



INSTITUTO POLITÉCNICO NACIONAL

ESCUELA SUPERIOR DE CÓMPUTO

Trabajo Terminal I

**“Desarrollo de una aplicación web para el
cálculo y graficación de series de Fourier”**

2025-A075

QUE PARA OBTENER EL GRADO DE:
Ingeniero en Sistemas Computacionales

PRESENTA:
Sebastián Morales Palacios

DIRECTORES:
Lic. Jesús Alfredo Martínez Nuño
M. en C. Gisela González Albarrán



Ciudad de México
Noviembre 2024

Documento Técnico

“Desarrollo de una aplicación web para el cálculo y graficación de series de Fourier”

Resumen - Dentro del presente trabajo se lleva a cabo el desarrollo de una aplicación web diseñada para calcular y graficar el desarrollo en series de Fourier de funciones periódicas. El desarrollo en serie puede ser tanto en forma trigonométrica como en forma exponencial compleja. Los módulos principales que contiene la aplicación son:

1. **Módulo para introducir la función** ya sea definida en una sola función o a trozos, además de su periodo. Además, se tienen opciones para ingresar extensiones de funciones de medio rango pares, impares.
2. **Módulo para el cálculo de coeficientes** mediante integración simbólica usando Maxima
3. **Módulo de graficación de funciones periódicas y coeficientes de la serie**, donde se podrá ver la aproximación de la serie a la función mientras se incrementa el número de términos de la expansión.

La aplicación pretende ser de utilidad como herramienta de apoyo para estudiantes y docentes en el tema del análisis de Fourier.

Palabras Clave - aplicación web, análisis de Fourier, cálculo, matemáticas avanzadas para la ingeniería.

Correo de Contacto -
smoralesp1700@alumno.ipn.mx

Índice general

Documento Técnico	III
1. Introducción	1
1.1. Antecedentes	1
1.2. Planteamiento del Problema	1
1.3. Justificación	2
1.4. Solución Propuesta	3
1.5. Objetivos	4
1.5.1. Objetivo General	4
1.5.2. Objetivos Específicos	4
1.6. Metodología de la Investigación	4
1.7. Estructura del documento técnico	5
2. Estado del Arte	6
2.1. Herramientas y Tecnologías Actuales	6
2.1.1. Comparativa del Funcionamiento de las Herramientas . . .	8
2.1.1.1. Cálculo de Serie de Fourier con Wolfram Alpha .	9
2.1.1.2. Cálculo de Serie de Fourier con MATLAB	11
2.1.1.3. Cálculo de Serie de Fourier con GeoGebra	11
2.1.1.4. Cálculo de Serie de Fourier con Python (Matplotlib/Manim)	11
2.2. Trabajos y Proyectos Relacionados	11
3. Marco Teórico	12
3.1. Origen e Historia	12
3.1.1. El problema de la cuerda oscilante	12
3.1.1.1. D’Alambert y Euler	13
3.1.1.2. Bernoulli	15
3.1.2. La ecuación de calor	16
3.1.2.1. Fourier	16
3.2. Series de Fourier	19
3.2.1. Funciones Periódicas	19
3.2.2. Frecuencia	21
3.2.3. Armónicos	23
3.2.4. Funciones Pares e Impares	23
3.2.5. Funciones suaves y funciones suaves a trozos	25
3.2.6. Funciones Ortogonales	26

3.2.7.	Series de Funciones	26
3.2.7.1.	Definición de Series de Funciones	26
3.2.7.2.	Convergencia de Series de Funciones	27
3.2.8.	Series de Fourier de Período T o de $-p$ a p	27
3.2.8.1.	Generalización de la Serie de Fourier	27
3.2.8.2.	Coefficientes de Fourier para Período T	27
3.2.9.	Funciones Suaves y Suaves por Partes	27
3.2.10.	Series de Fourier de Senos y Cosenos	27
3.2.10.1.	Condiciones para la Cancelación de Coeficientes	27
3.2.10.2.	Desarrollo de Series de Fourier con Senos y Cosenos	27
3.2.11.	Funciones Pares e Impares	27
3.2.11.1.	Definición y Propiedades de Funciones Pares	27
3.2.11.2.	Definición y Propiedades de Funciones Impares	28
3.2.12.	Extensiones de Medio Intervalo	28
3.2.12.1.	Extensión Periódica	28
3.2.12.2.	Extensión Par	28
3.2.12.3.	Extensión Impar	28
3.2.13.	Serie Compleja de Fourier de Período T o de $-p$ a p	28
3.2.13.1.	Definición de Serie Compleja de Fourier	28
3.2.13.2.	Coefficientes Complejos para Período T	28
3.2.14.	Serie Compleja de Fourier de 0 a p	28
3.2.14.1.	Definición y Desarrollo	28
3.2.14.2.	Aplicaciones y Ejemplos	29
3.3.	Tecnologías Utilizadas	29
3.3.1.	HTML, CSS y JavaScript	29
3.3.1.1.	HTML	29
3.3.1.2.	CSS	29
3.3.1.3.	JavaScript	29
3.3.2.	NodeJS	29
3.3.2.1.	Introducción a NodeJS	29
3.3.2.2.	Ventajas del uso de NodeJS para aplicaciones web	29
3.3.2.3.	Integración de NodeJS con Maxima	29
3.3.3.	Maxima	29
3.3.3.1.	Qué es Maxima y por qué se usa para cálculos simbólicos	29
3.3.3.2.	Implementación de Maxima en el servidor	29
3.3.4.	Angular (Posible implementación futura)	30
3.3.4.1.	Introducción a Angular	30
3.3.4.2.	Ventajas del uso de Angular para futuras versiones del proyecto	30
3.3.4.3.	Consideraciones para su implementación	30
3.3.5.	Metodologías de Desarrollo de Software	30
3.3.5.1.	Desarrollo Ágil	30
3.3.5.2.	Fases del Proyecto	30
3.3.6.	Usabilidad y Experiencia de Usuario (UX)	30
3.3.6.1.	Principios de Usabilidad	30
3.3.6.2.	Diseño de la Interfaz de Usuario (UI)	30
3.3.7.	Seguridad en Aplicaciones Web	30

3.3.7.1.	Protección contra ataques comunes	30
3.3.7.2.	Autenticación y manejo seguro de datos	31
4.	Análisis y Diseño del Sistema	32
4.1.	Metodología de Desarrollo	32
4.1.1.	¿Por qué prototipos para este trabajo?	33
4.2.	Análisis de Requerimientos	34
4.2.1.	Requerimientos Funcionales	34
4.2.2.	Requerimientos No Funcionales	38
4.3.	Análisis de Riesgos	40
4.3.1.	Identificación de Riesgos	40
4.3.2.	Jerarquización de Riesgos	40
4.4.	Estimación de Costos	40
4.4.1.	Método de Estimación de Costos	40
4.4.2.	Detalle de los Costos Estimados	40
4.5.	Diseño del Sistema	40
4.5.1.	Arquitectura General del Sistema	40
4.5.1.1.	Arquitectura Cliente-Servidor	40
4.5.1.2.	Flujo de Datos	40
4.5.2.	Diseño de la Interfaz de Usuario	40
4.5.2.1.	Estructura de la Interfaz	40
4.5.2.2.	Prototipo de la Interfaz	40
5.	Implementación	
	(Primer Prototipo)	41
A.	Cálculo de los Coeficientes de la Serie de Fourier	42
A.1.	Forma Trigonométrica	42
A.2.	Forma Exponencial Compleja	43
	Bibliografía	45

Índice de figuras

2.1. Gráfica de los coeficientes de Fourier calculados en el Apendice A <i>Fuente: Elaboración propia</i>	9
2.2. Calculo de una serie trigonométrica de Fourier en WolframAlpha .	10
2.3. Calculo de una serie trigonométrica de Fourier en WolframAlpha .	10
3.1. Ilustración del problema de la cuerda vibrante.	14
3.2. Ilustración de 3 armónicos y sus nodos de la cuerda al vibrar. . .	15
3.3. Ilustración del problema de transferencia de calor en una barra de longitud ℓ	17
3.4. Gráfica de una función periódica $f(t)$ con periodo T	20
3.5. Gráfica del área bajo la curva de $f(t)$ entre los intervalos desde a hasta $a + T$ y desde b hasta $b + T$	21
3.6. Gráfica de las funciones trigonométricas $\sin(t)$ y $\cos(t)$	21
3.7. Función $\sin(t)$ con diferentes frecuencias	22
3.8. Relación de la frecuencia angular entre la función $\sin(t)$ con el círculo unitario.	22
3.9. Una función coseno y la misma función coseno desplazada hacia la izquierda por un ángulo φ	23
3.10. Ejemplo de una función par	24
3.11. Ejemplo de una función impar	24
3.12. Comprobación de los teoremas f y g de funciones pares e impares	25
3.13. Ejemplo de una función suave	25
3.14. Ejemplo de una función suave a trozos completamente continua y una función a tozos con discontinuidades (saltos).	26
4.1. Etapas de la metodología por prototipos <i>Fuente: [20]</i>	32

Lista de tablas

2.1. Comparación de software para cálculos matemáticos y visualización de datos	8
4.1. Ventajas/Desventajas de la metodología de Prototipos Evolutivos	33
4.2. Requerimiento funcional No. 1	34
4.3. Requerimiento funcional No. 2	35
4.4. Requerimiento funcional No. 3	35
4.5. Requerimiento funcional No. 4	36
4.6. Requerimiento funcional No. 5	36
4.7. Requerimiento funcional No. 6	37
4.8. Requerimiento funcional No. 7	37
4.9. Requerimiento funcional No. 8	38
4.10. Tabla de requerimientos no funcionales	39

CAPÍTULO 1

Introducción

En este capítulo, examinaremos el problema que se abordará y los objetivos previstos para este proyecto. Además, investigaremos los productos existentes que desempeñan una función similar al proyecto que se llevará a cabo.

1.1. Antecedentes

Aquí voy a buscar los antecedentes

1.2. Planteamiento del Problema

Las series de Fourier, desde su concepción por Jean-Baptiste Joseph Fourier a principios del siglo XIX, han desempeñado un rol crucial en el análisis y la comprensión de señales y fenómenos periódicos. Las series de Fourier son parte esencial en campos como la ingeniería eléctrica, la física teórica y el procesamiento de señales, imágenes y audio, permitiendo descomponer funciones periódicas en sumas de senos y cosenos, lo que facilita su análisis y manipulación. Estas herramientas matemáticas no solo han impulsado avances significativos en la ciencia y tecnología, transformando nuestra interacción con el mundo, sino que también son cruciales en la enseñanza de los fundamentos teóricos de la ingeniería. En el ámbito de las ciencias computacionales, el estudio del análisis de Fourier se aplica a áreas vitales como la adquisición y procesamiento de señales, el procesamiento de imágenes y la robótica. Esta capacidad de simplificar señales complejas en componentes básicos no solo mejora el análisis, sino que también facilita la síntesis de nuevas tecnologías que se adaptan a necesidades y entornos cambiantes [1].

Este amplio espectro de aplicaciones destaca la importancia crítica de las series de Fourier no solo en la investigación avanzada, sino también en la formación académica de futuros ingenieros y científicos. Sin embargo, a pesar de su prevalencia en el currículo educativo, la implementación práctica de este análisis en entornos de aprendizaje a menudo revela áreas de oportunidad dentro de las herramientas disponibles, lo que impacta directamente en la eficacia con la que los estudiantes pueden aplicar y profundizar su comprensión de estos conceptos esenciales.

La necesidad de una herramienta más eficiente se hizo evidente durante los

cursos de Matemáticas Avanzadas para la Ingeniería y Procesamiento de Señales Digitales. Al enfrentar el análisis de Fourier, especialmente al resolver ejercicios sobre series de Fourier, se descubrió la falta de medios óptimos para verificar los resultados de manera directa y eficiente. A diferencia de otros cálculos matemáticos, donde las calculadoras especializadas permiten comprobaciones rápidas y fiables, las series de Fourier requerían un proceso mucho más tedioso. Para asegurar la corrección de los cálculos, era necesario utilizar software de resolución matemática para obtener los coeficientes y, posteriormente, otro software de graficación para visualizar si los resultados correspondían efectivamente a la función dada. Este desafío se ampliaba aún más cuando se presentaban variaciones mínimas en los ejercicios, como cambiar el intervalo de la función o alternar entre series trigonométricas y complejas, o incluso al aplicar extensiones pares o impares. Cada una de estas variaciones obligaba a repetir todos los pasos desde el inicio, lo que no solo consumía tiempo valioso, sino que también complicaba la gestión del tiempo disponible, especialmente cuando se tiene una carga académica intensa. Este tiempo podría utilizarse mejor en comprender los conceptos subyacentes y explorar en profundidad el por qué y el cómo de los fenómenos analizados mediante estas series.

1.3. Justificación

A pesar de la indiscutible importancia del análisis de Fourier en diversas áreas de la ciencia y la ingeniería, el acceso a herramientas que integren de manera eficiente el cálculo y la visualización de las series de Fourier aún presenta importantes oportunidades de mejora. Las herramientas actuales, aunque avanzadas y robustas, suelen fragmentar el proceso, obligando a los usuarios a utilizar distintos programas o plataformas para realizar los cálculos y la representación gráfica de los resultados. Esta división entre herramientas de cálculo y visualización añade una capa de complejidad y tiempo, especialmente para aquellos usuarios que, además de realizar cálculos precisos, requieren visualizar los resultados de manera rápida y efectiva.

En este contexto, surge la propuesta de desarrollar un prototipo de aplicación web que combine ambas funcionalidades en una sola interfaz, facilitando el proceso de resolver y graficar las series de Fourier de manera práctica, intuitiva y accesible desde cualquier dispositivo con acceso a Internet. La creación de una plataforma que integre estos aspectos tiene el potencial de mejorar significativamente la experiencia del usuario al permitirle resolver ecuaciones y obtener representaciones gráficas de las series de Fourier en un solo entorno, eliminando la necesidad de conocimientos técnicos avanzados o el uso de múltiples herramientas.

