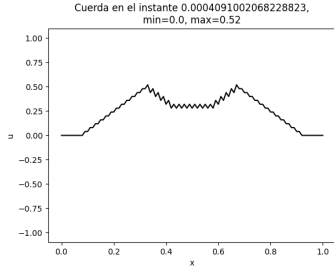


Figura 5.10: $m = 100$,
 $h = \frac{1}{100}$ y $k = \frac{1}{44000}$.

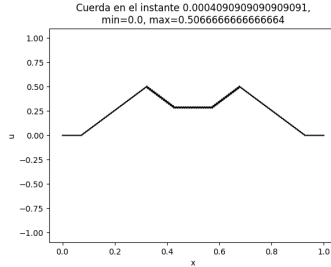


Figura 5.11: $m = 300$,
 $h = \frac{1}{300}$ y $k = \frac{1}{132000}$.

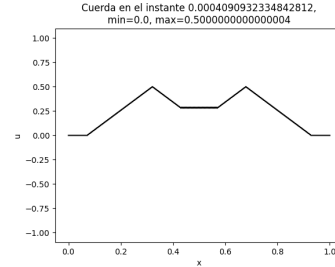


Figura 5.12: $m = 400$,
 $h = \frac{1}{400}$ y $k = \frac{1}{176000}$.

La forma de la onda en cada punto no es tan regular como en la sección 5.1.1, y cambia mucho dependiendo del punto que observemos, como ejemplo visualizaremos el movimiento de los puntos $x = 0,07$ y $x = 0,5$.

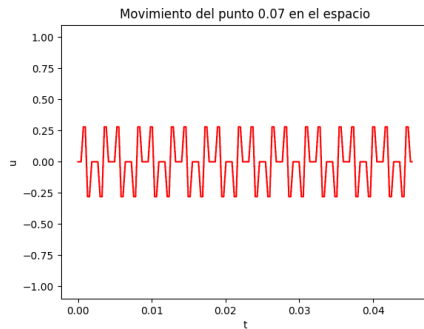


Figura 5.13: Archivo figura5-10.wav

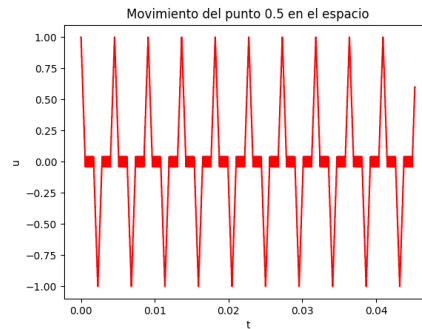


Figura 5.14: Archivo figura5-11.wav

Se observa que la onda varía dependiendo de la posición de la cuerda en las que se haga la observación, por ende, también el sonido producido, y el volumen del mismo.

5.1.3 $f(x) = 0$, $g(x)$ tipo Struck

En este caso partimos de una onda en reposo, con una velocidad inicial tipo Struck. A la derecha vemos una gráfica de la forma que tiene esta velocidad inicial.

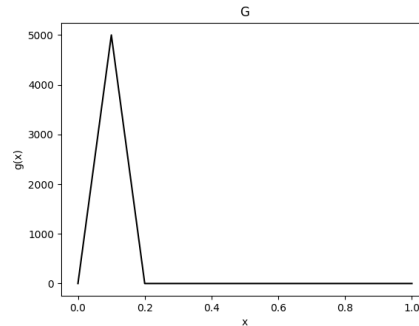


Figura 5.15: Velocidad inicial.

La definición de la función g es la siguiente:

$$g(x) = \begin{cases} 50000x & \text{si } 0 \leq x \leq \frac{1}{10} \\ 50000(0,2 - x) & \text{si } \frac{1}{10} < x \leq \frac{1}{5} \\ 0 & \text{si } \frac{1}{5} < x \leq 1 \end{cases}$$

En esta imagen vemos la posición inicial. Dependiendo del número de puntos que elijamos en nuestro mallado, podremos ver que la solución se hace más suave, lo ilustraremos eligiendo 50 y 200 puntos en el eje espacial, respetando $\mu = 1$ y $c = 440$.

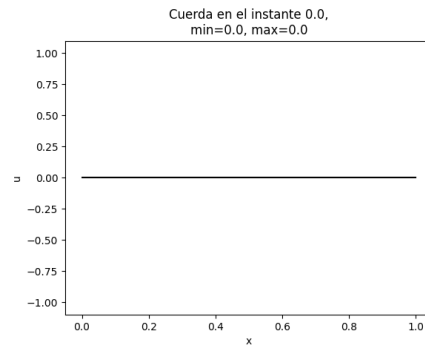


Figura 5.16: Posición inicial de la cuerda.

Muestra de la solución con $m = 50$, $h = \frac{1}{50}$ y $k = \frac{1}{22000}$.

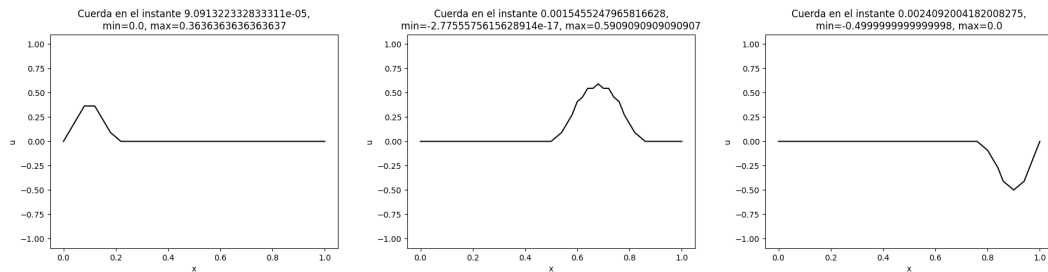


Figura 5.17: Observamos como la onda se desplaza hasta el lado derecho de la cuerda hasta desaparecer para luego aparecer en la parte inferior.

