

Ecuaciones Diferenciales

Transformada de Laplace I

Clase 12



INTRODUCCIÓN DE LA CLASE

La resolución de ecuaciones diferenciales ordinarias (EDO) mediante series de potencias representa una técnica fundamental en el análisis de sistemas matemáticos, especialmente cuando las soluciones no pueden obtenerse por métodos elementales o cuando se desea una descripción detallada del comportamiento local de la solución. Este enfoque consiste en representar la solución como una suma infinita de términos polinomiales, cada uno multiplicado por una potencia de la variable independiente, centrada alrededor de un punto determinado. La gran ventaja de este método es su aplicabilidad a ecuaciones con coeficientes variables y su capacidad para ofrecer soluciones en forma de series convergentes incluso en presencia de singularidades suaves. El método de los coeficientes indeterminados permite calcular de forma sistemática los términos de la serie al sustituir la expansión en la ecuación original y comparar coeficientes, generando así una secuencia de relaciones algebraicas que determinan los valores sucesivos de los coeficientes. Esta técnica no solo es útil para resolver EDO, sino que también proporciona una visión más profunda sobre el carácter de las soluciones, su dominio de validez y la influencia de las condiciones iniciales.

En este contexto, las series de Taylor constituyen una herramienta clave, ya que permiten aproximar la solución de una EDO alrededor de un punto con una precisión controlable, según el número de derivadas consideradas. Este enfoque es ampliamente utilizado en aplicaciones científicas y de ingeniería, especialmente cuando se requiere una solución local rápida y precisa. El desarrollo de la solución en serie, además de ser una técnica teórica poderosa, se presta perfectamente a la automatización mediante herramientas computacionales. En particular, el lenguaje Python, junto con bibliotecas como SymPy, permite realizar derivaciones simbólicas, construcciones de series y evaluaciones numéricas de manera eficiente y precisa. Esta capacidad de automatización no solo alivia la carga algorítmica del cálculo manual, sino que abre la puerta a la exploración dinámica de soluciones, comparación entre métodos y validación de resultados. Así, la integración del enfoque analítico con herramientas computacionales fortalece significativamente la capacidad de análisis, experimentación y resolución de problemas complejos en disciplinas que dependen del modelado mediante EDO.

Resultado o resultados de aprendizaje que será abordado con el contenido de la clase.

- 1) Comprender los conceptos fundamentales de las ecuaciones diferenciales y su aplicación en la resolución de problemas de ingeniería y ciencias computacionales.
- 2) Desarrollar habilidades para la resolución de ecuaciones diferenciales de primer y segundo orden con distintas técnicas.
- 3) Aplicar los conocimientos de ecuaciones diferenciales para la modelización de fenómenos físicos y su simulación mediante técnicas numéricas en herramientas computacionales.

Clase 12: Transformada de Laplace I

La transformada de Laplace es una de las herramientas más poderosas en el análisis y resolución de ecuaciones diferenciales lineales, especialmente en contextos donde se modelan sistemas dinámicos sujetos a condiciones iniciales. Desde una perspectiva computacional e ingenieril, esta transformada permite convertir problemas formulados en el dominio del tiempo —con derivadas y funciones complicadas— en problemas algebraicos en el dominio complejo, donde las operaciones de integración y diferenciación se simplifican sustancialmente. Esto resulta de enorme valor en el diseño de sistemas de control, procesamiento de señales, análisis de circuitos eléctricos, simulación de sistemas mecánicos y entrenamiento de modelos diferenciales en inteligencia artificial.

La clase inicia con la definición formal de la transformada de Laplace, que asocia a una función $f(t)$, definida para $t \geq 0$, una función $F(s)$ en el dominio complejo mediante la integral:

$$\mathcal{L}\{f(t)\} = F(s) = \int_0^{\infty} e^{-st} f(t) dt$$

Esta operación convierte ecuaciones diferenciales ordinarias en ecuaciones algebraicas en s , facilitando su solución y análisis computacional. Posteriormente, se estudian las propiedades fundamentales de la transformada —como linealidad, escalamiento, derivación y convolución—, que son esenciales para manipular y descomponer funciones complejas en sus componentes transformadas. Estas propiedades son

implementadas en software como MATLAB, Mathematica, Python (con SymPy o SciPy), y permiten resolver ecuaciones simbólicas o numéricas de manera eficiente.

A continuación, se aborda la transformada inversa de Laplace, la cual permite recuperar la solución original en el dominio del tiempo a partir de su representación en el dominio s . Este paso es crucial en la interpretación de resultados en contextos físicos y computacionales, como la respuesta temporal de un sistema ante una entrada dada. Finalmente, se introducen los teoremas de traslación (en tiempo y en frecuencia), que son herramientas clave para tratar desplazamientos temporales y cambios de frecuencia, fundamentales en aplicaciones de análisis de señales, diseño de filtros digitales y modelado de sistemas con retardos o memorias.

En conjunto, esta clase proporciona las bases teóricas necesarias para aplicar la transformada de Laplace como un puente entre la teoría matemática de ecuaciones diferenciales y su implementación computacional en aplicaciones reales. Su dominio es indispensable para ingenieros, científicos computacionales y desarrolladores de sistemas inteligentes que requieren analizar, controlar o simular sistemas dinámicos de manera precisa y eficiente.