El desarrollo de esta aplicación tiene el potencial de ser un aporte valioso para diversas comunidades, como estudiantes, ingenieros y profesionales del área de las ciencias exactas, que requieren manipular y visualizar series de Fourier en sus trabajos. La integración de funcionalidades de cálculo simbólico con visualización gráfica interactiva permitirá un mayor grado de experimentación

y aprendizaje, reduciendo el tiempo y esfuerzo necesarios para comprender y analizar las transformaciones de Fourier.

Además, este proyecto representa una oportunidad para integrar y aplicar los conocimientos y habilidades adquiridos a lo largo de mi formación en ingeniería en sistemas. Al abordar tanto aspectos de programación, como de matemáticas avanzadas y principios de ingeniería de software, este trabajo terminal no solo permite abordar una necesidad práctica, sino que también sirve como un medio para demostrar la capacidad de aplicar teoría y técnicas de ingeniería en un contexto real. De esta manera, no solo se busca desarrollar una herramienta útil para otros, sino también consolidar y exhibir competencias en áreas clave de mi carrera.

1.4. Solución Propuesta

La solución que se plantea en este proyecto es el desarrollo de una aplicación web que permita el cálculo y graficación de series de Fourier de manera integrada, eficiente y accesible, cubriendo la brecha existente en las herramientas actuales, que tienden a separar el cálculo matemático de la visualización gráfica en una sola interfaz. La solución propuesta combinará estas dos funcionalidades en una sola plataforma, diseñada para ser simple de usar, práctica y accesible desde cualquier navegador web, para usuarios que deseen analizar y visualizar series de Fourier, eliminando la necesidad de usar múltiples programas para resolver y graficar las series de Fourier.

La aplicación web contará con las siguientes características clave:

- **Cálculo automático de series de Fourier para diferentes tipos de funciones:**
 - La aplicación calculará **series de Fourier** para **funciones continuas** en un intervalo o para **funciones definidas a trozos**. El usuario podrá ingresar funciones matemáticas, incluso aquellas con discontinuidades o que estén definidas por partes en distintos intervalos, y la aplicación manejará estos casos automáticamente.
 - Se implementará tanto la **serie trigonométrica** como la **serie exponencial compleja**. La serie trigonométrica descompondrá la función en términos de senos y cosenos, mientras que la versión compleja lo hará en términos de exponenciales imaginarios, lo cual es útil en el contexto de análisis más avanzado y en aplicaciones de ingeniería.
- **Extensiones de medio rango:**
 - La herramienta permitirá calcular **extensiones de medio rango** para funciones en un medio intervalo. Esto incluirá la posibilidad de obtener:
 - **Serie de senos** para funciones impares.
 - **Serie de cosenos** para funciones pares.

- **Serie completa** para funciones periódicas en un intervalo definido, permitiendo extender la función para construir series de Fourier en un medio rango específico.
- **Visualización gráfica interactiva:**
 - Una vez calculados los coeficientes de Fourier, la aplicación generará una **gráfica interactiva** que mostrará la aproximación de la serie de Fourier a la función original. El usuario podrá ajustar el número de términos de la serie para observar cómo mejora la aproximación conforme se incluyen más términos en la suma.
- **Interfaz intuitiva y amigable:**
 - La aplicación ofrecerá una interfaz sencilla y accesible, en la que el usuario podrá ingresar las funciones, definir los intervalos y elegir el tipo de serie de Fourier que desea calcular. La experiencia estará diseñada para minimizar la curva de aprendizaje, permitiendo a usuarios sin experiencia en programación obtener resultados rápidamente.

1.5. Objetivos

1.5.1. Objetivo General

Desarrollar un prototipo de una aplicación web capaz de calcular las series de Fourier en su forma trigonométrica o exponencial compleja de una función continua en un intervalo o definida a trozos, así como ser capaz de obtener la extensión par o impar de dicha función y, finalmente, graficar tanto la función original como la función expandida como una serie de Fourier y poder añadir o eliminar términos de dicha forma para apreciar como esta se aproxima a la función original.

1.5.2. Objetivos Específicos

- Implementar una interfaz de usuario que permita ingresar la función continua o a trozos, seleccionar intervalos y definir el tipo de serie de Fourier (trigonométrica o exponencial) o tipo de expansión (par o impar).
- Implementar los módulos para el cálculo de coeficientes de la serie trigonométricas y serie exponencial compleja, adaptándose a funciones continuas o definidas a trozos.
- Desarrollar un módulo de visualización que permita graficar tanto la función original como la aproximación de la serie de Fourier. Este módulo debería incluir opciones para añadir o eliminar términos y visualizar cómo estas modificaciones afectan la aproximación a la función original

1.6. Metodología de la Investigación

Para el desarrollo de este proyecto se utilizará la metodología por prototipos...

1.7. Estructura del documento técnico

Al final voy a detallar como es que está estructurado todo el documento del TT

CAPÍTULO 2

Estado del Arte

Para la correcta comprensión del trabajo presente, se necesita conocer el estado actual de las herramientas y tecnologías disponibles para el cálculo y la visualización de series de Fourier, así como los estudios y proyectos previos que abordan la implementación de soluciones similares. En este sentido, se revisarán diversas plataformas de uso común que permiten realizar estos procesos de manera separada, además de destacar trabajos académicos relacionados que aportan al desarrollo de herramientas educativas y matemáticas interactivas. Esta revisión permitirá contextualizar la propuesta de una aplicación web que integre ambas funcionalidades en una sola plataforma.

2.1. Herramientas y Tecnologías Actuales

En esta sección se presentarán las principales herramientas tecnológicas utilizadas para el cálculo de series de Fourier y su visualización gráfica. Podemos verlos en la siguiente tabla:

SOFTWARE	CARACTERÍSTICAS	PRECIO
Wolfram Alpha	Se trata de un potente motor comercial de conocimiento computacional para cálculos matemáticos y gráficos. Contiene módulos que permiten el calculo de los coeficientes de Series de Fourier (Trigonométrica y compleja) así como las extensiones pares e impares para funciones simples o a trozos, así como poder expandir la serie y obtener una gráfica estática. [2]	Desde MXN \$1,200.00 anuales para estudiantes, plan gratuito limitado.

Continúa en la siguiente página

Tabla 2.1 – continuación

SOFTWARE	CARACTERÍSTICAS	PRECIO
Geogebra	Herramienta dinámica para construcciones geométricas y gráficas. Podemos graficar cualquier serie o extensión trigonométrica y poder ver como cambia la serie conforme añadimos más coeficientes. Requiere que hagamos los cálculos y armar la expresión de la serie [3]	Software libre y código abierto
Desmos	Similar a Geogebra es una calculadora gráfica en línea para cálculos y gráficos, donde de igual modo podemos graficar cualquier serie o extensión trigonométrica y poder ver como cambia la serie conforme añadimos más coeficientes. Requiere cálculos previos para series de Fourier. [4]	Software libre y código abierto
Python _{Manim}	Librería de animación en Python para visualizaciones matemáticas, incluida la animación de series de Fourier. Podemos graficar cualquier serie ya sea trigonométrica o exponencial y poder animar de infinitas maneras el como se aproxima la serie con sus coeficientes. Requiere hacer los cálculos, armar la expresión de la serie y tener conocimientos en Python, programación orientada a objetos y a librería de Manim [5]	Software libre y código abierto
Python _{SymPy}	Es una biblioteca de Python para realizar matemáticas simbólicas, que incluye el cálculo de series de Fourier. A menudo se usa en combinación con librerías de visualización como Matplotlib para representar gráficamente los resultados, pero de nuevo, esta combinación requiere conocimientos de programación. [6]	Software libre y código abierto

Continúa en la siguiente página

Tabla 2.1 – continuación

SOFTWARE	CARACTERÍSTICAS	PRECIO
Matlab	Es un entorno de programación comercial para cálculos numéricos y visualización de datos, con herramientas específicas para series de Fourier, además de tener funciones para poder graficarlas. Requiere conocimientos en matlab. [7]	Desde USD\$99 (aprox. MXN\$1627.82) anuales para estudiantes.

Tabla 2.1: Comparación de software para cálculos matemáticos y visualización de datos

2.1.1. Comparativa del Funcionamiento de las Herramientas

A continuación, se procederá a calcular la serie de Fourier para la función $f(x) = x$ en el intervalo de $-\pi$ a π utilizando cada una de las herramientas previamente descritas. El cálculo se realizará en su **forma trigonométrica** y, en los casos en que la herramienta lo permita, también se obtendrá la **forma exponencial compleja**. Asimismo, se graficará la serie de Fourier en las plataformas que lo permitan, lo que nos permitirá comparar tanto el proceso como los resultados obtenidos en cada herramienta.

Esta comparación servirá para identificar las capacidades, ventajas y limitaciones de cada una de las plataformas en el contexto del cálculo y la visualización de series de Fourier, evaluando también su facilidad de uso y precisión en la representación gráfica.

Primeramente, calcularemos la serie nosotros para hacer una comparativa con los resultados dados por los softwares:

Los coeficientes de Fourier para la función $f(x) = x$, en el intervalo $[-\pi, \pi]$, son los siguientes (ver Apéndice A para los cálculos completos):

- ****Forma Trigonométrica****:

$$a_0 = 0, \quad a_n = 0, \quad b_n = \frac{2(-1)^{n+1}}{n}$$

Por lo tanto, la serie trigonométrica de Fourier para $f(x) = x$ es:

$$f(x) = 2 \sum_{n=1}^{\infty} \frac{(-1)^{n+1}}{n} \sin(nx)$$

- ****Forma Exponencial Compleja****:

$$c_n = \frac{i(-1)^n}{n}, \quad \text{para } n \neq 0.$$

Por lo tanto, la serie exponencial compleja de Fourier para $f(x) = x$ es:

$$f(x) = \sum_{\substack{n=-\infty \\ n \neq 0}}^{\infty} \frac{i(-1)^n}{n} e^{inx}$$

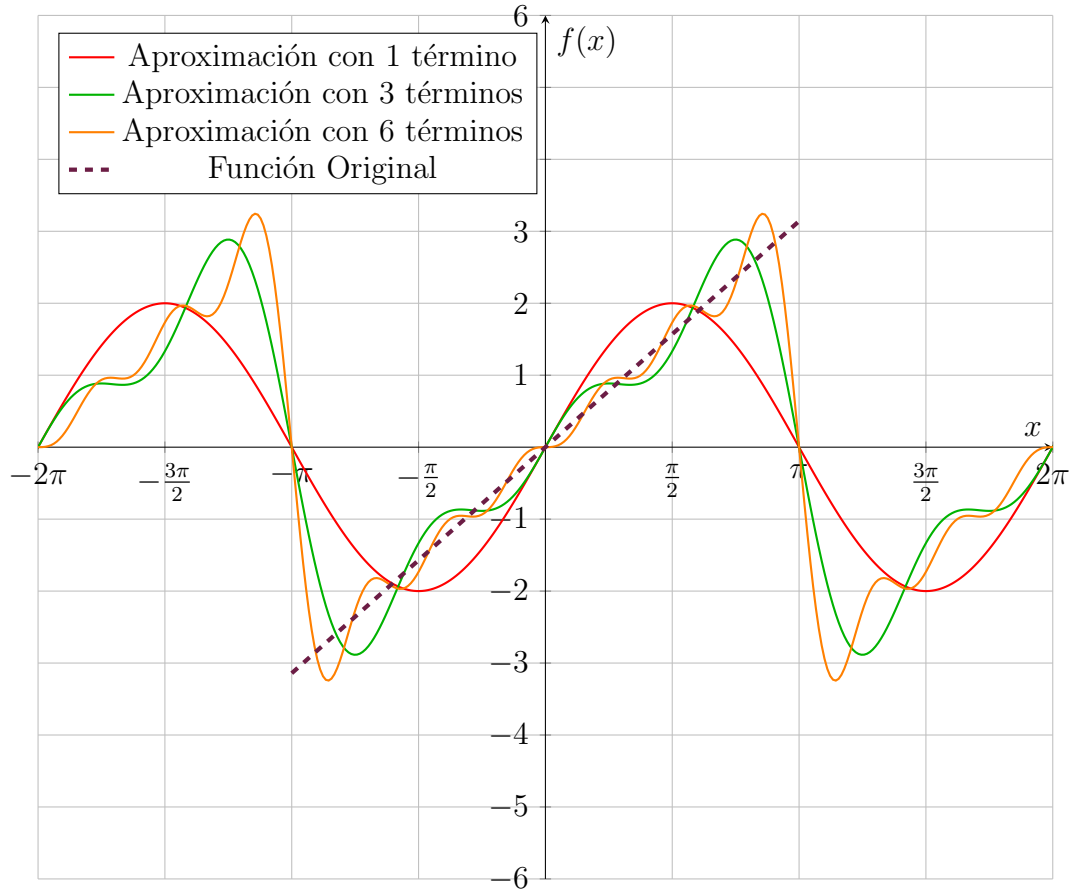


Figura 2.1: Gráfica de los coeficientes de Fourier calculados en el Apendice A
Fuente: *Elaboración propia*

Podemos ver en la Figura 2.1, la gráfica de la función y sus aproximaciones de Fourier, donde estas aproximaciones son equivalentes entre la serie trigonométrica y la serie exponencial

2.1.1.1. Cálculo de Serie de Fourier con Wolfram Alpha

WolframAlpha nos permite calcular series de Fourier trigonométricas y exponenciales complejas, además de que nos permite calcular extensiones de medio rango periódicas pares, impares y periódicas. Para nuestra primer prueba, calcularemos la serie de Fourier trigonométrica para la función $f(x) = x$ en el intervalo de $-\pi$ a π , para hacerlo usaremos la función *FourierTrigSeries*(*exp*, *t*, *n*) de WolframAlpha, que nos permite calcular la serie trigonométrica de Fourier de la función *exp* respecto a la variable *t* obteniendo la serie hasta el término *n*.