5. MODELADO EN PYTHON Y COMPARATIVA

Muestra de la solución con $m = 200$, $h = \frac{1}{200}$ y $k = \frac{1}{88000}$:

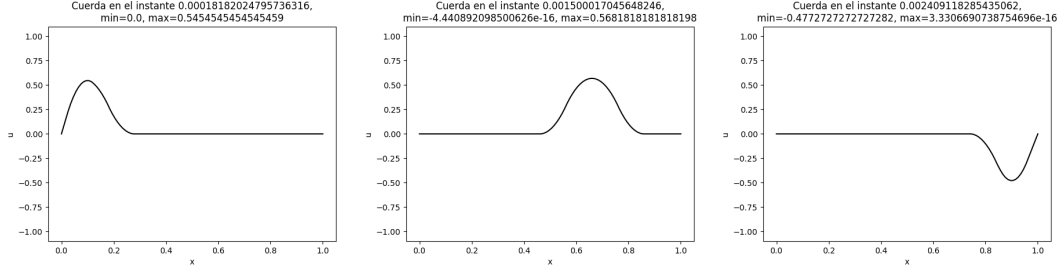


Figura 5.18: En este caso, el movimiento se mantiene pero la solución es mucho más suave, sin ondulaciones en la misma.

Aún así, para los puntos del mallado, en los que se aproxima la solución, el movimiento del mismo respecto al tiempo será prácticamente idéntico independientemente de si $m = 50$ o $m = 200$ y la diferencia en el audio indistinguible. Un ejemplo es el esquema de la derecha, hecho para $x = 0,48$.

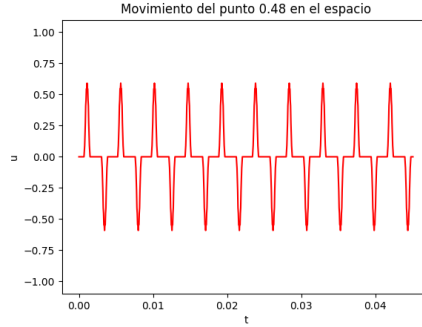


Figura 5.19: Archivo figura5-20.wav

5.2 Soluciones tipo onda viajera

En esta sección nuestro objetivo será ejemplificar lo explicado en la sección 3.3, añadiendo además la solución exacta de la ecuación para comprobar cuanto variará nuestra aproximación respecto a la misma. Utilizando los parámetros $c = 400$, $\mu = 1$, $m = 100$ y por lo tanto, $h = \frac{1}{100}$ y $k = \frac{1}{44000}$ de nuevo y la función seno, las soluciones de tipo onda viajera en los siguientes casos serían:

5.2 Soluciones tipo onda viajera

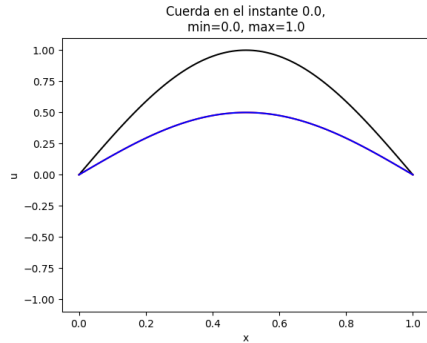


Figura 5.20: Ambas soluciones se encuentran superpuestas.

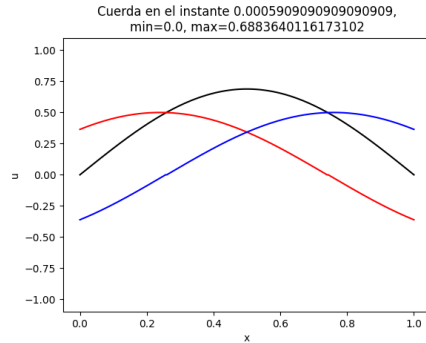


Figura 5.21: Coloreada de azul la solución que se desplaza a la derecha y de rojo la que se desplaza hacia la izquierda.

Agrandes rasgos parece que la suma de ambas soluciones, y por tanto la solución exacta (3.17), se acercará mucho a la aproximación que hemos conseguido numéricamente, coloreada de negro. Para ver que es así ilustremoslo dibujando la solución exacta.

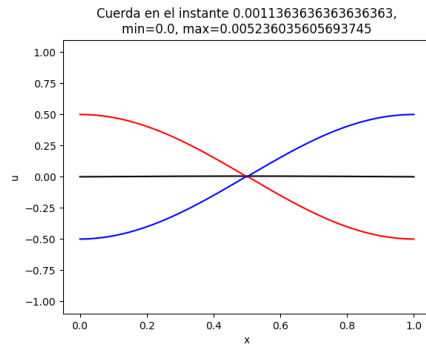


Figura 5.22: Movimiento de las soluciones de tipo onda viajera.