12.1. Definición de la transformada de Laplace

La **transformada de Laplace** es una herramienta matemática fundamental utilizada para transformar ecuaciones diferenciales en ecuaciones algebraicas, facilitando su resolución en dominios complejos como la ingeniería, la física y las ciencias computacionales. Formalmente, la transformada de Laplace de una función $(f(t))$, definida para $(t \geq 0)$, se expresa como:

$$\left[\mathcal{L}f(t) = F(s) = \int_0^{\infty} e^{-st} f(t) dt, \right]$$

donde (s) es una variable compleja $((s = \sigma + i\omega))$ y $(F(s))$ es la transformada en el dominio de la frecuencia. Para que la integral converja, $(f(t))$ debe ser de crecimiento exponencial limitado, es decir, $(|f(t)| \leq M e^{at})$ para alguna constante (M) y (a) , con $(\text{Re}(s) > a)$.

En el ámbito computacional, la transformada de Laplace simplifica el análisis de sistemas dinámicos, como circuitos eléctricos, sistemas mecánicos y algoritmos de

control. Por ejemplo, en el diseño de filtros digitales, la transformada convierte ecuaciones diferenciales en funciones de transferencia, que se implementan en software como MATLAB o Python (usando bibliotecas como SciPy). Esto permite analizar la respuesta de sistemas a entradas específicas sin resolver directamente las ecuaciones en el dominio del tiempo.

Un ejemplo clásico es la transformada de una función exponencial, ($f(t) = e^{at}$), cuya transformada es:

$$\left[\mathcal{L}e^{at} = \frac{1}{s-a}, \quad \text{Re}(s) > a. \right]$$

En aplicaciones prácticas, esta propiedad se usa en simulaciones de decaimiento exponencial, como en la gestión de cachés en sistemas operativos o en modelos de aprendizaje automático para tasas de aprendizaje adaptativas. La transformada de Laplace también es clave en el procesamiento de señales, donde convierte señales temporales en el dominio de la frecuencia, facilitando el análisis de componentes espectrales.

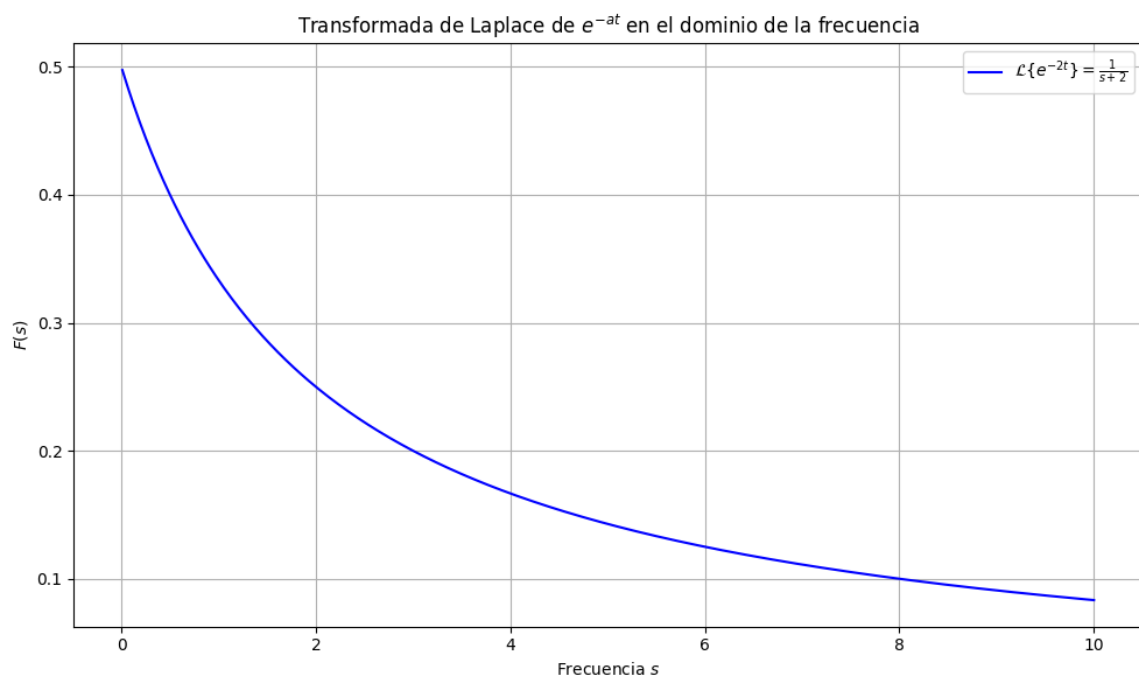


Gráfico 1: Representación de la transformada de Laplace de una función exponencial en el dominio de la frecuencia. Creación propia.

En computación, la transformada se implementa numéricamente para funciones no analíticas usando métodos como la transformada rápida de Laplace, optimizados en bibliotecas como NumPy. En control automático, permite diseñar controladores PID analizando polos y ceros de $(F(s))$, lo que es crucial en robótica y automatización.

En resumen, la transformada de Laplace es una herramienta esencial para transformar problemas dinámicos en algebraicos, con aplicaciones en simulación computacional, análisis de sistemas y procesamiento de datos.