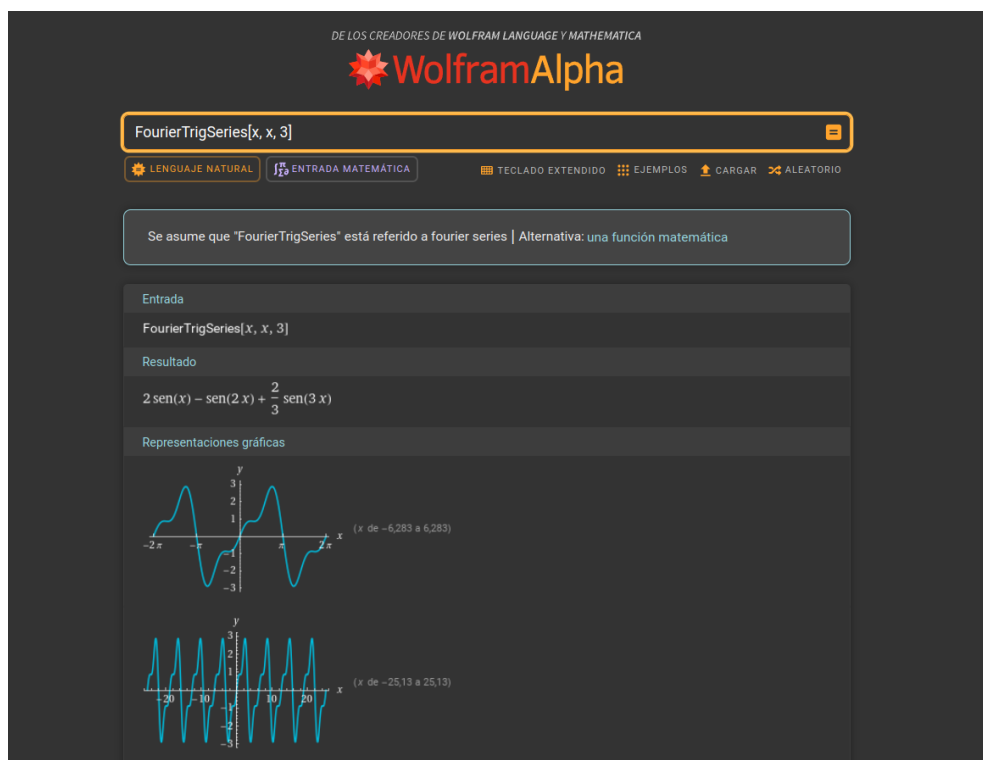


Figura 2.2: Calculo de una serie trigonométrica de Fourier en WolframAlpha

Ahora usaremos la función de $FourierSeries(exp, t, n)$ de WolframAlpha, esta función no permite calcular la serie exponencial compleja de Fourier de la función exp respecto a la variable t obteniendo la serie hasta el término n .

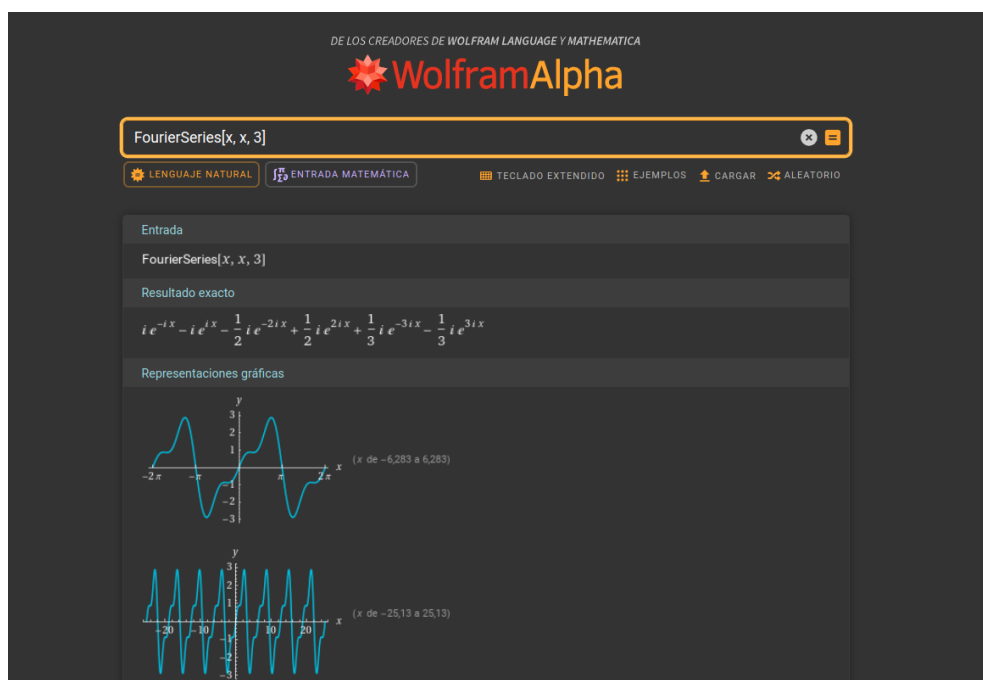


Figura 2.3: Calculo de una serie trigonométrica de Fourier en WolframAlpha

Wolfram nos devuelve la expansión de cada serie en su forma mas reducida, ademas de dos pequeñas gráficas de nuestra aproximación vista desde no nos

devuelve los coeficientes de la serie ni tampoco su expresión final en notación de serie.

2.1.1.2. Cálculo de Serie de Fourier con MATLAB

Matlab

2.1.1.3. Cálculo de Serie de Fourier con GeoGebra

Aquí se describirá cómo se realiza el cálculo y la graficación de la serie de Fourier utilizando GeoGebra...

2.1.1.4. Cálculo de Serie de Fourier con Python (Matplotlib/Manim)

Aquí se describirá cómo se realiza el cálculo y la graficación de la serie de Fourier utilizando Python con las librerías Matplotlib o Manim...

2.2. Trabajos y Proyectos Relacionados

En esta sección se analizarán trabajos académicos y proyectos de investigación que han desarrollado herramientas similares o que abordan la enseñanza de series de Fourier y el uso de plataformas interactivas para la educación matemática.

CAPÍTULO 3

Marco Teórico

En este capítulo se presentará la teoría fundamental necesaria para el desarrollo de nuestra calculadora de series de Fourier. Comenzaremos explorando la historia y evolución de las series de Fourier, se detallarán las fórmulas matemáticas necesarias para comprender la descomposición de funciones periódicas en sumas de senos y cosenos, resaltando los conceptos de coeficientes de Fourier. Además, se analizarán las tecnologías que emplearemos en la implementación del proyecto, para asegurar un rendimiento óptimo en los cálculos y visualización de resultados.

3.1. Origen e Historia

Uno de los problemas del que se ocuparon los matemáticos del siglo XVIII es el que se conoce con el nombre del *problema de la cuerda vibrante*. Este problema fue estudiado por D'Alembert y Euler (usando el método de propagación de las ondas) y un poco más tarde, concretamente en 1753, por Daniel Bernoulli. La solución dada por este difería de la proporcionada por los anteriores y consistió básicamente en expresar la solución del problema como superposición (en general infinita) de ondas sencillas.

Las ideas de Bernoulli fueron aplicadas y perfeccionadas por Fourier, en 1807, en el estudio de problemas relacionados con la conducción del calor. Quedaron plasmadas por escrito en el libro clásico *Théorie analytique de la chaleur*, publicado en 1822. Los razonamientos realizados por Fourier en este libro plantearon de manera inmediata numerosas controversias y cuestiones que han tenido una influencia significativa en la historia de la Matemática [8].

3.1.1. El problema de la cuerda oscilante

Uno de los problemas más interesantes que abordaron los científicos del siglo XVIII, y que aparece con frecuencia en problemas físicos relacionados con procesos oscilatorios, es el conocido como *problema de la cuerda vibrante*. Este se puede describir en su forma más elemental de la siguiente manera: Supongamos que tenemos una cuerda flexible y tensa, cuyos extremos están fijos, convenientemente, en los puntos $(0, 0)$ y $(\ell, 0)$ sobre el eje horizontal. Si la cuerda se tira de modo que su forma inicial corresponde a la curva definida por $y = f(x)$, y luego se suelta, la pregunta es: ¿Cuál será el movimiento resultante de la cuerda?

Los desplazamientos de la cuerda siempre se encuentran en un mismo plano, y el vector desplazamiento es perpendicular en cualquier momento. Para describir este movimiento se utiliza una función $u(x, t)$, donde $u(x, t)$ representa el desplazamiento vertical de la cuerda en la posición x (con $0 \leq x \leq \ell$) y en el instante t (con $t \geq 0$). El problema es determinar $u(x, t)$ a partir de $f(x)$ [8].

3.1.1.1. D’Alambert y Euler

El primer matemático que propuso un modelo adecuado para este problema fue Jean Le Rond D’Alambert [9]. Bajo diversas hipótesis (asumiendo, por ejemplo, que las vibraciones son ”pequeñas”), en 1747 D’Alambert dedujo la ecuación de onda en la siguiente forma (usando notación moderna):

$$\frac{\partial^2 u(x, t)}{\partial t^2} = \frac{\partial^2 u(x, t)}{\partial x^2}, \quad 0 < x < \ell, \quad t > 0 \quad (3.1)$$

en donde:

- $u(x, t)$: Es la función que describe el desplazamiento de la onda en función de la posición x y el tiempo t .
- $\frac{\partial^2 u(x, t)}{\partial t^2}$: Segunda derivada parcial de $u(x, t)$ respecto al tiempo t , que representa la aceleración del desplazamiento de la onda.
- $\frac{\partial^2 u(x, t)}{\partial x^2}$: Segunda derivada parcial de $u(x, t)$ respecto a la posición x , que describe la curvatura espacial de la onda.

y esta debe satisfacer las condiciones siguientes:

$$\begin{aligned} u(x, 0) &= f(x), \quad 0 \leq x \leq \ell \\ \frac{\partial u(x, 0)}{\partial t} &= 0, \quad 0 \leq x \leq \ell \\ u(0, t) &= u(\ell, t) = 0, \quad t \geq 0. \end{aligned} \quad (3.2)$$

En donde la primera ecuación establece la posición inicial de la cuerda, mientras que la segunda indica que la velocidad inicial de la cuerda es cero (recordando que, una vez desplazada a la posición $f(x)$, la cuerda es liberada). La última condición expresa que, para cualquier tiempo, los extremos de la cuerda permanecen fijos. En la figura 3.1 La variable $u = u(x, t)$ mide el desplazamiento sobre la vertical a tiempo $t > 0$ en la posición $x \in [0, \ell]$.

Y a partir de esto, construyó la solución [10]:

$$u(x, t) = F(x + t) + G(x - t) \quad (3.3)$$

Que, estando sujeta a las condiciones de frontera 3.2 se podía reducir a [10]:

$$u(x, t) = F(x + t) + F(x - t) \quad (3.4)$$

siempre y cuando la función F fuese periódica, impar y diferenciable en todas partes. D’Alambert demostraba con esto que la ecuación de onda admitía “un

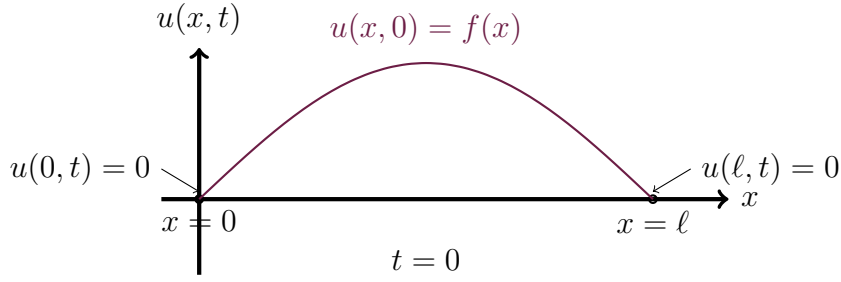


Figura 3.1: Ilustración del problema de la cuerda vibrante. *Fuente: Elaboración propia*

número infinito de soluciones”.

Euler entra en escena al criticar varias de las suposiciones que D’Alembert hiciera para encontrar su solución. Una de particular interés es la de suponer que la forma inicial de la curva o cuerda era continua y diferenciable “como una anguila”. Esta suposición para Euler, le quitaba generalidad a la solución de D’Alembert pues omitía funciones, notablemente la que representa una cuerda al ser pulsada, como se muestra en la figura 3.1. Euler derivó una ecuación de onda ligeramente más general que la de D’Alembert 3.1:

$$\frac{1}{c^2} \frac{\partial^2 u(x, t)}{\partial t^2} = \frac{\partial^2 u(x, t)}{\partial x^2} \quad (3.5)$$

con la solución [10]:

$$u(x, t) = F(x + ct) + G(x - ct) \quad (3.6)$$

que con las mismas condiciones de frontera resulta en [10]:

$$u(x, t) = F(x + ct) + F(x - ct) \quad (3.7)$$

Euler argumentaba, a diferencia de D’Alembert, que la función F quedaba determinada por la posición y velocidad inicial de la cuerda. Si denotamos por $w(x)$ y $v(x)$ a éstas, respectivamente, entonces la solución puede representarse como:

$$u(x, t) = \frac{1}{2} \left(w(x + ct) + w(x - ct) + \frac{1}{c} \int_{x-ct}^{x+ct} v(s) ds \right) \quad (3.8)$$

Las funciones w y v podían ser arbitrarias (en el sentido que podían representar cualquier curva que pudiera “dibujarse a mano”) e incluía, implícitamente, a la curva “triangular” de la figura. Como dicen los autores [11] a D’Alembert no le quitaba el sueño sacrificar realismo físico por pureza matemática; Euler, por el contrario, usaba la realidad física para contraargumentar la generalidad de las soluciones obtenidas por el francés aunque su interés no fuera tanto realismo físico sino la generalidad de las soluciones admitidas por la ecuación de onda.