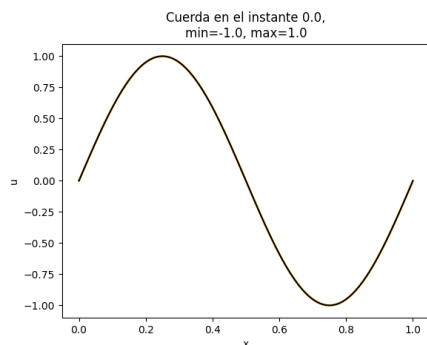


Figura 5.23: Al principio la solución exacta, coloreada de naranja, y la aproximación se superponen.

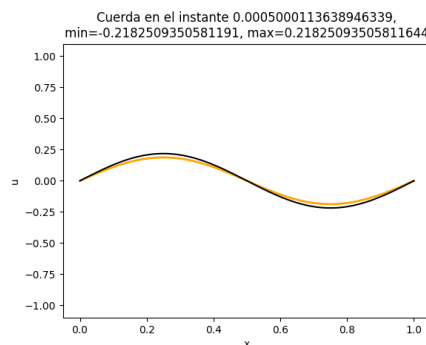


Figura 5.24: Conforme se acercan a la posición $u \equiv 0$, las diferencias son cada vez más notables.

5. MODELADO EN PYTHON Y COMPARATIVA

Podemos observar de manera incluso más clara el movimiento de las soluciones de tipo onda viajera utilizando la función $f(x) = \text{sen}(2\pi x)$.

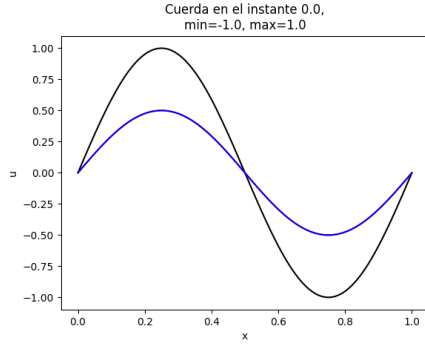


Figura 5.25

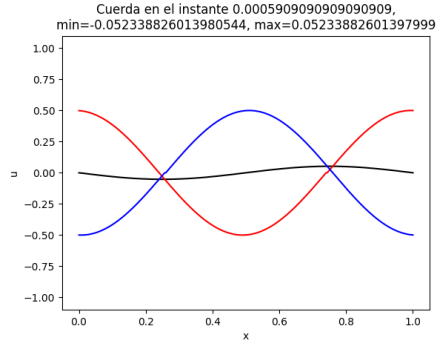


Figura 5.26

5.3 Comparación variando el valor de μ

Ahora veremos como cambiando el parámetro μ , las soluciones obtenidas propagarán inestabilidades.

5.3.1 $\mu \ll 1$

Cuando elegimos μ mucho menor que 1, es decir, alejándonos de la solución óptima, poco a poco comienzan a generarse inestabilidades. Comparemos la solución con $\mu = 1$, para $f(x)$ de tipo Struck definida como en 5.1.2, es decir,

$$f(x) = \begin{cases} 0 & \text{si } 0 \leq x \leq \frac{1}{4} \\ 4x - 1 & \text{si } \frac{1}{4} < x \leq \frac{1}{2} \\ 3 - 4x & \text{si } \frac{1}{2} < x \leq \frac{3}{4} \\ 0 & \text{si } \frac{3}{4} < x \leq 1 \end{cases}$$

y tomando como parámetros $c = 440$, $m = 100$ y $h = \frac{1}{100}$, compararemos estas soluciones.

5.3 Comparación variando el valor de μ

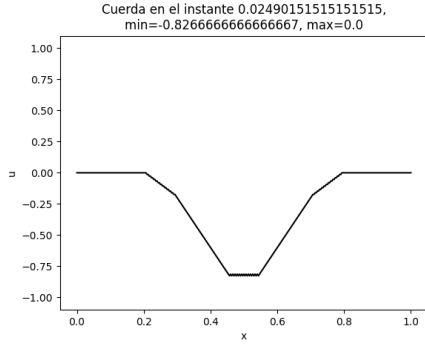


Figura 5.27: Solución para $\mu = 1$ y $k = \frac{1}{44000}$

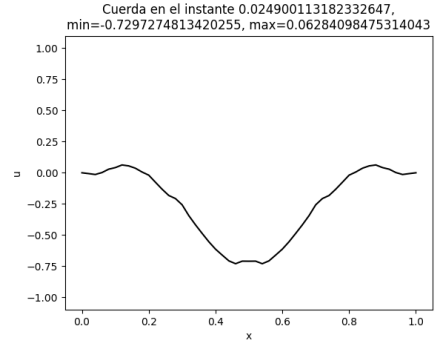


Figura 5.28: Solución para $\mu = 0,1$ y $k = \frac{1}{440000}$

En la imagen de la derecha podemos observar las inestabilidades en forma de ondas por toda la cuerda. Por otro lado, vemos claramente las diferencias en las imágenes del movimiento de la onda dependiendo del tiempo, con el punto $x = 0,5$ fijado.