Enlace 1:

<https://www.youtube.com/watch?v=YkGRw3BpmDw&pp=ygVDc2VyaWVzIGRlIFRheWxvciBzZSB1dGlsaXphbiBwYXJhIHJlc29sdmVyIGVjdWFjaW9uZXMGZGlmZXJlbnNpYWxlc9IHCQnHCQGHKiGM7w%3D%3D>

Este video ofrece una explicación detallada sobre cómo las series de Taylor se utilizan para resolver ecuaciones diferenciales, con ejemplos prácticos.

Descripción del enlace: Este recurso visual es ideal para comprender la aplicación de las series de Taylor en problemas matemáticos y computacionales.

12.1.1. Propiedades fundamentales

Las propiedades fundamentales de la *transformada de Laplace* permiten manipular ecuaciones diferenciales de manera eficiente, convirtiendo operaciones en el dominio del tiempo en operaciones algebraicas en el dominio de la frecuencia. A continuación, se describen las propiedades más relevantes y sus aplicaciones computacionales:

1. **Linealidad:** $[\mathcal{L}af(t) + bg(t) = a\mathcal{L}f(t) + b\mathcal{L}g(t)]$. Esta propiedad permite descomponer problemas complejos en sumas de términos más simples. En procesamiento de señales, se usa para analizar superposiciones de señales, implementado en software como MATLAB para diseñar filtros.
2. **Derivada:** $[\mathcal{L}f'(t) = sF(s) - f(0^-), [\mathcal{L}f''(t) = s^2F(s) - sf(0^-) - f'(0^-).]$ Esta propiedad transforma ecuaciones diferenciales en polinomios en (s) . En circuitos RLC, permite resolver para la corriente o voltaje sin integrar directamente, usando solvers en Python (SciPy).

3. **Integral:** $\left[\mathcal{L}\left\{\int_0^t f(\tau) d\tau\right\} = \frac{F(s)}{s}\right]$ Útil en sistemas con memoria, como en modelado de acumuladores en simulaciones de energía, implementado en herramientas como Simulink.

4. **Convolución:** $[\mathcal{L}\{f(t) * g(t)\} = F(s)G(s),]$ donde $(f(t) * g(t) = \int_0^t f(\tau)g(t - \tau) d\tau)$. Esta propiedad es crucial en procesamiento de señales, donde la convolución modela la respuesta de sistemas lineales. En computación, se implementa en algoritmos de filtrado digital con bibliotecas como SciPy.signal.

En aplicaciones computacionales, estas propiedades facilitan el análisis de sistemas dinámicos. Por ejemplo, en control automático, la propiedad de la derivada convierte ecuaciones de movimiento en funciones de transferencia, analizadas mediante diagramas de Bode en MATLAB. En machine learning, la convolución en el dominio de Laplace optimiza redes convolucionales continuas, integradas en frameworks como TensorFlow.

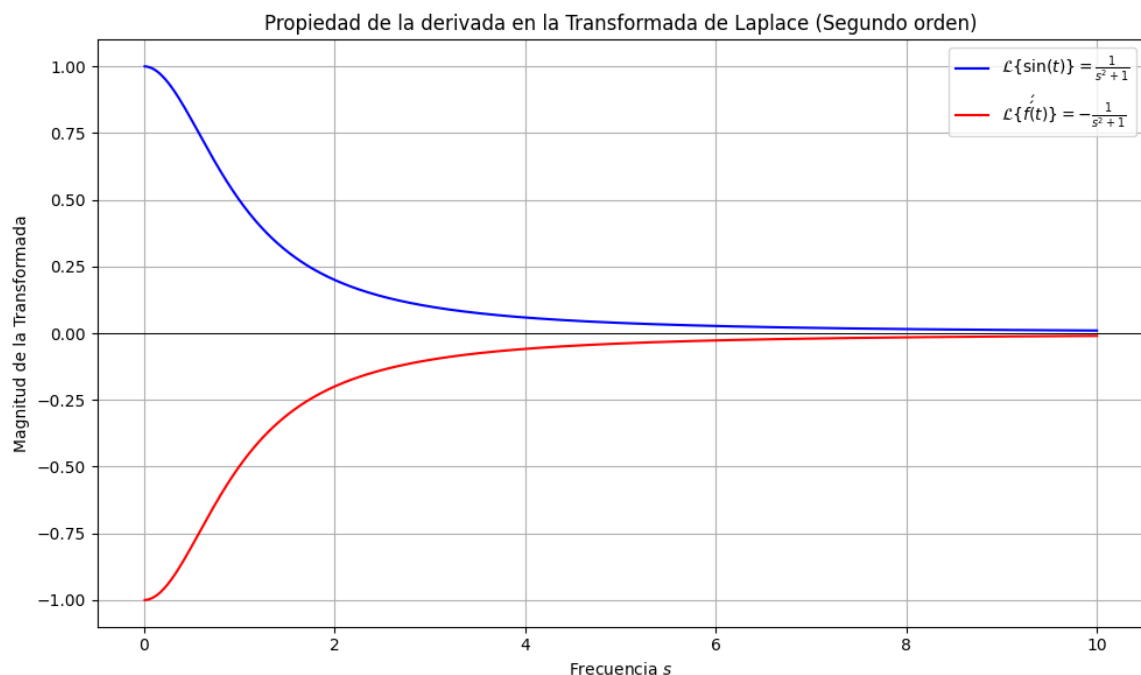


Gráfico 2: Ilustración de la propiedad de la derivada en la transformada de Laplace para una ecuación de segundo orden. Creación propia.