3.1.1.2. Bernoulli

Otra manera de obtener la solución del problema 3.1, distinta a la de sus distinguidos interlocutores, fue propuesta por Daniel Bernoulli en 1753 [11] La idea clave es obtener la solución de 3.1 como superposición de ondas más sencillas, concretamente aquellas que son de la forma [12]:

$$u(x) = \alpha \sin \frac{\pi x}{\ell} + \beta \sin \frac{2\pi x}{\ell} + \gamma \sin \frac{3\pi x}{\ell} + \delta \sin \frac{4\pi x}{\ell} + \dots \quad (3.9)$$

que incorporaban el hecho experimental de que cualquier modo de vibración de una cuerda puede obtenerse superponiendo modos vibratorios simples representados por cada término. Estas funciones representan, para $n = 1$, el modo fundamental, y para $n > 1$, sus armónicos. De esta manera, cualquier vibración de la cuerda puede describirse como una superposición de estos armónicos, como se puede ver en la figura 3.2.

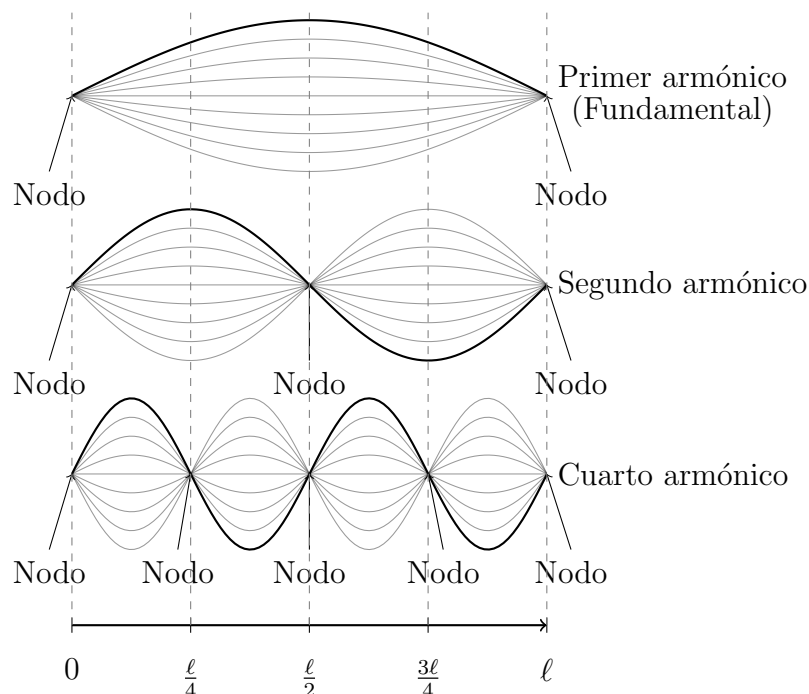


Figura 3.2: Ilustración de 3 armónicos y sus nodos de la cuerda al vibrar. *Fuente: Elaboración propia*

Si la solución propuesta por Bernoulli fuese correcta, ello obligaría a que

$$u(x, 0) = \sum_{n=1}^{\infty} C_n \sin(nx) \quad (3.10)$$

y por tanto a que

$$f(x) = \sum_{n=1}^{\infty} C_n \sin(nx), \quad \forall x \in [0, \ell], \quad (3.11)$$

para “una adecuada elección de los coeficientes C_n ”.

Pero esta fórmula no es correcta del todo pues le falta, a cada término, el producto por la función coseno (función del tiempo). Más aún, Bernoulli fue incapaz de mostrar cómo podían obtenerse los coeficientes $\alpha, \beta, \gamma, \delta, \dots$ cosa que sí hizo Euler un poco después con el método de integrar la expansión después de multiplicarla por senos o cosenos. Es claro que la solución de Bernoulli no es tan general como la de D'Alembert pero algo que aún el gran Euler no pudo realmente entender es que la solución de Bernoulli permitía funciones iniciales más generales que, a decir de otros autores, fue tema de críticas por parte de Euler a la propuesta de D'Alembert [12]. Bernoulli y Euler mantuvieron una correspondencia epistolar productiva, y se puede afirmar que ambos dependían del otro para realizar su trabajo: Bernoulli necesitaba de Euler para orientar sus estudios matemáticos, mientras que Euler dependía de Bernoulli para comprender los fenómenos físicos que sustentaban su investigación matemática. A pesar de que partían de estudios matemáticos comunes, al final no lograron resolver ni juntos, ni de manera individual, ni desde una perspectiva experimental, donde la matemática jugara un papel metodológico predominante, ni desde el punto de vista matemático donde la física del problema guiara y definiera la teoría. Euler nunca consideró especialmente relevante el trabajo matemático de Daniel Bernoulli, ya que sus métodos eran algo imprecisos, aunque con gran intuición física. Por su parte, Bernoulli tampoco valoraba mucho los estudios matemáticos de Euler, pues estaban distantes de los experimentos. Bernoulli incluso llegó a comentar que Euler "tomaba sus pruebas de la naturaleza y no de algún principio de análisis". [12]

3.1.2. La ecuación de calor

Hubo que esperar 54 años hasta que las ideas de Bernoulli fueron tomadas en cuenta por el barón Jean Baptiste-Joseph Fourier, matemático y físico francés, quien, entre otras actividades acompañó a Napoleón, en calidad de científico, en la campaña de éste en Egipto. Allí, como secretario del "Instituto de Egipto", hizo gala de gran competencia en diversos asuntos administrativos. [1]

3.1.2.1. Fourier

Al regresar a Francia, y como profesor de Análisis de la Escuela Politécnica, Fourier se interesó por la teoría de la conducción del calor en los cuerpos sólidos. En 1807 envió un artículo a la Academia de Ciencias de París, que trataba sobre dicho tema. Más concretamente, Fourier consideró una varilla delgada de longitud dada ℓ , cuyos extremos se mantienen a 0° centígrados y cuya superficie lateral está aislada. Si la distribución inicial de temperatura en la varilla viene dada por una función $f(x)$ (se supone que la temperatura de la varilla en cada sección transversal de la misma es constante), ¿Cuál será la temperatura de cualquier punto x de la varilla en el tiempo t ? Suponiendo que la varilla satisface condiciones físicas apropiadas, Fourier demostró que si $u(x, t)$ representa la temperatura en la sección x y en el tiempo t , entonces la función u debe tener la siguiente forma:

$$\frac{\partial u(x, t)}{\partial t} = \alpha^2 \frac{\partial^2 u(x, t)}{\partial x^2}, \quad 0 < x < \ell, \quad 0 < t < \infty \quad (3.12)$$

en donde:

- $\frac{\partial u}{\partial t}(x, t)$: Representa la derivada parcial de u con respecto al tiempo t , es decir, cómo cambia la temperatura $u(x, t)$ en el punto x a lo largo del tiempo t .
- $u(x, t)$: Es la función que describe la temperatura en un punto x en el espacio y en un tiempo t . Depende tanto de la posición espacial x como del tiempo t .
- α^2 : Es el coeficiente de difusión térmica, que depende del material en cuestión. Este coeficiente es constante y está relacionado con la capacidad del material para difundir el calor. La unidad de α es metros cuadrados por segundo (m^2/s).
- $\frac{\partial^2 u}{\partial x^2}(x, t)$: Representa la derivada parcial segunda de u con respecto a la posición x , es decir, cómo cambia la pendiente (o curvatura) de la temperatura a lo largo del espacio. Esta cantidad indica cómo el calor se distribuye espacialmente.

Esta ecuación modela la difusión del calor en un medio a lo largo del tiempo. La derivada en el tiempo $\left(\frac{\partial u}{\partial t}\right)$ está relacionada con la derivada segunda en el espacio $\left(\frac{\partial^2 u}{\partial x^2}\right)$, lo que refleja que el cambio en la temperatura con el tiempo depende de cómo está distribuido el calor en el espacio. y esta debe satisfacer las siguientes condiciones:

$$\begin{aligned} u(0, t) &= u(\ell, t) = 0, & 0 \leq t \leq \infty \\ u(x, 0) &= f(x), & 0 \leq x \leq \ell. \end{aligned} \quad (3.13)$$

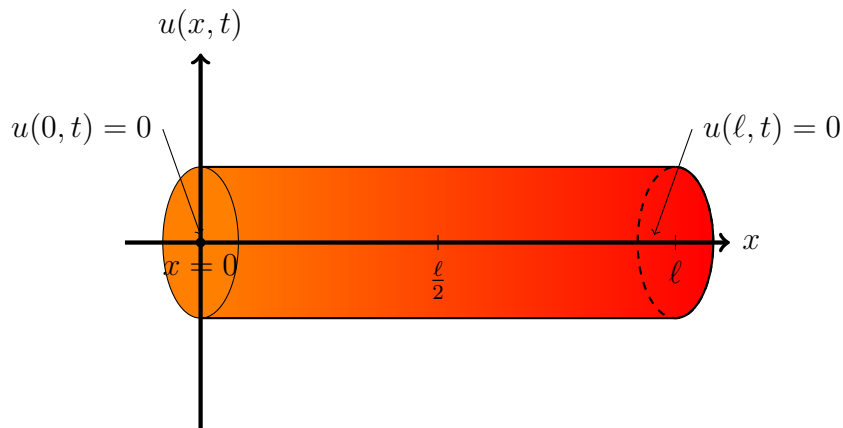


Figura 3.3: Ilustración del problema de transferencia de calor en una barra de longitud π . Fuente: *Elaboración propia*

La primera condición en 3.12 es una Ecuación en Derivadas Parciales de segundo orden, conocida con el nombre de Ecuación del Calor. La segunda significa que la temperatura, en los extremos de la varilla, se mantiene a 0° centígrados en cualquier tiempo, mientras que la última relación representa la distribución inicial de temperatura en la varilla considerada.

Partiendo de las ideas de Bernoulli, para la ecuación de ondas, Fourier buscó las soluciones más sencillas que puede presentar la ecuación del calor: aquellas que son de la forma:

$$u(x, t) = X(x)P(t) \quad (3.14)$$

Imponiendo la condición de que tales funciones satisfagan, formalmente, dicha ecuación, obtenemos, como en el caso de la ecuación de ondas, los dos problemas siguientes de ecuaciones diferenciales ordinarias [13]:

$$\begin{aligned} X''(x) + \mu X(x) &= 0, & x \in (0, \ell), & X(0) = X(\ell) = 0 \\ P'(t) + \mu P(t) &= 0, & t > 0 \end{aligned} \quad (3.15)$$

Utilizando el método de separación de variables [14], Fourier propuso que la solución a la ecuación del calor podría expresarse como el producto de dos funciones, una dependiente de la posición $X(x)$ y otra del tiempo $P(t)$, como se indica en (3.13). Esto descompone la ecuación en dos ecuaciones diferenciales ordinarias: una espacial (3.15) y una temporal (3.15). Fourier resolvió cada una de estas ecuaciones por separado. La solución espacial $X(t)$ resulta en una serie de funciones sinusoidales que satisfacen las condiciones de frontera en los extremos de la varilla, mientras que la solución temporal $P(t)$ resulta en exponentes negativos que representan la disipación de calor en el tiempo. Así, disponemos de un procedimiento que nos permite calcular infinitas “soluciones elementales” de la ecuación del calor, a saber, las funciones de la forma $b_n v_n$, donde v_n se define como:

$$v(x, t) = \sin\left(\frac{n\pi}{\ell}x\right) e^{-\alpha^2 \frac{n^2 \pi^2}{\ell^2} kt} \quad (3.16)$$

Es trivial que, si la distribución inicial de temperatura, f , es algún múltiplo de $\sin(nx)$ (o una combinación lineal finita de funciones de este tipo), entonces la solución buscada de (3.16) es un múltiplo adecuado de v_n (respectivamente, una adecuada combinación lineal de funciones de esta forma).

Ahora bien, $f(x)$ no es, en general, de la forma justo mencionada, pero, y aquí demostró Fourier, como Bernoulli, una enorme intuición, ¿Será posible obtener la solución $u(x, t)$ de (3.16), para cualquier $f(x)$ dada, como superposición de las anteriores soluciones sencillas v_n ? Es decir, ¿Será posible elegir adecuadamente los coeficientes b_n tal que la única solución de (3.16) sea de la forma:

$$u(x, t) = \sum_{n=1}^{\infty} b_n \sin\left(\frac{n\pi}{\ell}x\right) e^{-\alpha^2 \frac{n^2 \pi^2}{\ell^2} kt} \quad (3.17)$$

Fourier afirmó en su artículo que esto era correcto. Observemos que nuevamente llegamos a que, entonces, se debe satisfacer la relación (3.9). Esto plantea la misma cuestión para dos problemas completamente distintos, el problema (3.1) y el problema (3.12), en donde los coeficientes b_n toman la forma:

$$b_n = \frac{2}{\ell} \int_0^{\ell} f(x) \sin\left(\frac{n\pi x}{\ell}\right) dx \quad (3.18)$$

Los coeficientes b_n se determinan en función de la distribución inicial de temperatura $f(x)$ a través de la expresión en (3.18), lo que permite que la solución refleje la distribución inicial específica de la varilla. Así, la solución general toma la forma de una serie de Fourier, como se observa en (3.17), donde cada término de la serie combina una función sinusoidal en el espacio con un decaimiento exponencial en el tiempo.