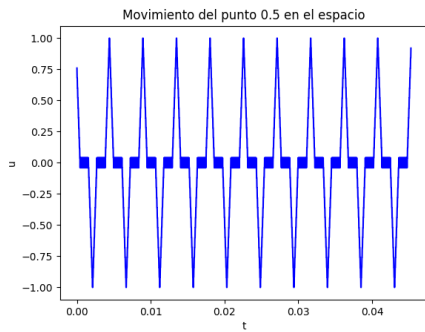


Figura 5.29: Movimiento de la onda para $\mu = 1$. Archivo figura5-32.wav

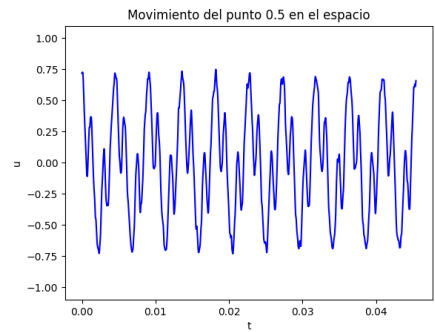


Figura 5.30: Movimiento de la onda para $\mu = 0,1$. Archivo figura5-33.wav

Aún viendo claramente las inestabilidades de forma gráfica, no es tan fácil escucharlas en los archivos de audio adjuntos.

5.3.2 $\mu > 1$

Cuando violamos la condición CFL, las inestabilidades se propagarán aún más rápido. Utilizando el mismo ejemplo que en el apartado anterior, podremos comprobar como se propagan a todos los puntos de la cuerda antes de llegar a los 0,002 segundos. Para este caso utilizaremos $c = 440$, $m = 40$, $h = \frac{1}{40}$, tomamos $\mu = 1,001$, y por tanto $k = \frac{1}{16000}$

5. MODELADO EN PYTHON Y COMPARATIVA

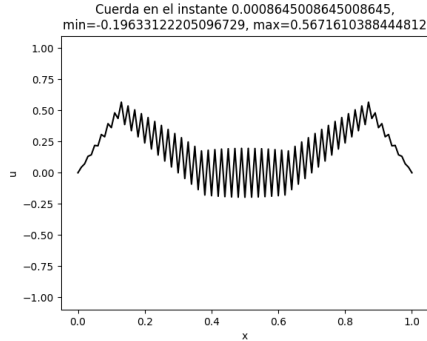


Figura 5.31: Las inestabilidades comienzan rápidamente.

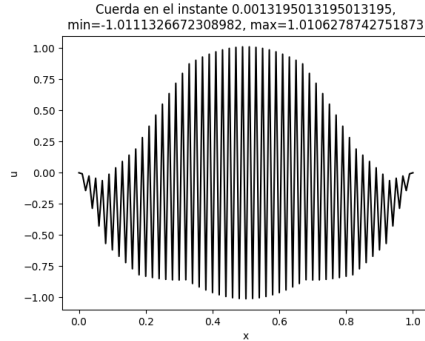


Figura 5.32: Las inestabilidades cubren la cuerda al completo.

En este caso, a mayor sea m antes se propagarán las inestabilidades, por ello hemos elegido $m = 40$, para observar como cambia la onda dependiendo del tiempo, en la posición $x = 0,5$.

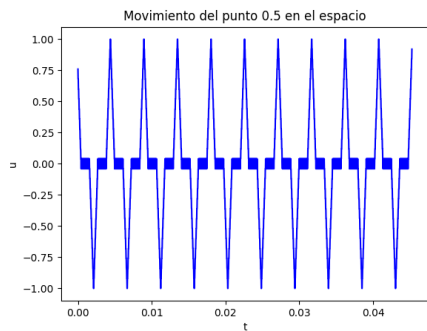


Figura 5.33: Movimiento de la onda para $\mu = 1.5.27$

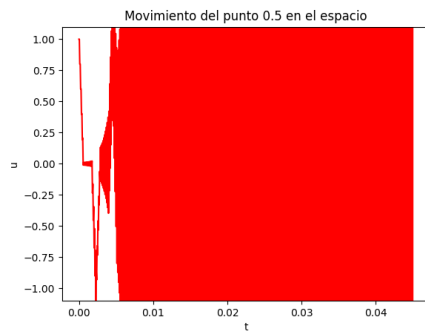


Figura 5.34: Movimiento de la onda para $\mu = 1,1$. Archivo fig5-38.wav

Si intentamos reproducir el sonido de esta última situación no se escuchará nada, puesto que las inestabilidades son tan grandes que hacen imposible su escucha, dos causas posibles son:

- El guardado en el formato .wav, puesto que cuando la amplitud es mayor que 1, no lo capta de manera adecuada.
- La frecuencia, al ser tan elevada, es imposible de escuchar para el oído humano.

5.4 Archivos .wav y síntesis de sonido

Continuando con la reflexión anterior, es importante mencionar como se ha conseguido la síntesis del sonido, y el guardado del mismo.

5.4.1 $f(x) = \sin(2\pi x)$, $g(x) = 0$

(CAMBIAR PARA QUE SEA ACORDE AL SONIDO) En cada uno de los casos anteriores, tal y como hemos dicho al principio de la sección, $c = 440$ y como solo hay dos nodos, puntos en los que la amplitud es siempre 0, en todos los casos la frecuencia será $440Hz$. Esta frecuencia corresponde a un La central (4ª octava), la hemos elegido al ser la más usual para ejemplificar este tipo de modelado.

Ahora, elijamos un caso en el que tengamos 3 nodos, por ejemplo $f(x) = \sin(2\pi x)$, La frecuencia se corresponderá a un La de 5ª octava.