En simulaciones numéricas, las propiedades se aplican para resolver sistemas con condiciones iniciales no nulas, como en dinámica de fluidos computacional (CFD), donde solvers ODE aprovechan la linealidad para descomponer flujos complejos.

En conclusión, las propiedades fundamentales de la transformada de Laplace —como la linealidad, derivación en el dominio del tiempo, traslación, y convolución— son herramientas matemáticas poderosas que permiten transformar ecuaciones diferenciales complejas en expresiones algebraicas manejables. Esta capacidad de simplificación facilita el análisis y la resolución de sistemas dinámicos que, de otro modo, serían difíciles de abordar en el dominio temporal.

En el ámbito de la ingeniería y la computación, estas propiedades son clave para diseñar y optimizar sistemas de control, modelar circuitos eléctricos, analizar señales y simular respuestas en tiempo real. Gracias a su integración con herramientas computacionales como MATLAB, Python (SciPy), Simulink y entornos de hardware embebido, la transformada de Laplace permite una implementación eficiente en entornos reales.

Asimismo, en el contexto del aprendizaje automático y los modelos diferenciales modernos, las transformadas se emplean para estudiar el comportamiento de sistemas entrenables en el dominio de la frecuencia, permitiendo entender y controlar dinámicas continuas como las que aparecen en redes neuronales diferenciales (Neural ODEs). Por tanto, el dominio de estas propiedades no solo fortalece el conocimiento teórico, sino que también habilita aplicaciones prácticas de alto impacto en la ciencia y la tecnología actuales.

12.1.2. Transformada inversa

La **transformada inversa de Laplace**, denotada $(\mathcal{L}^{-1}F(s) = f(t))$, recupera la función original $(f(t))$ a partir de su transformada $(F(s))$. Este proceso es crucial para interpretar soluciones en el dominio del tiempo, especialmente en simulaciones de sistemas físicos y computacionales.

El cálculo de la transformada inversa se realiza comúnmente mediante:

1. **Tablas de transformadas:** Se usan tablas predefinidas para identificar $(f(t))$ correspondiente a $(F(s))$. Por ejemplo: $\left[\mathcal{L}^{-1}\left\{\frac{1}{s-a}\right\} = e^{at}, \quad \mathcal{L}^{-1}\left\{\frac{\omega}{s^2+\omega^2}\right\} = \right.$

$\sin(\omega t).$] En Python, SymPy automatiza este proceso buscando correspondencias en tablas.

2. **Descomposición en fracciones parciales:** Para $(F(s))$ racional, se descompone en sumas de términos simples. Por ejemplo, para $(F(s) = \frac{2s+3}{(s+1)(s+2)})$, se escribe: $[F(s) = \frac{A}{s+1} + \frac{B}{s+2},]$ resolviendo para (A) y (B) , y luego aplicando la transformada inversa a cada término. En computación, NumPy resuelve sistemas lineales para obtener los coeficientes.
3. **Métodos numéricos:** Cuando $(F(s))$ es complejo, se usan algoritmos como la transformada inversa de Laplace numérica (NILT), implementados en bibliotecas como mpmath en Python.

En aplicaciones, la transformada inversa es clave en el diseño de sistemas de control, donde $(F(s))$ representa una función de transferencia y $(f(t))$ la respuesta temporal. Por ejemplo, en un circuito RLC, la transformada inversa convierte $(I(s))$ en la corriente $(i(t))$, simulada en SPICE para analizar transitorios.