El artículo Fourier fue evaluado por Lagrange, Laplace y Legendre y fue rechazado por la Academia Francesa, principalmente debido a la forma en que dedujo la ecuación del calor y por la falta de rigor en sus conclusiones (según la opinión de los académicos mencionados). Sin embargo, los miembros de dicha institución reconocían la relevancia de los problemas relacionados con la propagación del calor, y los resultados teóricos que Fourier presentó mostraban gran concordancia con varios experimentos realizados previamente [8].

Debido a esto, la Academia estableció un premio sobre el tema. Dicho premio fue otorgado a Fourier en 1812, pero, a pesar de esto, los académicos continuaron criticando la falta de rigor, de modo que, aunque ganó el premio, Fourier no logró publicar su trabajo en la famosa serie “*Mémoires*” de la Academia Francesa. Con gran perseverancia, Fourier continuó trabajando en el tema, y en 1822 publicó su célebre libro *Théorie Analytique de la Chaleur*, Firmin Didot, Père et Fils, 1822, París, donde incluyó gran parte de su artículo de 1812 casi sin modificaciones. Este libro es actualmente una de las obras clásicas en matemáticas [8].

Dos años después, obtuvo el puesto de Secretario de la Academia Francesa, lo que le permitió finalmente publicar su artículo en la serie “*Mémoires*” [8].

3.2. Series de Fourier

Luego de que las soluciones para la ecuación de onda y de calor fueron desarrolladas, Joseph Fourier propuso un nuevo método para expresar funciones periódicas en función de sinusoides. Las bases de lo que hoy conocemos como series de Fourier fueron establecidas por esta propuesta.

A continuación, se describen los conceptos matemáticos en los cuales se basan las series de Fourier, ofreciendo una base firme para entenderlas y aplicarlas en el estudio de funciones que se repiten periódicamente.

3.2.1. Funciones Periódicas

Una función es llamada *función periódica* si existe una constante $T > 0$ tal que

$$f(t) = f(t \pm T) \quad (3.19)$$

para todo valor de t en el dominio de definición de $f(t)$, donde la constante T se denomina *periodo* de la función. El periodo también se puede definir como el tiempo transcurrido entre dos puntos equivalentes de una función [15].

Las funciones periódicas aparecen en muchas aplicaciones de matemáticas para problemas de física e ingeniería. Es evidente que la suma, diferencia, producto o

cociente de dos funciones con período T también será una función con período T [15].

Si graficamos una función periódica $f(t)$ en un intervalo cerrado $a \leq t \leq a \pm T$, podemos obtener la gráfica completa de $f(t)$ repitiendo periódicamente la porción de la gráfica correspondiente a $a \leq t \leq a \pm T$ 3.4.

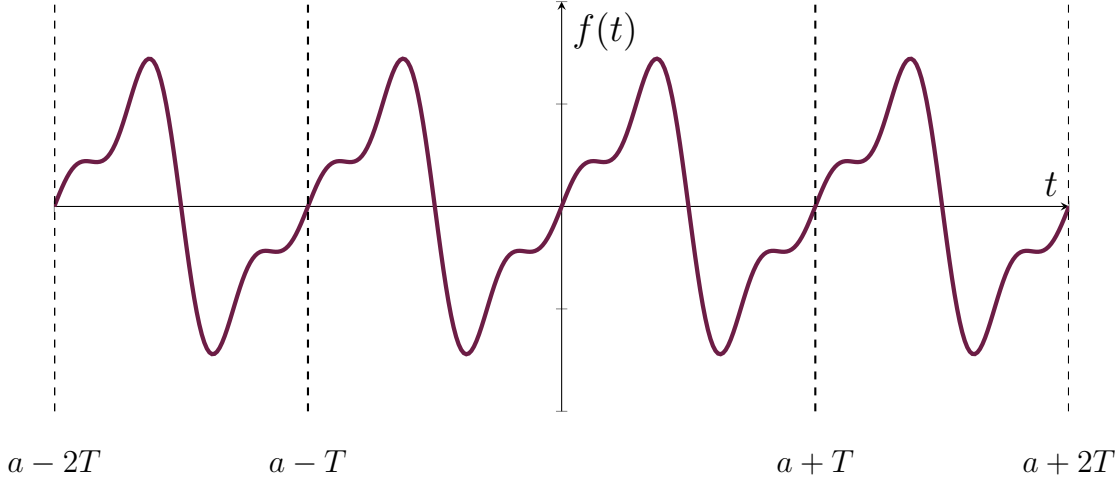


Figura 3.4: Gráfica de una función periódica $f(t)$ con periodo T . Fuente: *Elaboración propia*

Si T es un período de la función $f(t)$, entonces los números $2T$, $3T$, $4T$, ... también son períodos. Esto se deduce fácilmente al inspeccionar la gráfica de una función periódica o a partir de la serie de igualdades [16]

$$f(t) = f(t \pm T) = f(t \pm 2T) = f(t \pm 3T) = \dots f(t \pm nT), \quad n = 0, \pm 1, \pm 2, \dots \quad (3.20)$$

que se obtiene mediante el uso repetido de la condición (3.19). Así, si T es un periodo, también lo es nT , donde n es cualquier entero positivo, es decir, si existe un periodo, no es único [16]. Además, podemos definir como el *periodo fundamental* T_0 de $f(t)$ a el valor positivo más pequeño de T para el cual se satisface (3.19). [17]

Ahora, consideremos la siguiente propiedad de cualquier función $f(t)$ con período T :

Si $f(t)$ es integrable en cualquier intervalo de longitud T , entonces es integrable en cualquier otro intervalo de la misma longitud, y el valor de la integral es el mismo. Es decir,

$$\int_a^{a+T} f(t) dx = \int_b^{b+T} f(t) dt \quad (3.21)$$

para cualquier a y b .

Esta propiedad es una consecuencia directa de la interpretación del área bajo la curva como la integral. De hecho, cada integral de la ecuación anterior representa el área entre la curva $f(t)$, el eje t , y las líneas verticales en los puntos extremos del intervalo. Las áreas sobre el eje t se consideran positivas y las áreas bajo el

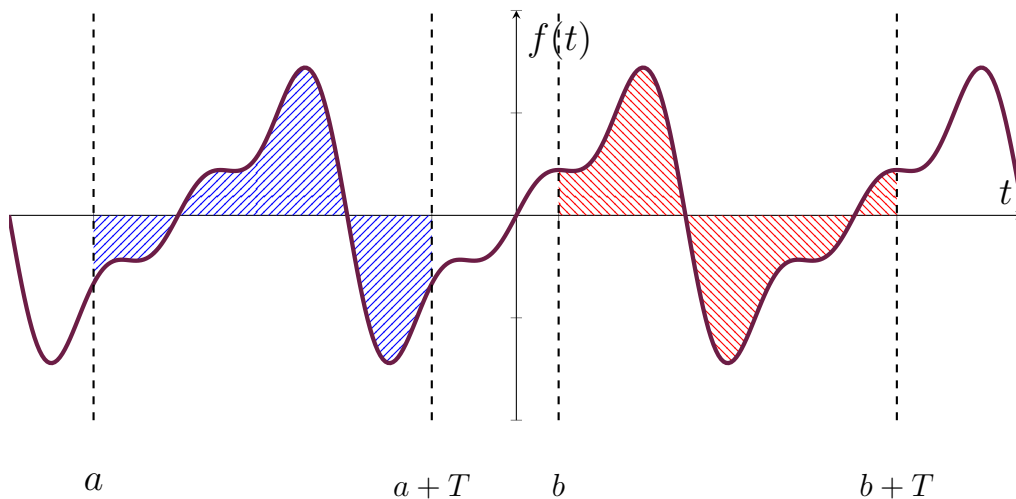


Figura 3.5: Gráfica del área bajo la curva de $f(t)$ entre los intervalos desde a hasta $a + T$ y desde b hasta $b + T$. Fuente: *Elaboración propia*

eje t como negativas. En este caso, las áreas representadas por ambas integrales son iguales debido a la periodicidad de $f(t)$ [15] 3.5.

A partir de ahora, cuando afirmemos que una función $f(t)$ de período T es integrable, queremos decir que es integrable en un intervalo de longitud T . Esto implica, a partir de la propiedad recién demostrada, que $f(t)$ es integrable en cualquier intervalo de longitud finita [15].

Las funciones periódicas más representativas son las funciones trigonométricas $\sin(t)$ y $\cos(t)$, ambas con periodo $T = 2\pi$ (Figura 3.6). Por lo tanto, de acuerdo con (3.19), se tiene que $\sin(t + 2\pi) = \sin(t)$ y $\cos(t + 2\pi) = \cos(t)$ [17].

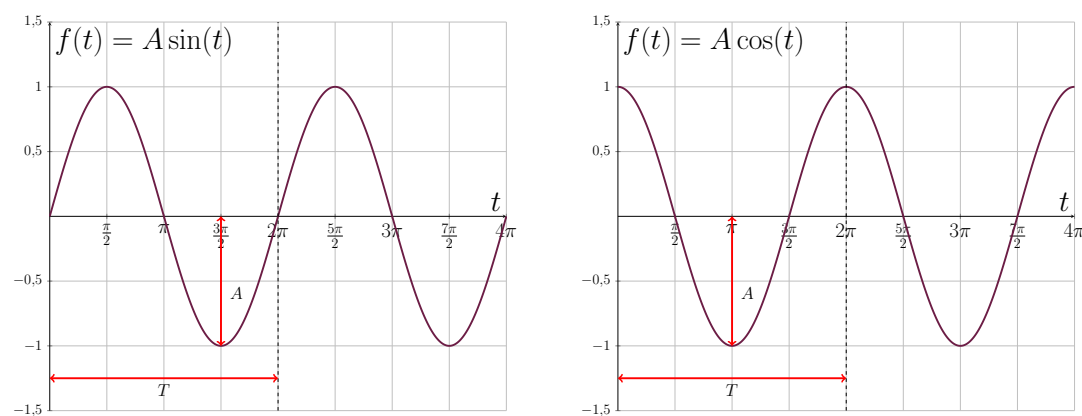


Figura 3.6: Gráfica de las funciones trigonométricas $\sin(t)$ y $\cos(t)$. Fuente: *Elaboración propia*

3.2.2. Frecuencia

La relación entre la *frecuencia* f y el *periodo* T de una función oscilatoria está dada por

$$f = \frac{1}{T} \quad (3.22)$$

lo cual indica el número de oscilaciones de la función por unidad de tiempo. La unidad de medida de la frecuencia es el hertz (Hz). Un hertz quiere decir que una función oscila una vez por segundo [17]. La Figura 3.7 muestra una función senoidal con tres frecuencias diferentes.

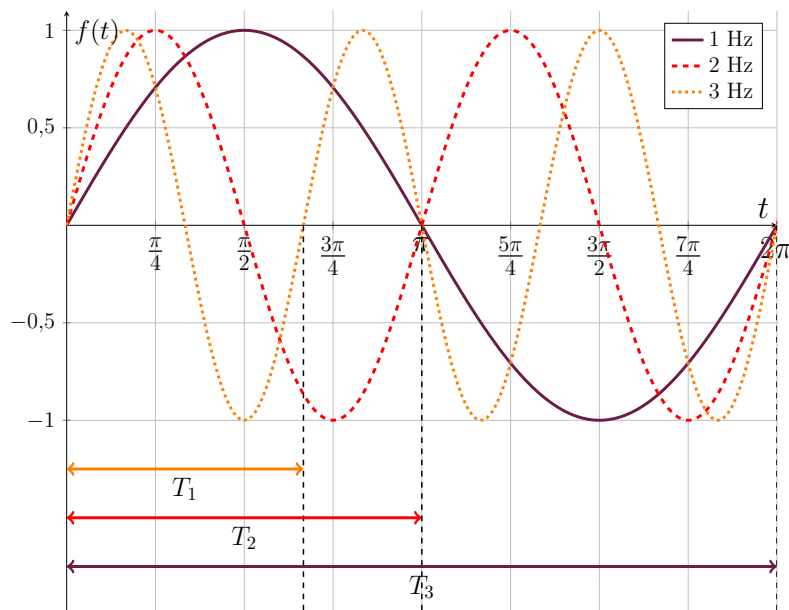


Figura 3.7: Función $\sin(t)$ con diferentes frecuencias. *Fuente: Elaboración propia*

La *frecuencia angular*, usualmente denotada por ω_0 , se define como la razón de cambio del desplazamiento angular θ durante la oscilación (o rotación) [17]:

$$\frac{d\theta}{dt} = \omega_0 = 2\pi f = \frac{2\pi}{T}. \quad (3.23)$$

La frecuencia angular se mide comúnmente en radianes por segundo (rad/s). La Figura 3.8 ejemplifica la relación de una onda senoidal en el intervalo $[0, 2\pi]$ con una frecuencia angular ω_0 . [17]

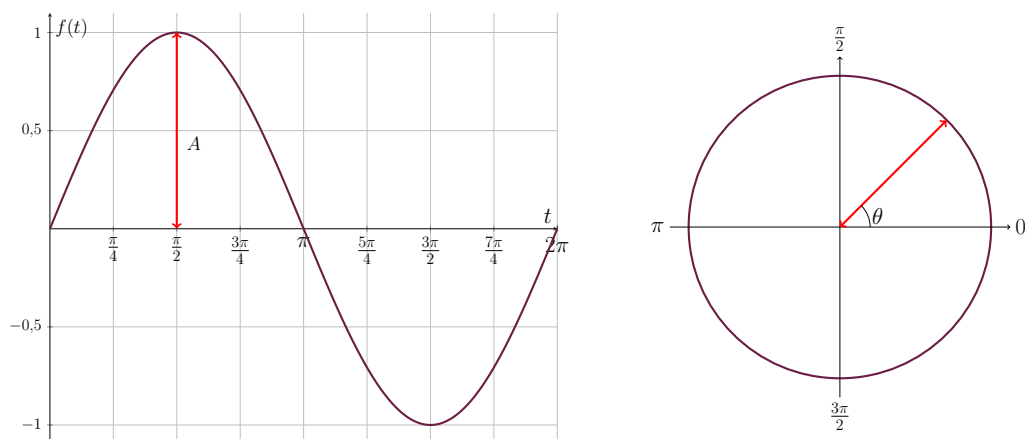


Figura 3.8: Relación de la frecuencia angular entre la función $\sin(t)$ con el círculo unitario. *Fuente: Elaboración propia*

3.2.3. Armónicos

La función periódica más simple, y la de mayor importancia para las aplicaciones, es

$$\begin{aligned} f(t) &= A \sin(\omega_0 t + \varphi) \\ f(t) &= A \cos(\omega_0 t + \varphi) \end{aligned} \quad (3.24)$$

Cabe señalar que, dado que las funciones seno y coseno solo se diferencian por un deslizamiento de fase, este movimiento podría modelarse usando la función coseno o la función seno, en donde A , ω_0 , y φ son constantes. En una función armónica, la *amplitud* A representa el valor máximo de la oscilación. La *frecuencia angular* ω_0 determina la rapidez con la que se realiza la oscilación y viene dada por $\frac{2\pi}{T}$, recordando que T es el periodo de la función, y la *fase inicial* φ indica el desplazamiento de la función en el tiempo al inicio de la observación, es decir, indica con que desfase comienza. 3.9 [18].