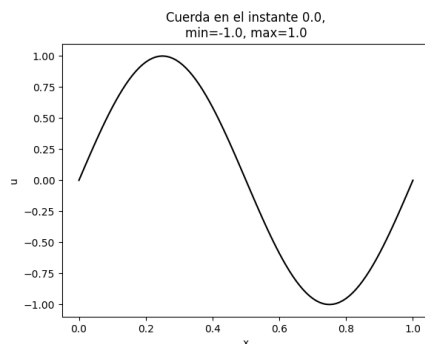


Figura 5.35: Posición inicial.

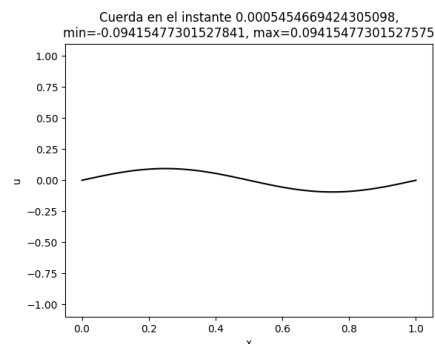


Figura 5.36

Podemos observar esto eligiendo los puntos adecuados, $x = 0,5$ y $x = 0,25$, para seno de πx , 5.1.1, y seno de $2\pi x$. Vemos claramente que llega a su máxima amplitud el doble de veces.

5. MODELADO EN PYTHON Y COMPARATIVA

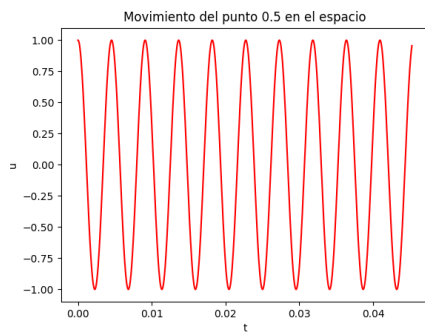


Figura 5.37: $\sin(x)$ en $x = 0,5$ Archivo figura5-5.wav

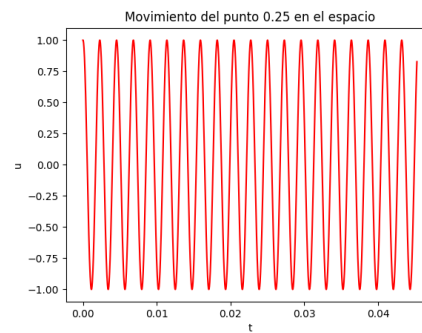


Figura 5.38: $\sin(2x)$ en $x = 0,25$ Archivo figura5-24.wav

Conclusiones y proyectos futuros

A lo largo de este trabajo, hemos introducido conceptos importantes, tales como los espacios de Lebesgue y el producto interior, los hemos generalizado para el caso discreto, en un mallado definido para dicho ambiente.

Se ha contextualizado la ecuación en derivadas parciales de onda, utilizando una perspectiva física. También se han analizado ciertas soluciones de la ecuación, y la Ley de la conservación de energía en la misma.

Una vez hecho esto, hemos examinado la solución numérica de la ecuación de onda mediante el método de las diferencias finitas, observando en 5.1, como se obtienen buenos resultados para la misma. Hemos comparado distintas condiciones iniciales que cumplen con los parámetros obtenidos mediante el análisis de Von Neumann 4.5 y hemos conseguido archivos de audio que corroboran la estabilidad dentro del marco obtenido.

Por otro lado, hemos logrado la implementación de las soluciones de tipo onda viajera 5.2, de las que derivamos que la suma de las mismas describe inequívocamente el movimiento de la cuerda de nuestro esquema.

A continuación hemos representado la relación entre la condición CFL y la inestabilidad de las soluciones, observando como su incumplimiento lleva a una aproximación muy

6. CONCLUSIONES Y PROYECTOS FUTUROS

mala que presenta multitud de oscilaciones espúreas.

En definitiva, hemos podido realizar un estudio general de la ecuación de onda en derivadas parciales, abordando desde los conceptos fundamentales hasta la comprobación de como las condiciones teóricas encontradas afectan la solución numérica de la misma.

Este estudio sería fácilmente ampliable tomando varios enfoques,

- Utilizando argumentos similares a los encontrados en [20], podríamos hablar de la controlabilidad y observabilidad de nuestro esquema, en lugar de en un ambiente semi-discreto como el que propone Enrique Zuazua.
- Se podría enriquecer el estudio comparando diferentes esquemas usados asiduamente en el método de las diferencias finitas, o revisando diversos conceptos para nuestro esquema, como la controlabilidad y la observabilidad [20, 21].
- También sería interesante el uso de métodos de resolución diferentes, como por ejemplo los métodos espectrales, o elementos finitos [22].
- Un enfoque más aplicado que se podría dar es, la utilización de los archivos .wav sintetizados para la creación de música, ya que hemos visto, que dependiendo de las condiciones iniciales, la forma de la onda obtenida cambia, de manera similar a como lo hace al tocar la misma nota en dos instrumentos diferentes.



Anexo

En este capítulo anexaremos el código realizado en Python. También se puede encontrar en el repositorio de Github del trabajo: **Simulación numérica de ecuaciones de onda**.