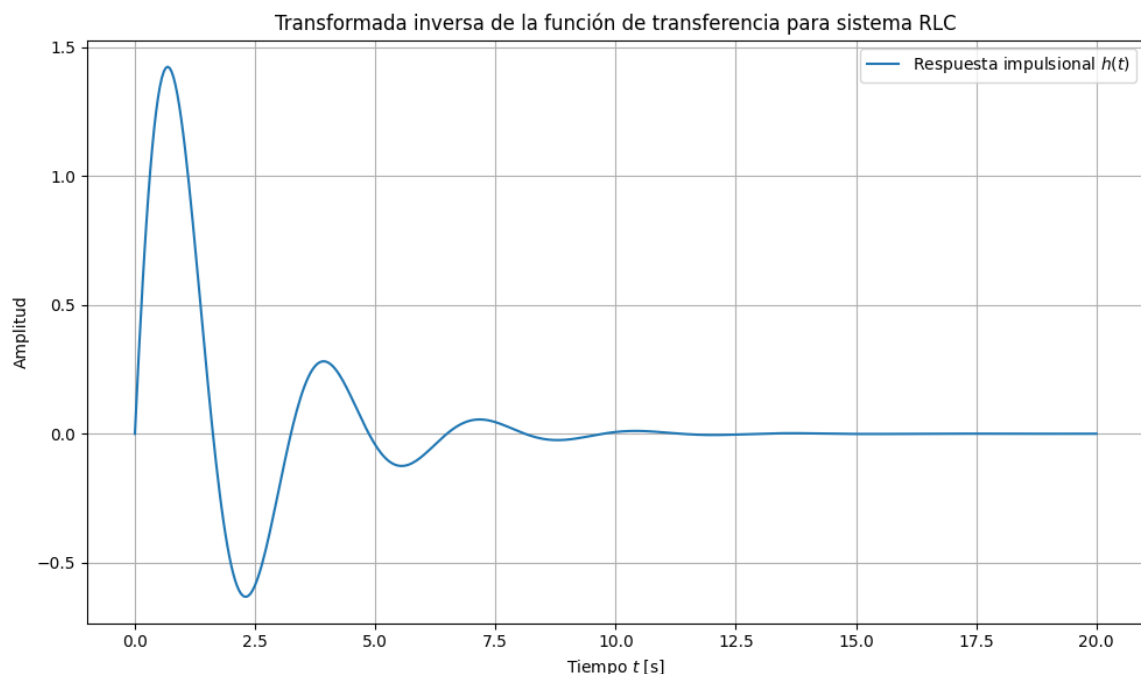


Gráfico 3: Transformada inversa de una función de transferencia para un sistema RLC. Creación propia.

En procesamiento de señales, la transformada inversa recupera señales temporales de espectros de frecuencia, crucial en audio y video. En machine learning, se usa en modelos dinámicos continuos, como redes neuronales diferenciales, para interpretar trayectorias temporales.

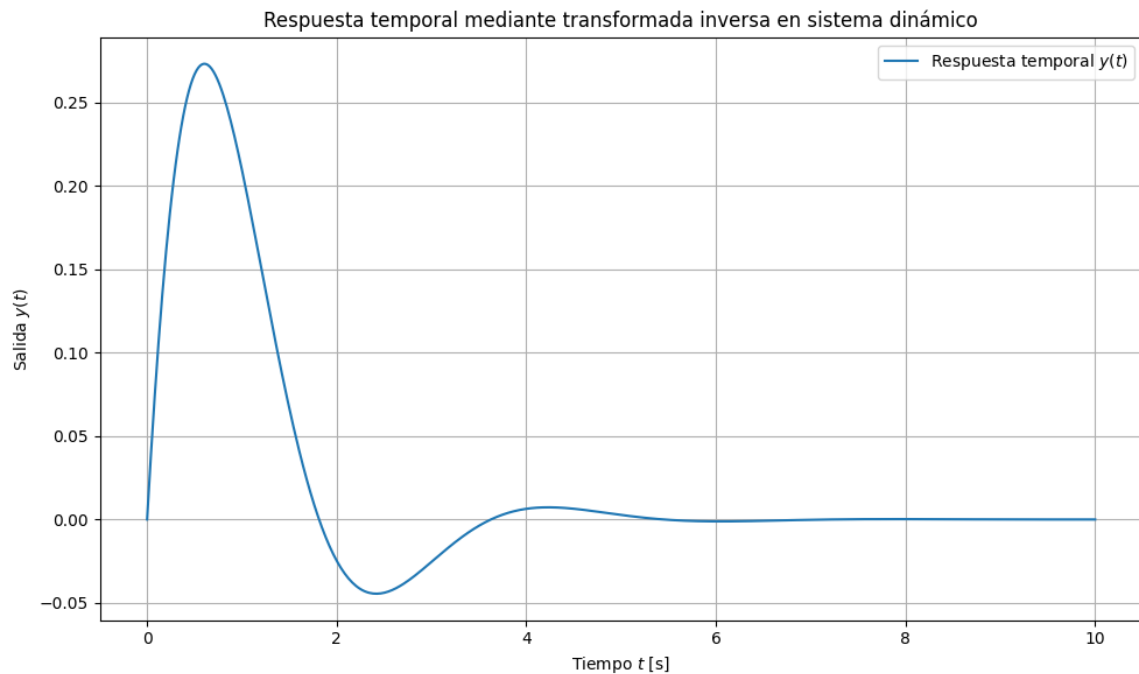


Gráfico 4: Respuesta temporal obtenida mediante transformada inversa en un sistema dinámico. Creación propia.

La transformada inversa también es vital en simulaciones de dinámica de fluidos, donde se recuperan perfiles temporales de velocidad o presión a partir de soluciones en el dominio de Laplace.

En resumen, la transformada inversa de Laplace es esencial para interpretar soluciones en el dominio del tiempo, con aplicaciones en simulación, control y análisis de datos.

12.1.3. Teoremas de traslación

Los *teoremas de traslación* de la transformada de Laplace son herramientas clave para manipular funciones desplazadas en el dominio del tiempo o de la frecuencia, simplificando el análisis de sistemas dinámicos con retardos o modulaciones. Los dos teoremas principales son:

1. **Teorema de traslación en el tiempo** (o primer teorema de traslación):
 $[\mathcal{L}f(t-a)u(t-a) = e^{-as}F(s),]$ donde $(u(t-a))$ es la función escalón unitario y $(F(s) = \mathcal{L}f(t))$. Este teorema modela retardos temporales, comunes en sistemas de control con latencia, como en redes de comunicación. En Python, SciPy.signal simula sistemas con retardos usando esta propiedad.
2. **Teorema de traslación en la frecuencia** (o segundo teorema de traslación):
 $[\mathcal{L}e^{at}f(t) = F(s-a).]$ Este teorema es útil para funciones moduladas exponencialmente, como en señales amortiguadas en circuitos eléctricos. En simulaciones, permite analizar sistemas con crecimiento o decaimiento exponencial.