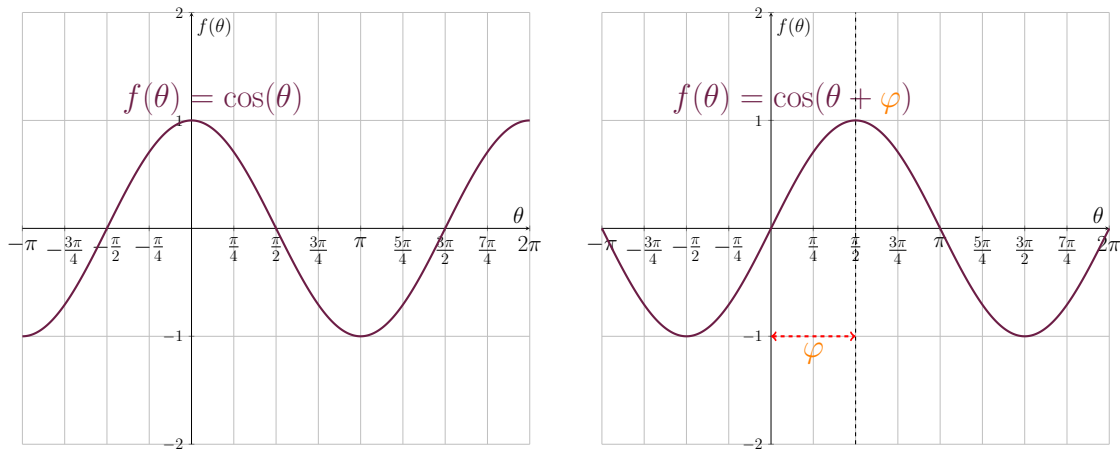


Figura 3.9: Una función coseno y la misma función coseno desplazada hacia la izquierda por un ángulo φ . Fuente: *Elaboración propia*

3.2.4. Funciones Pares e Impares

Más adelante se verá como es que la paridad o imparidad de una función $f(t)$ muchas veces puede ahorrar diversos cálculos, comenzamos definiendo que una función es par si se cumple que: [19]

$$f(t) = f(-t), \quad -L < t < L \quad (3.25)$$

En la Figura 3.10 se muestra un ejemplo de una función para en donde se puede observar la definición (3.25)

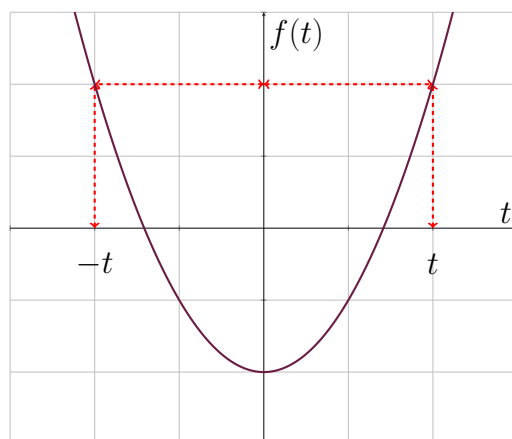


Figura 3.10: Ejemplo de una función par. *Fuente: Elaboración propia*

Ahora, decimos que una función es impar si se cumple que: [19] esquina

$$f(t) = -f(-t), \quad -L < t < L \quad (3.26)$$

En la Figura 3.11 se muestra un ejemplo de una función impar en donde se puede observar la definición (3.26)

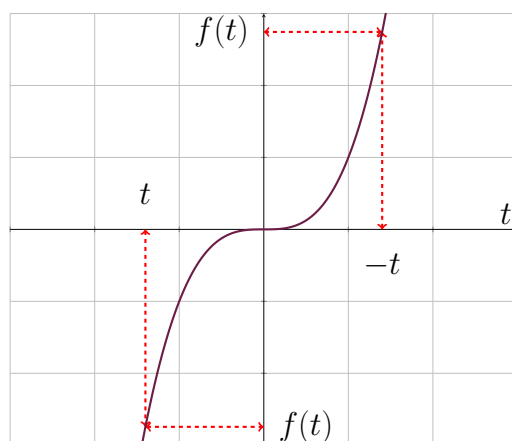


Figura 3.11: Ejemplo de una función impar. *Fuente: Elaboración propia*

Podemos decir entonces que en un intervalo simétrico $-L < t < L$, la gráfica de una función par tiene simetría respecto al eje y, mientras que la gráfica de una función impar tiene simetría en relación con el origen. Además, las funciones pares e impares cumplen con los siguientes teoremas [19]:

- El producto de dos funciones pares es par.
- El producto de dos funciones impares es par.
- El producto de una función par y una impar es impar.
- La suma (resta) de dos funciones pares es par.
- La suma (resta) de dos funciones impares es impar.

f) Si f es par, entonces $\int_{-a}^a f(t) dt = 2 \int_0^a f(t) dt$.

g) Si f es impar, entonces $\int_{-a}^a f(t) dt = 0$.

En la Figura 3.12 podemos observar como es que se cumplen los teoremas f y g, ya que en la primera el área de la función desde $-a$ hasta 0 es el negativo a su contraparte que se encuentra desde 0 hasta a , lo que significa que el área conjunta desde $-a$ hasta a se anulará dando 0, por otra parte, la segunda imagen nos muestra que el área desde $-a$ hasta 0 es la misma que la que hay desde 0 hasta a , lo que implica que el área desde $-a$ hasta a es lo doble de el área desde 0 hasta $\mp a$.

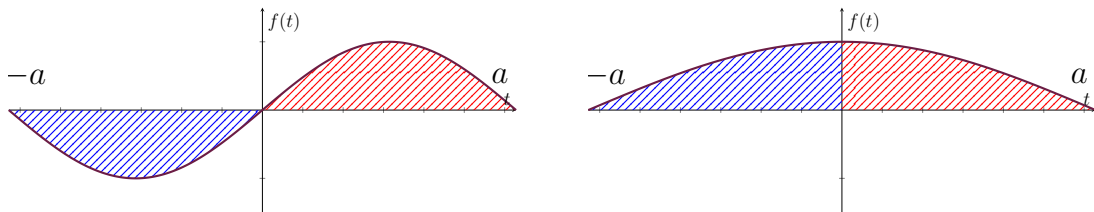


Figura 3.12: Comprobación de los teoremas f y g de funciones pares e impares
Fuente: *Elaboración propia*

3.2.5. Funciones suaves y funciones suaves a trozos

Una función $f(t)$ se dice que es *suave* en el intervalo $[a, b]$ si tiene una derivada *continua* en $[a, b]$. En términos geométricos, esto significa que la dirección de la tangente cambia de forma *continua*, sin saltos, mientras se desplaza a lo largo de la curva $f(t)$ [ver Figura 3.13]. Así, el gráfico de una función suave es una curva suave sin “saltos”. [15]

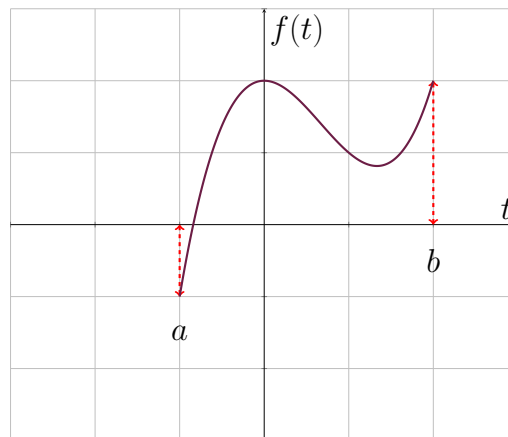


Figura 3.13: Ejemplo de una función suave. Fuente: *Elaboración propia*

La función $f(t)$ se dice que es *suave a trozos* en el intervalo $[a, b]$ si tanto $f(t)$ como su derivada son continuas en $[a, b]$, o si tienen solamente un número finito de discontinuidades de salto en $[a, b]$. Es fácil ver que el gráfico de una función suave por partes es o bien una curva continua o una curva discontinua que puede tener un número finito de saltos (en las cuales la derivada tiene saltos). A medida

que nos acercamos a cualquier discontinuidad o salto (desde un lado o el otro), la dirección de la tangente se aproxima a una posición límite definida, ya que la derivada sólo puede tener discontinuidades de salto. Las figuras 3.14 ilustran las gráficas de funciones suaves a trozos continuas y discontinuas. [15]

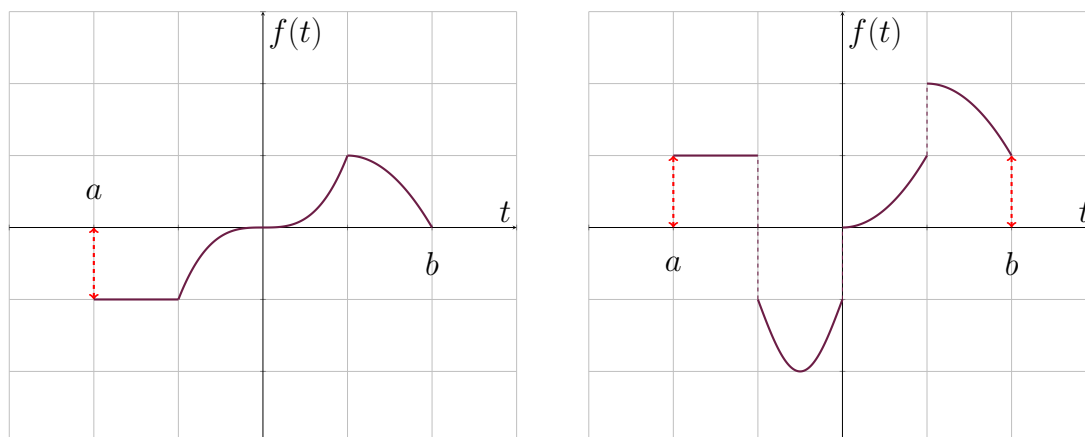


Figura 3.14: Ejemplo de una función suave a trozos completamente continua y una función a trozos con discontinuidades (saltos).]Fuente: Elaboración propia

Entonces, una función continua o discontinua $f(t)$ definida en todo el eje t se dice que es *suave a trozos* si, en cualquier intervalo finito, se comporta de manera que su derivada es continua o tiene pocas discontinuidades. Esto significa que la función puede tener ciertos "saltos." cambios bruscos en su forma, pero entre estos puntos, su comportamiento es regular y suave. Este concepto es útil para funciones periódicas, que se repiten en ciclos. Toda función $f(t)$ suave a trozos (ya sea continua o discontinua) está acotada, lo que implica que no puede crecer indefinidamente y tiene una derivada acotada en todas partes, excepto en sus "esquinas" puntos de discontinuidad (es decir, en los puntos donde la función cambia abruptamente o tiene saltos), donde la derivada $f'(t)$ no existe. [15]

3.2.6. Funciones Ortogonales

Las funciones ortogonales son fundamentales en el análisis de Fourier, ya que permiten descomponer funciones en términos independientes entre sí.

3.2.7. Series de Funciones

Una serie de funciones es una suma infinita de funciones que depende de un parámetro. Este concepto es fundamental para el desarrollo de muchas herramientas en análisis matemático.

3.2.7.1. Definición de Series de Funciones

La definición de una serie de funciones involucra una suma infinita donde cada término es una función. Estas series pueden converger a una función límite.

3.2.7.2. Convergencia de Series de Funciones

La convergencia de una serie de funciones describe cuándo y cómo la suma de los términos se aproxima a una función límite. Existen distintos tipos de convergencia, como la convergencia puntual y la convergencia uniforme.

3.2.8. Series de Fourier de Período T o de $-p$ a p

Esta sección aborda la construcción de series de Fourier para funciones periódicas en intervalos definidos.

3.2.8.1. Generalización de la Serie de Fourier

La serie de Fourier puede generalizarse a cualquier intervalo de periodo T o simétrico en $[-p, p]$.

3.2.8.2. Coeficientes de Fourier para Período T

Los coeficientes de Fourier para funciones periódicas de periodo T se calculan integrando sobre un intervalo de longitud T .

3.2.9. Funciones Suaves y Suaves por Partes

Las funciones suaves y suaves por partes tienen importancia en el análisis de Fourier por su comportamiento en el límite.

3.2.10. Series de Fourier de Senos y Cosenos

Las series de Fourier de senos y cosenos permiten descomponer funciones en términos de senos y cosenos exclusivamente.

3.2.10.1. Condiciones para la Cancelación de Coeficientes

Para que ciertos coeficientes desaparezcan en una serie de Fourier, la función debe tener propiedades particulares de simetría.