A.1 Resolución numérica

Comenzamos implementando el esquema (4.6) y así resolviendo la ecuación de onda en ese contexto.

```
import numpy as np
import time
import os as os
from shutil import rmtree
import matplotlib.pyplot as plt
from scipy.integrate import quad

#Funciones necesarias

# Para inicializar los valores iniciales que dependen de la derivada lo haremos con
una aproximación progresiva del tiempo

def g(x):
    return 0

'''
# En este caso tomamos una condición tipo struck en 0.4 , 0.5 max
def g(x):
    if ( x<=0.4):
        return 2.5*x
    else:
```

A. ANEXO

```
        return -(10/6)*x
    ...

    ...
# En este caso tomamos una condición tipo struck en 0.1 , 1 max
def g(x):
    if ( x<=0.1):
        return 50000*x
    elif (x <= 0.2):
        return 50000*(0.2-x)
else:
    return 0
...
...
# En este caso tomamos una condición tipo crc c0=1000 x0 = 0.5
def g(x):
    if (0.4<=x <= 0.6):
        return 1000/2 * (1+np.cos(np.pi*(x-0.4)/0.1))
    else:
        return 0
...

# Condición de contorno en el eje de espacial tipo seno
def f(x):
    aux = []
    for i in x:
        aux.append(np.sin(2*np.pi*i))
    return aux # Tener cuidado con la función por si se sale del rango
...

#Condición de contorno en el eje de posición tipo pico
def f(x):
    aux = []
    for i in x:
        if ( i<=0.25):
            aux.append(0)
        elif ( i<=0.5):
            aux.append(4*(i-0.25))
        elif ( i<=0.75):
            aux.append(4*(0.75-i))
        else:
            aux.append(0)
    return aux
...

...
#Condición de contorno en el eje de posición tipo trapecio
def f(x):
    aux = []
    for i in x:
        if ( i<=0.1):
            aux.append(0)
        elif ( i<=0.3):
            aux.append(5*i-0.5)
```

```

        elif ( i <=0.7):
            aux.append(1)
        elif ( i <=0.9):
            aux.append(-5*i+4.5)
        else:
            aux.append(0)
    return aux
'''

'''
#condición de tipo c_rc con x_hw= 0.2 x_0 = 0.5 c_0= 1
def f(x):
    aux = []
    for i in x:
        if( i <=0.3):
            aux.append(0)
        elif ( i <=0.7):
            aux.append((1+np.cos(np.pi * (i-0.5)/0.2))/2)
        else:
            aux.append(0)
    return aux
'''

'''
# Condición de contorno en el eje de posición tipo plano
def f(x):
    lon = len(x)
    return np.zeros(lon)
'''

# Definimos una función para usar en el método explícito y poder cambiarlo de manera
rápida si hace falta. Si queremos hacerlo en un file independiente tendremos
que añadir el mu
def Explicito(x1, x2, x3, x4,mu):
    return mu ** 2 * (x1 + x2) + 2 * (1- mu ** 2) * x3 - x4

start = time.time()
print("Vamos a hacer una resolución numérica de la ecuación de onda u_tt-c^2u_xx=0")
print("Introduce c:")
c= float(input())
print("Introduce el nombre del directorio donde quieres guardar los archivos:")
directorio=input()

#Creamos una carpeta en la que se guardarán los datos
home = 'C:\\Users\\Ana Cuevas de Cózar\\PycharmProjects\\pythonProject2 '
directoriofinal=home+'\\'+directorio+'\\'
#Si existe el directorio lo borra
try:
    rmtree(directoriofinal)
except:
    print('No existía el directorio , lo creamos')
os.mkdir(directoriofinal)

```

A. ANEXO

```
#Constantes necesarias
l=1
tmax= 1
# Constante representativa del número de trozos en los que separamos el espacio
m = int(1*100)
h=1/m #Cambiamos la h para que sea entera

#Lo tomamos de esta forma para ganar en exactitud pagina 133 numerical sound synthesis
k =h/c

# Cambiamos k para hacer las comparaciones
',,
k =h/c * 0.1
k =1.001*h/c
',,

n= int(tmax/k)
mu = c * k / h

if mu>1 :
    raise Exception('Mu es mayor que 1')
print(f'n{n}, c{c}, k{k}, h{h}, m{m}, mu{mu}')
```

```
# Creamos un mallado de puntos en los que aproximaremos la solución
x= np.linspace(0,l,num=m +1)
t= np.linspace(0,tmax,num=n +1)

# Creamos la matriz u de soluciones
# u(x,t)
u = np.empty((m + 1, n + 1), float)

# Añadimos antes que nada los nodos conocidos por las condiciones de contorno
u[:, 0] = f(x)

u[0, :] = np.zeros(n + 1)

u[m, :] = np.zeros(n + 1)
aux=[0]

# Para inicializar los valores iniciales que dependen de la derivada lo haremos con una
aproximación progresiva del tiempo
for i in range(1, m):
    u[i, 1] = u[i, 0] + k * g(x[i])
    aux.append(g(x[i]))

    aux.append(g(x[m]))

for j in range(2, n + 1):
    u[1:m, j] = Explicito(u[2:m+1,j-1], u[0:m-1,j-1], u[1:m,j-1], u[1:m,j-2],mu)
```



```
#Implementamos la solución de tipo onda viajera.

# Creamos la matriz wl la solución viajera hacia la izquierda y wr hacia la derecha

wl = np.empty((m + 1, n + 1), float) #Solo para mu=1
wr = np.empty((m + 1, n + 1), float) #Solo para mu=1

#Inicializamos la solución
for i in range(0, m+1):
    for j in range(0, n+1):
        wl[i, j] = (1/2)*f([x[i]+c*t[j]])[0] + (1/(2*c))*quad(g, 0, x[i]+c*t[j])[0]
        wr[i, j] = (1/2)*f([x[i]-c*t[j]])[0] + (1/(2*c))*quad(g, 0, x[i]-c*t[j])[0]

end = time.time()

plt.plot(x, aux, 'k')
plt.xlabel('x')
plt.ylabel('g(x)')
plt.title(f'G')
plt.savefig(directoriofinal+'prueba g.png')
plt.close()

print(f'Ha tardado {np.floor(end-start)} segundos.')
np.save(directoriofinal+'u', u)
np.save(directoriofinal+'x', x)
np.save(directoriofinal+'t', t)

np.save(directoriofinal+'wl', wl)
np.save(directoriofinal+'wr', wr)

np.save(directoriofinal+'constantes', [c, m, h, n, k, mu, tmax, l])
#Guardar en cons las constantes que necesitamos para usarlas luego en el guardado
del sonido
```