En aplicaciones computacionales, el teorema de traslación en el tiempo es crucial en el análisis de sistemas con retardos, como en protocolos de red donde los paquetes tienen latencia. En robótica, modela el retardo en sensores, simulado en ROS para optimizar trayectorias.

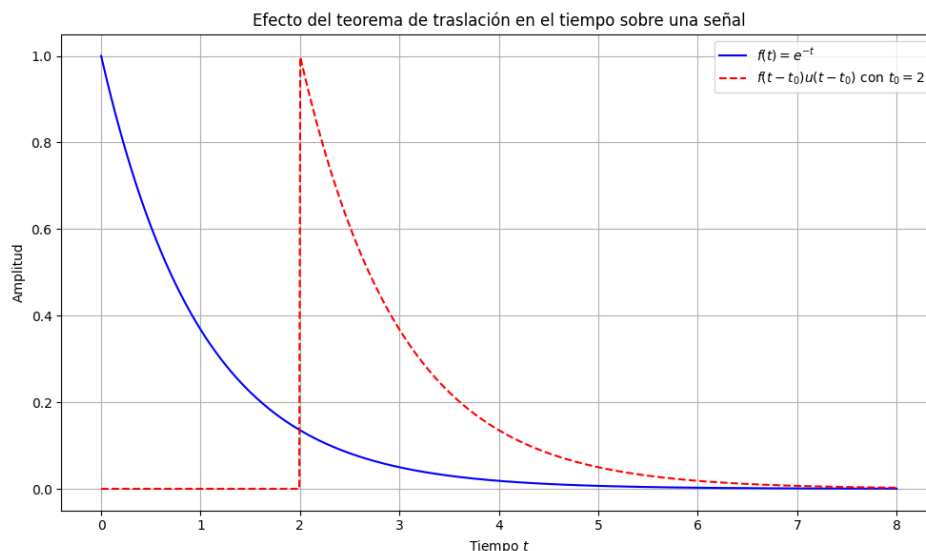


Gráfico 5: Efecto del teorema de traslación en el tiempo sobre una señal desplazada.

Creación propia.

El teorema de traslación en la frecuencia dentro del marco de la transformada de Laplace establece que multiplicar una función temporal $f(t)$ por una exponencial e^{at} equivale a desplazar su transformada $F(s)$ a $F(s-a)$. Esta propiedad es esencial en el procesamiento de señales, particularmente en la modulación de señales en

radiofrecuencia, donde las señales de entrada se desplazan en frecuencia para ser transmitidas o aisladas en bandas específicas del espectro. Gracias a este teorema, es posible modelar con precisión cómo una señal se comporta tras ser modulada, permitiendo su análisis, filtrado y reconstrucción con técnicas eficientes tanto analíticas como computacionales.

En el ámbito del machine learning, especialmente en el subcampo del aprendizaje continuo o basado en ecuaciones diferenciales, este teorema encuentra aplicaciones relevantes al modelar dinámicas con tasas de cambio exponenciales, como en redes neuronales diferenciales (Neural ODEs). Al incorporar funciones de tipo $e^{at} \cdot f(t)$ en la formulación de un sistema, el teorema de traslación en la frecuencia permite adaptar los solvers para considerar variaciones rápidas, amortiguadas o explosivas en el tiempo. Frameworks como TensorFlow y TorchDyn integran estos modelos en sus solvers ODE, facilitando el entrenamiento de redes que aprenden dinámicas físicas, financieras o biológicas directamente a partir de datos temporales, con ajustes en frecuencia implícitos en sus funciones de pérdida o arquitectura.

Así, este teorema actúa como un puente entre la teoría matemática clásica y las tecnologías modernas, permitiendo que conceptos fundamentales como la traslación en frecuencia se apliquen a desafíos reales en comunicaciones, inteligencia artificial y sistemas complejos modelados computacionalmente.