3.2.10.2. Desarrollo de Series de Fourier con Senos y Cosenos

El desarrollo de series de Fourier en senos y cosenos se basa en las propiedades de simetría de las funciones.

3.2.11. Funciones Pares e Impares

Las funciones pares e impares presentan propiedades específicas que simplifican sus desarrollos en series de Fourier.

3.2.11.1. Definición y Propiedades de Funciones Pares

Una función par es simétrica respecto al eje y ; sus series de Fourier contienen solo términos de coseno.

3.2.11.2. Definición y Propiedades de Funciones Impares

Una función impar es simétrica respecto al origen; sus series de Fourier contienen solo términos de seno.

3.2.12. Extensiones de Medio Intervalo

Las extensiones de medio intervalo son técnicas para definir una función en un dominio extendido, facilitando el análisis de Fourier.

3.2.12.1. Extensión Periódica

La extensión periódica replica la función en intervalos sucesivos, creando una función periódica.

3.2.12.2. Extensión Par

La extensión par hace que la función sea simétrica respecto al eje y , facilitando el análisis con cosenos.

3.2.12.3. Extensión Impar

La extensión impar hace que la función sea simétrica respecto al origen, facilitando el análisis con senos.

3.2.13. Serie Compleja de Fourier de Período T o de $-p$ a p

La serie compleja de Fourier representa funciones periódicas utilizando exponentes complejos, simplificando el cálculo.

3.2.13.1. Definición de Serie Compleja de Fourier

La serie compleja de Fourier usa exponentes complejos para expresar funciones periódicas de manera compacta.

3.2.13.2. Coeficientes Complejos para Período T

Los coeficientes complejos se obtienen integrando sobre un intervalo de longitud T y representan la amplitud y fase de cada frecuencia.

3.2.14. Serie Compleja de Fourier de 0 a p

Esta serie compleja se utiliza para representar funciones definidas en intervalos específicos como $[0, p]$.

3.2.14.1. Definición y Desarrollo

Se describe cómo desarrollar la serie compleja de Fourier en un intervalo de $[0, p]$, utilizando exponentes complejos.

3.2.14.2. Aplicaciones y Ejemplos

Ejemplos de aplicación de la serie compleja de Fourier en análisis de señales y sistemas periódicos.

3.3. Tecnologías Utilizadas**3.3.1. HTML, CSS y JavaScript****3.3.1.1. HTML**

Explicación sobre cómo define la estructura de la aplicación web.

3.3.1.2. CSS

Justificación para su uso en el diseño y estilo visual de la aplicación.

3.3.1.3. JavaScript

Detalle del uso de JavaScript para la interacción dinámica de la interfaz y control de eventos.

3.3.2. NodeJS**3.3.2.1. Introducción a NodeJS**

Explicación básica sobre NodeJS.

3.3.2.2. Ventajas del uso de NodeJS para aplicaciones web

Descripción de por qué se eligió NodeJS para la parte del servidor.

3.3.2.3. Integración de NodeJS con Maxima

Detalle de cómo se usa NodeJS para comunicarse con Maxima y realizar cálculos matemáticos.

3.3.3. Maxima**3.3.3.1. Qué es Maxima y por qué se usa para cálculos simbólicos**

Explicación sobre el software Maxima y su importancia en los cálculos de Series de Fourier.

3.3.3.2. Implementación de Maxima en el servidor

Cómo se utiliza Maxima en el servidor junto con NodeJS para realizar cálculos matemáticos de forma eficiente.

3.3.4. Angular (Posible implementación futura)

3.3.4.1. Introducción a Angular

Explicación básica sobre el framework Angular.

3.3.4.2. Ventajas del uso de Angular para futuras versiones del proyecto

Justificación sobre la posibilidad de usar Angular en futuras versiones del proyecto.

3.3.4.3. Consideraciones para su implementación

Consideraciones técnicas y cómo Angular podría mejorar la estructura de la aplicación.

3.3.5. Metodologías de Desarrollo de Software

-

3.3.5.1. Desarrollo Ágil

Breve descripción de la metodología ágil aplicada al desarrollo de la aplicación. -

3.3.5.2. Fases del Proyecto

Descripción de las fases de planificación, desarrollo y pruebas que seguiste.

3.3.6. Usabilidad y Experiencia de Usuario (UX)

-

3.3.6.1. Principios de Usabilidad

Breve descripción de los principios que seguiste para garantizar una buena usabilidad de la aplicación. -

3.3.6.2. Diseño de la Interfaz de Usuario (UI)

Descripción de cómo el diseño de la interfaz facilita la interacción de los usuarios con las Series de Fourier.

3.3.7. Seguridad en Aplicaciones Web

-

3.3.7.1. Protección contra ataques comunes

Breve descripción de las medidas tomadas para evitar ataques como la inyección de código y el cross-site scripting (XSS). -

3.3.7.2. Autenticación y manejo seguro de datos

Si tu aplicación maneja datos sensibles o autenticación de usuarios, menciona las técnicas utilizadas para asegurar los datos.

CAPÍTULO 4

Análisis y Diseño del Sistema

En este capítulo se evaluaron los datos del marco teórico para seleccionar los componentes más adecuados para el proyecto. También se realizó un análisis detallado de los requerimientos y riesgos asociados al desarrollo del sistema, asegurando un diseño eficiente y viable para su implementación.

4.1. Metodología de Desarrollo

La metodología a usar La metodología propuesta para el desarrollo del proyecto es Prototipos Evolutivos, en la cual se emplearán dos iteraciones mínimas. Esta metodología es adecuada para proyectos en los que la comprensión completa de los requisitos puede surgir gradualmente y en donde es importante recibir retroalimentación continua [20]. Al utilizar prototipos, se permite una aproximación más ágil al desarrollo, ya que se pueden generar soluciones tempranas que serán ajustadas en función de los comentarios de los usuarios y evaluaciones constantes.

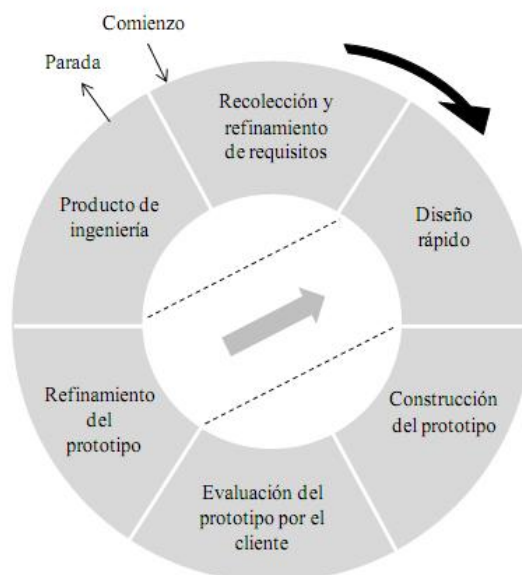


Figura 4.1: Etapas de la metodología por prototipos

Fuente: [20]

El uso de prototipos evolutivos permite abordar los requerimientos del sistema

de manera iterativa. Cada prototipo no es desechado sino que se refina progresivamente hasta obtener una versión completa del sistema. Esto asegura que el producto final no solo cumpla con los requisitos establecidos inicialmente, sino también con las expectativas del cliente o usuario.

La retroalimentación temprana que se recibe desde las primeras etapas del proyecto es invaluable para identificar mejoras y posibles problemas antes de que se conviertan en obstáculos mayores. Esto permite un ahorro significativo de tiempo y recursos, ya que se pueden realizar correcciones antes de que los problemas se propaguen en el ciclo de desarrollo.

A continuación, se muestra una tabla comparativa de ventajas/desventajas de utilizar esta metodología.

Ventajas	Desventajas
Permite obtener retroalimentación temprana y frecuente del cliente, lo que ayuda a ajustar los requerimientos de manera rápida.	Puede generar una dependencia excesiva de los prototipos, llevando a malentendidos sobre la funcionalidad final del producto.
Facilita la detección temprana de errores o problemas de diseño, lo que reduce costos de desarrollo en fases posteriores.	Requiere recursos adicionales (tiempo y esfuerzo) para desarrollar y revisar los prototipos.
Mejora la comprensión de los requisitos por parte del equipo de desarrollo y el cliente, reduciendo malentendidos.	Si no se maneja adecuadamente, puede desviar el proyecto del alcance original, añadiendo características no planificadas (creeping scope).
Aumenta la satisfacción del cliente al ver un producto tangible en las primeras fases del desarrollo.	El prototipo puede ser confundido con el producto final, lo que puede generar expectativas irreales respecto a tiempos y calidad.
Fomenta la iteración y mejora continua del producto, asegurando que se ajuste mejor a las necesidades reales del cliente.	El uso excesivo de prototipos puede ralentizar el desarrollo del producto final si las iteraciones no están bien gestionadas.

Tabla 4.1: Ventajas/Desventajas de la metodología de Prototipos Evolutivos

4.1.1. ¿Por qué prototipos para este trabajo?

Está perrona

4.2. Análisis de Requerimientos

Los requerimientos funcionales y no funcionales descritos en este documento son fundamentales para el desarrollo y éxito de la aplicación web de cálculo y visualización de Series de Fourier. Estos requerimientos definen no solo las funcionalidades que la aplicación debe cumplir, sino también las expectativas de rendimiento, seguridad y usabilidad. A través de su análisis, se asegura que el sistema propuesto sea capaz de resolver la problemática planteada de manera eficaz, proporcionando a los usuarios una herramienta confiable y eficiente para el análisis de series matemáticas.

4.2.1. Requerimientos Funcionales

Los requerimientos funcionales de un sistema describen lo que el sistema debe hacer. Describen las interacciones entre el sistema y su ambiente, en forma independiente a su implementación. El ambiente incluye al usuario y cualquier otro sistema externo con el cual interactúa el sistema.

Requerimiento	Ingreso de Funciones Matemáticas
Identificador	RF1
Prioridad de desarrollo	Alta
Entrada	Funciones matemáticas ingresadas por el usuario usando MathQuill
Salida	Funciones ingresadas en formato compatible con Maxima
Descripción	El usuario debe poder ingresar funciones matemáticas definidas en un solo trozo o en múltiples trozos usando una notación matemática intuitiva. Debe soportar funciones trigonométricas, polinomios, exponenciales y cualquier otra función que Maxima pueda interpretar.
Precondición	El usuario debe estar en el área de ingreso de funciones.
Postcondición	Las funciones son ingresadas correctamente y listas para su procesamiento.

Tabla 4.2: Requerimiento funcional No. 1

Requerimiento	Selección de Parámetros
Identificador	RF2

Prioridad de desarrollo	Alta
Entrada	Parámetros seleccionados por el usuario (tipo de serie, período T)
Salida	Parámetros listos para el cálculo de coeficientes
Descripción	Los usuarios deben poder seleccionar entre las diferentes representaciones de Series de Fourier y especificar el período T de la función, ya sea de $-T/2$ a $T/2$ o de 0 a T.
Precondición	El usuario debe haber ingresado una función matemática y estar en el área de selección de parámetros.
Postcondición	Los parámetros seleccionados están listos para el cálculo de coeficientes.

Tabla 4.3: Requerimiento funcional No. 2

Requerimiento	Cálculo de Coeficientes
Identificador	RF3
Prioridad de desarrollo	Alta
Entrada	Función matemática y parámetros seleccionados por el usuario
Salida	Coeficientes de la serie de Fourier calculados
Descripción	La aplicación debe calcular automáticamente los coeficientes de la serie de Fourier según la función ingresada y los parámetros especificados, enviando la información a Maxima y recibiendo los resultados.
Precondición	El usuario debe haber ingresado una función y seleccionado los parámetros necesarios.
Postcondición	Los coeficientes de la serie de Fourier se calculan y se devuelven al sistema.

Tabla 4.4: Requerimiento funcional No. 3

Requerimiento	Interactividad en la Gráfica
Identificador	RF4

Prioridad de desarrollo	Media
Entrada	Control del usuario sobre la gráfica (acercar, alejar, cantidad de términos n)
Salida	Gráfica interactiva modificada según la selección del usuario
Descripción	El usuario debe poder acercar, alejar y desplazar la gráfica, así como seleccionar la cantidad de términos n de la suma parcial a visualizar.
Precondición	El usuario debe haber ingresado la función y los parámetros.
Postcondición	La gráfica es ajustada interactivamente según la entrada del usuario.

Tabla 4.5: Requerimiento funcional No. 4

Requerimiento	Visualización de las Gráficas
Identificador	RF5
Prioridad de desarrollo	Alta
Entrada	Función original y términos de la Serie de Fourier
Salida	Gráficas de la función original y las sumas parciales de la Serie de Fourier
Descripción	El usuario debe poder visualizar la gráfica de la función original y las sumas parciales de la Serie de Fourier en un lienzo canvas, que se actualizará en tiempo real a medida que se agregan o quitan términos.
Precondición	El usuario debe haber ingresado la función y los parámetros necesarios.
Postcondición	La gráfica se actualiza en tiempo real según los términos seleccionados.

Tabla 4.6: Requerimiento funcional No. 5

Requerimiento	Visualización de resultados
Identificador	RF6

Prioridad de desarrollo	Alta
Entrada	Función matemática y parámetros seleccionados
Salida	Expresión matemática de la serie de Fourier y coeficientes individuales
Descripción	La aplicación debe mostrar la serie de Fourier y los valores individuales de los coeficientes en notación matemática, utilizando MathQuill.
Precondición	El usuario debe haber ingresado la función y los parámetros necesarios.
Postcondición	Se muestran la serie de Fourier y los coeficientes en notación matemática.