Al final guardamos los datos obtenidos para su posterior procesado.

A.2 Wave

Creo ahora un archivo de audio a partir de los datos anteriores y lo guardo en un archivo de sonido .wav

```
import numpy as np
from scipy.io.wavfile import write

print("Vamos a guardar nuestros datos en formato wav")
print("Introduce el nombre de la carpeta:")
dir = str(input())

#Creamos el directorio
directorio = 'C:\\Users\\Ana Cuevas de Cózar\\PycharmProjects\\pythonProject2\\'+dir+'\\'
```

A. ANEXO

```
#Añadimos los datos
datos = np.load(directorio+'u.npy', mmap_mode='r')
t = np.load(directorio+'t.npy', mmap_mode='r')
x = np.load(directorio+'x.npy', mmap_mode='r')
[f, ml, hl, nl, kl, mu, tmax, l]= np.load(directorio+'constantes.npy', mmap_mode='r')

ml=int(ml)
nl=int(nl)

#Constantes necesarias
sample_rate = 44100
n2 = int(tmax*sample_rate)
t_necesario= np.linspace(0,tmax,n2 +1)

# Escogiendo un punto en particular
# Comprobamos donde está el punto de mayor amplitud en el instante inicial y en ese
punto es donde veremos como se mueve la onda.
pos_readout= np.where(datos[:,0]==max(datos[:,0]))[0][0]

if pos_readout==0:
    pos_readout= int(ml/(2*1))
    print('Cambio la posicion a 0.5')

# Elijo un punto en el espacio en el que veremos el movimiento de la cuerda
wave_table = datos[pos_readout]

output = np.interp(t_necesario, t, wave_table)

#Guardamos
write(directorio+f'Sonido.wav', sample_rate, output)
```

A.3 Imágenes

Creamos y guardamos las imágenes de la posición de la cuerda en varios instantes de tiempo y de los cambios de posición de la onda en un punto del eje X durante un periodo de tiempo.

```
import numpy as np
import matplotlib.pyplot as plt
from os import mkdir
from shutil import rmtree

print("Vamos a guardar nuestras imagenes")
print("Introduce el nombre de la carpeta:")
dir= str(input())

#Creamos los directorios
directorio='C:\\Users\\Ana Cuevas de Cózar\\PycharmProjects\\pythonProject2\\'+dir+'\\'
```

```

cuerda=directorio + 'Imágenes Cuerda\\'
onda= directorio+ 'Imágenes Onda\\'
try:
    rmtree(cuerda)
    rmtree(onda)
except:
    print('No existían las imágenes, las creamos')
mkdir(onda)
mkdir(cuerda)

#Añadimos los datos
datos = np.load(directorio+'u.npy', mmap_mode='r')

#Solo para ondas viajeras.
datoswl = np.load(directorio+'wl.npy', mmap_mode='r') #Solo para mu=1
datoswr = np.load(directorio+'wr.npy', mmap_mode='r') #Solo para mu=1

t = np.load(directorio+'t.npy', mmap_mode='r')
x = np.load(directorio+'x.npy', mmap_mode='r')
[f, m,h,n,k,mu,tmax,l]= np.load(directorio+'constantes.npy', mmap_mode='r')

n=int(n)
m=int(m)

#Vamos a pintar las imágenes de a posición de la cuerda en instantes seleccionados

#Solución exacta
plt.plot(x, datoswr[:, 0]+datoswl[:, 0], color='orange', linewidth=2)

plt.plot(x, datos[:, 0], 'k')

#Ondas viajeras
plt.plot(x, datoswl[:, 0], 'r')
plt.plot(x, datoswr[:, 0], 'b')

plt.ylim(-1.1,1.1)
plt.xlabel('x')
plt.ylabel('u')
plt.title(f'Cuerda en el instante {t[0]}, \nmin={min(datos[:,0])}, max={max(datos[:,0])}')
plt.savefig(cuerda+f'Cuerda_instante_{0}.png')
plt.close()

if (int(2*n/f)+2)/100 > 1:
    aux=int((int(2*n/f)+2)/100)
else:
    aux=1

for j in range(1, 2*int(2*n/f)+2):
    if j \%aux ==0 :
        #Solución exacta
        plt.plot(x, datoswr[:, j] + datoswl[:, j], color='orange', linewidth=2)