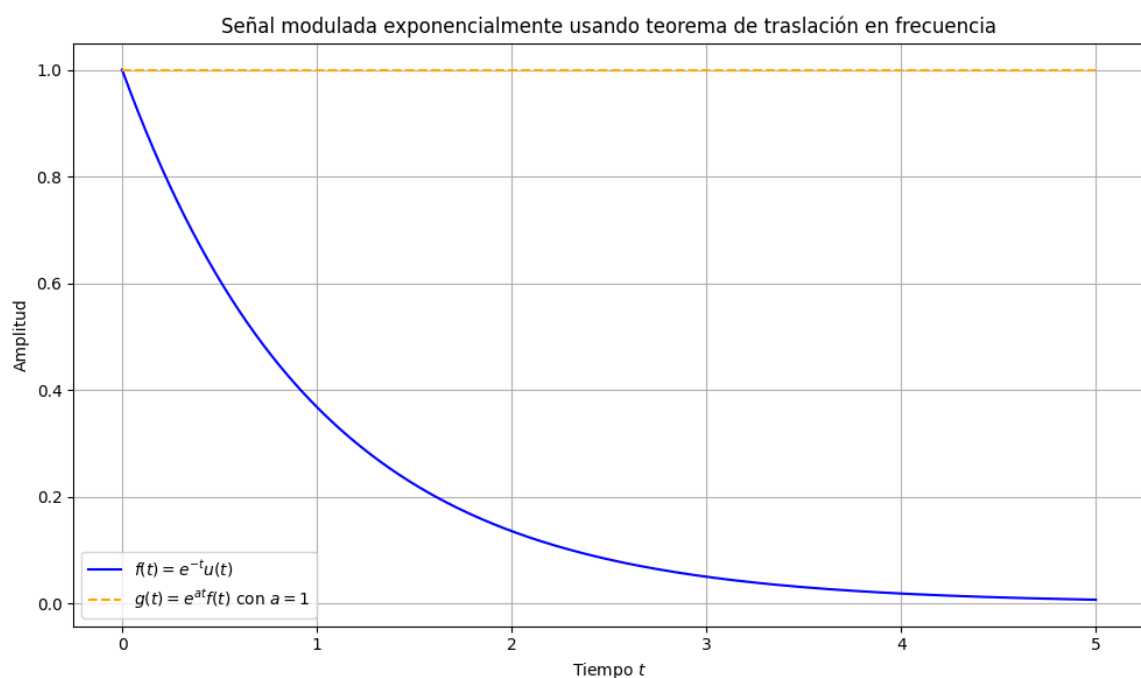


Gráfico 6: Representación de una señal modulada exponencialmente usando el teorema de traslación en la frecuencia. Creación propia.

En dinámica de fluidos computacional (CFD), los teoremas de traslación de la transformada de Laplace son especialmente útiles para modelar y analizar fenómenos donde existen retardos temporales, amortiguamiento o desplazamientos en el comportamiento ondulatorio de los fluidos. Estos teoremas permiten representar matemáticamente el efecto de ondas que se propagan con cierta demora, como sucede en tuberías, cámaras de combustión o canales con condiciones no estacionarias. La traslación en el tiempo, en particular, facilita la formulación de condiciones iniciales desplazadas, mientras que la traslación en frecuencia se asocia a efectos de amortiguamiento o filtros dinámicos en la propagación de energía.

Estos modelos transformados pueden ser luego resueltos numéricamente en entornos como COMSOL Multiphysics, donde las herramientas de análisis en el dominio de Laplace se utilizan para simular transitorios complejos y analizar el comportamiento de fluidos en estructuras con retardo, válvulas dinámicas o controladores adaptativos. El uso de estos teoremas permite reducir la complejidad del sistema original, transformándolo en un problema algebraico o en un sistema con condiciones simplificadas para su solución numérica.

En este contexto, la combinación de teoría matemática con capacidades de simulación computacional avanzada demuestra cómo los teoremas de traslación no solo facilitan el análisis formal de ecuaciones diferenciales, sino que impulsan soluciones prácticas y eficientes en el diseño y validación de sistemas de fluidos en ingeniería mecánica, aeroespacial, civil e industrial.