Tabla 4.7: Requerimiento funcional No. 6

Requerimiento	Manejo de Errores
Identificador	RF7
Prioridad de desarrollo	Media
Entrada	Función ingresada por el usuario, parámetros seleccionados
Salida	Mensajes de error claros en caso de problemas con la función o los parámetros
Descripción	La aplicación debe detectar errores en la entrada de funciones o parámetros no válidos y mostrar mensajes de error claros. También debe notificar si las integrales no convergen o si Maxima no puede procesar la función.
Precondición	El usuario ha ingresado una función o parámetros y ha intentado procesar los datos.
Postcondición	Se muestra un mensaje de error claro si ocurre algún problema.

Tabla 4.8: Requerimiento funcional No. 7

Requerimiento	Optimización de Resultados
Identificador	RF8

Prioridad de desarrollo	Baja
Entrada	Parámetros y resultados de coeficientes previos
Salida	Optimización de cálculo y presentación de resultados mejorada
Descripción	La aplicación debe optimizar el cálculo de los coeficientes para mejorar el rendimiento y la experiencia del usuario, manejando grandes cantidades de términos de manera eficiente.
Precondición	El usuario ha ingresado parámetros complejos o grandes cantidades de términos.
Postcondición	Los resultados optimizados se calculan y presentan rápidamente.

Tabla 4.9: Requerimiento funcional No. 8

4.2.2. Requerimientos No Funcionales

Los requerimientos no funcionales describen una restricción sobre el sistema que limita nuestras elecciones en la construcción de una solución al problema. Describen atributos sólo del sistema o del ambiente del sistema que no están relacionados directamente con los requisitos funcionales. Los requisitos no funcionales incluyen restricciones cuantitativas, tipo de plataforma.

Nombre	Descripción
Rendimiento y Escalabilidad	El sistema debe calcular los coeficientes de Fourier en un tiempo razonable, incluso para funciones con múltiples trozos o con un alto número de términos en la serie. La visualización gráfica debe ser fluida y responsiva, incluso al agregar o quitar muchos términos de la serie. Las interacciones en el lienzo canvas deben realizarse sin latencia perceptible.
Interfaz de Usuario Amigable e Intuitiva	La interfaz debe ser intuitiva y fácil de usar, con botones y controles claramente identificados. Los mensajes de error y advertencia deben ser comprensibles, evitando el uso de jerga técnica complicada. La notación matemática debe ser clara y precisa, usando MathQuill para entradas y visualización de resultados.
Manejo de la Concurrencia	La aplicación debe soportar múltiples usuarios concurrentes enviando cálculos al backend sin generar bloqueos ni conflictos en las llamadas a Maxima.

Portabilidad y Compatibilidad	La aplicación debe ser compatible con los principales navegadores modernos (Chrome, Firefox, Edge). Aunque no es recomendable usar la aplicación en dispositivos móviles, el diseño debe ser responsivo para ajustarse a diferentes tamaños de pantalla, incluidos dispositivos móviles y tabletas.
Modularidad y Extensibilidad del Código	El frontend debe estar diseñado para facilitar la migración a frameworks como Angular en el futuro. El código debe ser modular y seguir buenas prácticas de desarrollo para facilitar la adición de nuevas características. Aunque el sistema no requiere una base de datos en su versión actual, el backend debe ser fácilmente extensible para permitir el almacenamiento de sesiones o configuraciones del usuario en el futuro.
Seguridad de la Aplicación	Las peticiones a la API deben estar protegidas contra inyecciones de código. Implementar seguridad básica en el backend para evitar solicitudes maliciosas o repetitivas que puedan afectar el rendimiento del servicio.
Mantenibilidad de Código	La aplicación debe tener una guía de usuario clara para explicar cómo ingresar funciones, seleccionar series y usar la interfaz. El código debe estar bien documentado para que futuros desarrolladores puedan entender la lógica y estructura de la aplicación fácilmente.

Tabla 4.10: Tabla de requerimientos no funcionales

4.3. Análisis de Riesgos

4.3.1. Identificación de Riesgos

4.3.2. Jerarquización de Riesgos

4.4. Estimación de Costos

4.4.1. Método de Estimación de Costos

4.4.2. Detalle de los Costos Estimados

4.5. Diseño del Sistema

4.5.1. Arquitectura General del Sistema

4.5.1.1. Arquitectura Cliente-Servidor

4.5.1.2. Flujo de Datos

4.5.2. Diseño de la Interfaz de Usuario

4.5.2.1. Estructura de la Interfaz

4.5.2.2. Prototipo de la Interfaz

CAPÍTULO 5

Implementación (Primer Prototipo)

En este capítulo vamos a juntar todo lo primero

APÉNDICE A

Cálculo de los Coeficientes de la Serie de Fourier

Dada la función $f(x) = x$, vamos a calcular los coeficientes correspondientes a su serie de Fourier en el intervalo $[-\pi, \pi]$.

A.1. Forma Trigonométrica

La serie trigonométrica de Fourier para una función $f(x)$ está dada por la siguiente expresión:

$$f(x) = \frac{a_0}{2} + \sum_{n=1}^{\infty} (a_n \cos(n\omega_0 x) + b_n \sin(n\omega_0 x))$$

Donde los coeficientes a_0 , a_n y b_n se definen como:

$$a_0 = \frac{2}{T} \int_{-T/2}^{T/2} f(x) dx$$

$$a_n = \frac{2}{T} \int_{-T/2}^{T/2} f(x) \cos(n\omega_0 x) dx$$

$$b_n = \frac{2}{T} \int_{-T/2}^{T/2} f(x) \sin(n\omega_0 x) dx$$

Donde $\omega_0 = \frac{2\pi}{T}$, y en este caso, $T = 2\pi$, por lo que $\omega_0 = 1$.

Calculamos cada uno de los coeficientes para $f(x) = x$:

- El coeficiente a_0 es:

$$a_0 = \frac{2}{2\pi} \int_{-\pi}^{\pi} x dx = \frac{1}{\pi} \left[\frac{x^2}{2} \right]_{-\pi}^{\pi} = \frac{1}{\pi} \left(\frac{\pi^2}{2} - \frac{(-\pi)^2}{2} \right) = 0$$

Por lo tanto, $a_0 = 0$.

- El coeficiente a_n es:

Dado que $x \cos(nx)$ es una función impar, ya que x es impar y $\cos(nx)$ es par, su integral en un intervalo simétrico es cero:

$$a_n = \frac{1}{\pi} \int_{-\pi}^{\pi} x \cos(nx) dx = 0$$

- El coeficiente b_n es:

Dado que $x \sin(nx)$ es una función par (producto de dos funciones impares), podemos calcular:

$$b_n = \frac{1}{\pi} \int_{-\pi}^{\pi} x \sin(nx) dx = \frac{2}{\pi} \int_0^{\pi} x \sin(nx) dx$$

Aplicamos integración por partes, tomando $u = x$ y $dv = \sin(nx) dx$, lo que nos da $du = dx$ y $v = -\frac{1}{n} \cos(nx)$. Entonces:

$$\int_0^{\pi} x \sin(nx) dx = -\frac{x}{n} \cos(nx) \Big|_0^{\pi} + \frac{1}{n} \int_0^{\pi} \cos(nx) dx$$

Evaluamos el primer término:

$$-\frac{\pi}{n} \cos(n\pi) + 0 = -\frac{\pi}{n}(-1)^n$$

La integral restante es:

$$\frac{1}{n} \int_0^{\pi} \cos(nx) dx = \frac{1}{n} \left[\frac{\sin(nx)}{n} \right]_0^{\pi} = \frac{1}{n^2} (0 - 0) = 0$$

Por lo tanto, el coeficiente b_n es:

$$b_n = \frac{2}{\pi} \left(-\frac{\pi}{n}(-1)^n \right) = \frac{2(-1)^{n+1}}{n}$$

Entonces, la serie trigonométrica de Fourier para $f(x) = x$ es:

$$f(x) = 2 \sum_{n=1}^{\infty} \frac{(-1)^{n+1}}{n} \sin(nx)$$

A.2. Forma Exponencial Compleja

La serie exponencial compleja de Fourier está dada por la siguiente expresión:

$$f(x) = \sum_{n=-\infty}^{\infty} c_n e^{in\omega_0 x}$$

Donde los coeficientes c_n se definen como:

$$c_n = \frac{1}{T} \int_{-T/2}^{T/2} f(x) e^{-in\omega_0 x} dx$$

Calculamos el coeficiente c_n para $n \neq 0$:

$$c_n = \frac{1}{2\pi} \int_{-\pi}^{\pi} x e^{-inx} dx$$

Aplicamos integración por partes con $u = x$ y $dv = e^{-inx} dx$, obteniendo $du = dx$ y $v = \frac{e^{-inx}}{-in}$. Entonces:

$$c_n = \frac{1}{2\pi} \left(\left[x \frac{e^{-inx}}{-in} \right]_{-\pi}^{\pi} - \int_{-\pi}^{\pi} \frac{e^{-inx}}{-in} dx \right)$$

El primer término es:

$$\left[x \frac{e^{-inx}}{-in} \right]_{-\pi}^{\pi} = \frac{\pi e^{-in\pi} - (-\pi) e^{in\pi}}{-in} = \frac{2\pi(-1)^n}{-in}$$

El segundo término es cero, ya que:

$$\int_{-\pi}^{\pi} e^{-inx} dx = 0$$

Por lo tanto:

$$c_n = \frac{1}{2\pi} \left(\frac{2\pi(-1)^n}{-in} \right) = \frac{i(-1)^n}{n}$$

Para $n = 0$:

$$c_0 = \frac{1}{2\pi} \int_{-\pi}^{\pi} x dx = 0$$

Así que la serie exponencial compleja de Fourier para $f(x) = x$ es:

$$f(x) = \sum_{\substack{n=-\infty \\ n \neq 0}}^{\infty} \frac{i(-1)^n}{n} e^{inx}$$

Ambas representaciones son equivalentes y proporcionan la expansión de Fourier correcta para la función dada.

Bibliografía

- [1] J.M. Almira. *Fourier: un matemático al servicio de la física*. Genios de las matemáticas. RBA, 2017. ISBN: 9788447387755.
- [2] Wolfram — Alpha: Computational Intelligence. *Fourier Analysis—Wolfram Language Documentation*. <https://reference.wolfram.com/language/guide/FourierAnalysis.html>. Accedido: 17 Junio, 2024. 2024.
- [3] GeoGebra. *GeoGebra - the world's favorite, free math tools used by over 100 million students and teachers*. <https://www.geogebra.org>. Accedido: 14 Junio, 2024. 2024.
- [4] Desmos. *Desmos — Let's learn together*. <https://www.desmos.com>. Accedido: 14 Junio, 2024. 2024.
- [5] Manim Community. *Manim Community*. <https://www.manim.community>. Accedido: 14 Junio, 2024. 2024.
- [6] Ettore Messina. *Fourier Series in Python*. <https://computationalmindset.com/en/mathematics/fourier-series-in-python.html>. Accedido: 3 Octubre, 2024. 2024.
- [7] MathWorks. *MathWorks - Creadores de MATLAB y Simulink - MATLAB y Simulink*. <https://la.mathworks.com/>. Accedido: 14 Junio, 2024. 2024.
- [8] Cañada A. *Una perspectiva histórica de las series de Fourier: de las ecuaciones de ondas y del calor a los operadores compactos y autoadjuntos*. <https://www.nieuwarchief.nl/serie5/pdf/naw5-2000-01-3-242.pdf>. Accedido: 23 Octubre, 2024. 2000.
- [9] Campo Bedoya Oscar Leandro. *Ecuación de D'Alembert, de la cuerda vibrante, bajo la teoría de Lie*. <https://repositorio.unal.edu.co/handle/unal/52162>. Accedido: 23 Octubre, 2024. 2014.
- [10] Chamorro D. *Lección n°5: Ecuación de Ondas*. https://www.amarun.org/images/amarun/materiales/EDP/edp/Leccion_5.pdf. Accedido: 23 Octubre, 2024. 2015.
- [11] I. Grattan-Guinness. *Daniel Bernoulli and the varieties of mechanics*. <https://www.nieuwarchief.nl/serie5/pdf/naw5-2000-01-3-242.pdf>. Accedido: 23 Octubre, 2024. 2000.
- [12] Garber E. *The Language of Physics: The Calculus and the Development of Theoretical Physics in Europe, 1750-1914*. Springer, 1999. ISBN: 978-1-4612-7272-4.

- [13] Brereton J. T. *FOURIER SERIES: SOLVING THE HEAT EQUATION*. <https://math.berkeley.edu/~jbrere/heatequation>. Accedido: 23 Octubre, 2024. 2015.
- [14] Feldman J. *Solution of the Heat Equation by Separation of Variables*. <https://personal.math.ubc.ca/~feldman/m267/heatSln.pdf>. Accedido: 23 Octubre, 2024. 2007.
- [15] P.G. Tolstov. *Fourier Series*. Prentice-Hall, Inc., 1962. ISBN: 9780486633176.
- [16] P.H. Hsu. *Analisis de Fourier*. Addison-Wesley Iberoamericana, 1987. ISBN: 9685000476.
- [17] Dr. Wilfrido Gómez Flores. *Introducción al Análisis de Fourier*. https://www.tamps.cinvestav.mx/~wgomez/documentos/analisis_de_fourier.pdf. Accedido: 26 Oct, 2024. 2024.
- [18] W. Moebis. *15.1 Movimiento armónico simple - Física universitaria volumen 1 — OpenStax*. <https://openstax.org/books/fisica-universitaria-volumen-1/pages/15-1-movimiento-armonico-simple>. Accedido: 26 Oct, 2024. 2024.
- [19] D. G. Zill. *Matemáticas Avanzadas para la Ingeniería*. McGraw-Hill, 2011. ISBN: 9786071507723.
- [20] MathWorks. *Metodologías de desarrollo*. <https://woodyweb.wordpress.com/2015/08/24/metodologias-de-desarrollo/>. Accedido: 22 Oct, 2024. 2015.