```

A. ANEXO

```
plt.plot(x, datos[:, j], 'k')

'''
#Ondas viajeras
plt.plot(x, datoswl[:, j], 'r')
plt.plot(x, datoswr[:, j], 'b')
'''

plt.ylim(-1.1, 1.1)
plt.xlabel('x')
plt.ylabel('u')
plt.title(f'Cuerda en el instante {t[j]}, \nmin={min(datos[:, j])},
max={max(datos[:, j])}')
plt.savefig(cuerda+f'Cuerda_instante_{j}.png')
plt.close()

#Pintamos la imagen de la onda en los 1000 primeros instantes en cada punto
for j in range(0, m + 1):
    plt.plot(t[0:10*int(2*n/f)+2], datos[j, 0:10*int(2*n/f)+2], 'r')
    plt.ylim(-1.1, 1.1)
    plt.xlabel('t')
    plt.ylabel('u')
    plt.title(f'Movimiento del punto {x[j]} en el espacio') #axis tight
    plt.savefig(onda+f'PrimerosmilPunto_{j}.png')
    plt.close()

    plt.plot(t[0:10*int(2*n/f)+2], datos[j, n - (10*int(2*n/f)+2):n], 'b')
    plt.ylim(-1.1, 1.1)
    plt.xlabel('t')
    plt.ylabel('u')
    plt.title(f'Movimiento del punto {x[j]} en el espacio') # axis tight
    plt.savefig(onda + f'UltimosmilPunto_{j}.png')
    plt.close()
```

Bibliografía

- [1] Agustín Martín Domingo. *Apuntes de vibraciones y ondas*. El autor, Madrid, Septiembre 2020. 1, 14
- [2] Xiaobo Liu, Jingyi Chen, Zhencong Zhao, Haiqiang Lan, and Fuping Liu. Simulating seismic wave propagation in viscoelastic media with an irregular free surface. *Pure and applied geophysics*, 175(10):3419–3439, 2018. 2
- [3] L.E. Kinsler, A.R. Frey, A.B. Coppens, and J.V. Sanders. *Fundamentals of Acoustics*. Wiley, 2000. 2
- [4] E. Hecht. *Optics*. Pearson Education, 2016. 2
- [5] J. W. Thomas. *Numerical partial differential equations: Finite difference methods*, volume 22. Springer, 1995. 2
- [6] Alexandre Ern and Jean-Luc Guermond. *Theory and practice of finite elements*, volume 159. Springer, New York, 2004. 2
- [7] Alfio Quarteroni, Tom Hou, and Claude Le Bris. *Numerical Models Differential Problems*, volume 2. Springer Milan, 2009. 2
- [8] H.L. Royden and P. Fitzpatrick. *Real Analysis*. Prentice Hall, 2010. 5
- [9] Bo Strand. Summation by parts for finite difference approximations for d/dx . *Journal of Computational Physics*, 110(1):47–67, 1994. 7
- [10] Abdul-Majid Wazwaz. *Partial Differential Equations and Solitary Waves Theory*. 01 2009. 13

BIBLIOGRAFÍA

- [11] K. W. Morton and D. F. Mayers. *Numerical solution of partial differential equations: an introduction*. Cambridge University Press, Cambridge, 2nd edition, 2005. 13, 30
- [12] Patricia Camarena. La transferencia del conocimiento: ecuaciones diferenciales parciales hacia una cuerda que vibra. pages 156–162, 2004. 14
- [13] Victor G. Ganzha and Evgenii V. Vorozhtsov. *Numerical solutions for partial differential equations: problem solving using Mathematica*. CRC Press, 1996. 16
- [14] T. Amaranath. *An Elementary Course in Partial Differential Equations*. Jones & Bartlett Learning, 2009. 16
- [15] W.A. Strauss. *Partial Differential Equations: An Introduction*. Wiley, 2008. 17, 23
- [16] John C. Strikwerda. *Finite Difference Schemes and Partial Differential Equations, Second Edition*. Society for Industrial and Applied Mathematics, 2004. 22, 28
- [17] R.C. McOwen. *Partial Differential Equations: Methods and Applications*. Tsinghua University Press, 2004. 24
- [18] Shoichiro Nakamura. *Métodos numéricos aplicados con software*. Prentice-Hall Hispanoamericana, 1992. 28
- [19] W.A. Smith and F.J.S. Bernabe. *Análisis numérico*. Prentice Hall, 1988. 28
- [20] Enrique Zuazua. Optimal and approximate control of finite-difference approximation schemes for the 1d wave equation. *Rendiconti di Matematica e delle sue Applicazioni. Serie VII*, 2004. 48
- [21] David L. Russell. Controllability and stabilizability theory for linear partial differential equations: Recent progress and open questions. *SIAM Review*, 20(4):639–739, 1978. 48
- [22] Stig Larsson and Vidar Thomée. *Partial differential equations with numerical methods*, volume 45. Springer, 2009. 48
- [23] Stefan Bilbao. *Numerical Sound Synthesis: Finite Difference Schemes and Simulation in Musical Acoustics*. Wiley Publishing, 2009.
- [24] Juan Carlos Solares Carreño and Rosalba Mendoza Suárez. Acerca de algunas soluciones de ciertas ecuaciones de onda. *Boletín de Matemáticas*, 19(2), jul. 2012.