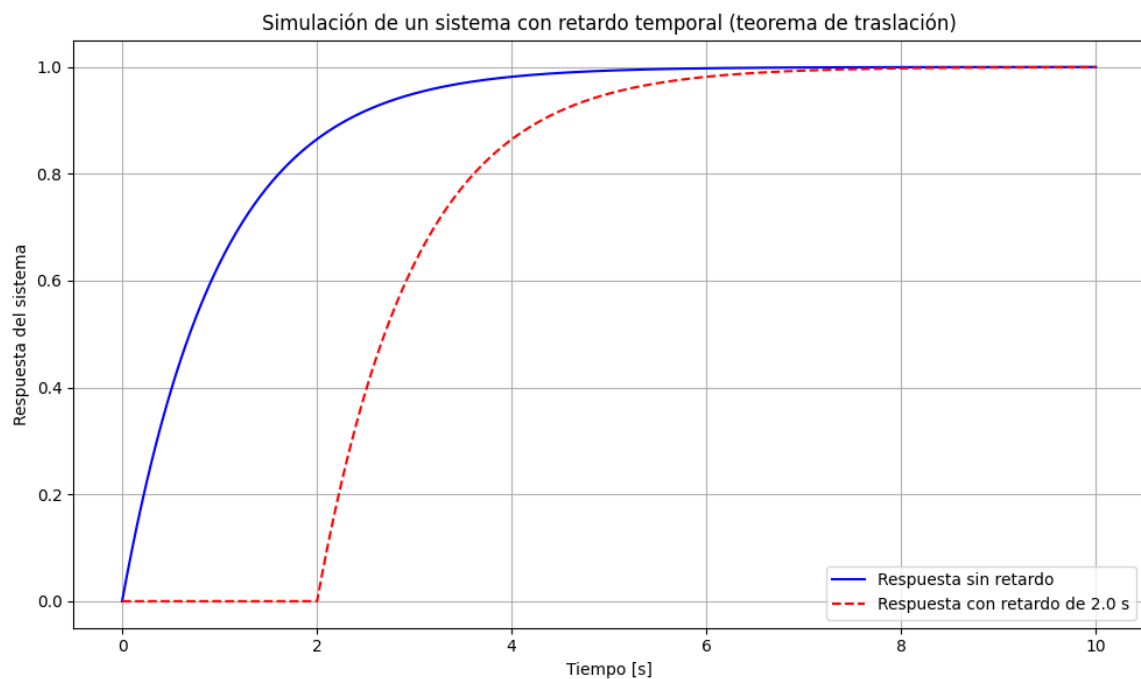


Gráfico 7: Simulación de un sistema con retardo temporal modelado con el teorema de traslación. Creación propia.

En conclusión, los teoremas de traslación en el contexto de la transformada de Laplace ofrecen una herramienta esencial para analizar sistemas que presentan desplazamientos temporales (retardos) o modificaciones en el dominio de frecuencia. El teorema de traslación en el tiempo, por ejemplo, permite modelar efectos de retardo o activación diferida en señales y respuestas de sistemas dinámicos, lo cual es clave en el diseño de controladores en tiempo real y en la implementación de protocolos de comunicación con latencias. Por su parte, el teorema de traslación en frecuencia es utilizado para entender cómo las transformaciones afectan el comportamiento de sistemas en el espectro complejo, útil en análisis espectral y modulación de señales.

Desde el punto de vista computacional, estos teoremas permiten simplificar la estructura algebraica de funciones transformadas, facilitando su manipulación simbólica y numérica en entornos como MATLAB, Python (con SymPy o SciPy), y otras herramientas de simulación. Además, su aplicación se extiende a simulaciones de sistemas físicos que involucren retardos (como robots, redes neuronales diferenciales, o procesos industriales), así como al procesamiento digital de señales, donde el modelado preciso de desplazamientos y filtros depende directamente de estas propiedades.

En síntesis, los teoremas de traslación no solo enriquecen el tratamiento teórico de la transformada de Laplace, sino que habilitan su uso eficiente y versátil en aplicaciones reales que requieren un análisis riguroso del comportamiento de sistemas ante desplazamientos temporales o transformaciones estructurales.

Enlace 2:

<https://www.youtube.com/watch?v=HDRJca79Aag&list=PLHwXkLexR9MDJewziHzhHcEKUg4VRwUFq>

Este video muestra cómo implementar series de potencias en Python utilizando SymPy, con ejemplos aplicados a EDO.

Descripción del enlace: Este tutorial práctico te ayudará a automatizar cálculos de series de potencias en entornos computacionales.

Definición de los términos citados en la clase 12

- **Transformada de Laplace:** Una integral que convierte una función $(f(t))$ del dominio del tiempo en una función $(F(s))$ en el dominio de la frecuencia, definida como $(\mathcal{L}f(t) = \int_0^{\infty} e^{-st} f(t), dt)$, utilizada para simplificar ecuaciones diferenciales.
- **Transformada inversa:** El proceso de recuperar la función original $(f(t))$ a partir de su transformada $(F(s))$, mediante tablas, fracciones parciales o métodos numéricos, esencial para interpretar soluciones en el dominio del tiempo.

Referencias

- Boyce, W. E., & DiPrima, R. C. (2017). *Elementary Differential Equations and Boundary Value Problems* (11th ed.). Wiley.
- Coddington, E. A. (1961). *An Introduction to Ordinary Differential Equations*. Prentice-Hall.
- Zill, D. G. (2018). *A First Course in Differential Equations with Modeling Applications* (11th ed.). Cengage Learning.
- Bender, C. M., & Orszag, S. A. (1999). *Advanced Mathematical Methods for Scientists and Engineers*. Springer.

- Olver, F. W. J. (1997). *Asymptotics and Special Functions*. AK Peters.

Recursos de Profundización

1. Power Series Solutions of Differential Equations

Autor: Carlos Mayorga

Descripción: Este artículo de explica detalladamente cómo resolver ecuaciones diferenciales usando series de potencias, con ejemplos prácticos y aplicaciones en problemas computacionales.

Enlace:

<https://drive.google.com/file/d/1zzgIOSWeEdBX2zep8nZOonceX0WR3CpT/view?usp=sharing>

2. Taylor Series and Numerical Analysis

Autor: Carlos Mayorga

Descripción: Este artículo explora cómo las series de Taylor se utilizan en análisis numérico, con un enfoque en su implementación en algoritmos computacionales.

Enlace:

<https://drive.google.com/file/d/1Aek-idbluTOVeGabGTn48zSrcH7oruMl/view?usp=sharing>

3. Using SymPy to Solve Differential Equations

Autor: Carlos Mayorga

Descripción: Este artículo de la documentación oficial de SymPy detalla cómo utilizar esta biblioteca para resolver ecuaciones diferenciales con series de potencias y Taylor, con ejemplos de código prácticos.

Enlace:

<https://drive.google.com/file/d/1EqACC-ZW3Nm64-OxPAoxLhDIOKn8cGcF/view?usp=sharing>



La excelencia no se improvisa

síguenos

