

CLASE

8

Métodos Numéricos

Ecuaciones Diferenciales
Ordinarias II

Educación de calidad

INTRO DUC CIÓN DE LA CLASE

GRADO / POSGRADO / TECNOLOGÍAS

Estudia a tu tiempo
son interrupciones



En esta sesión se profundiza en los métodos numéricos de mayor orden para resolver ecuaciones diferenciales ordinarias, destacando la necesidad de utilizar esquemas más precisos que el método de Euler cuando se busca una mejor aproximación y control del error, especialmente en problemas donde la acumulación de imprecisiones afecta la estabilidad del modelo, introduciendo métodos como Runge-Kutta de segundo y cuarto orden que mejoran la precisión mediante evaluaciones intermedias de la función, junto con variantes como el método del punto medio y el de Heun, analizando cómo la combinación de pendientes permite aproximaciones más cercanas a la solución real sin reducir excesivamente el tamaño del paso, y realizando una comparación entre estos métodos en términos de orden de convergencia, costo computacional y estabilidad, con el fin de que el estudiante pueda seleccionar el más adecuado según el problema.

Además, se introduce el estudio de las ecuaciones diferenciales parciales como una extensión de las ecuaciones ordinarias cuando intervienen varias variables independientes, comunes en fenómenos como la transferencia de calor o la propagación de ondas, analizando su formulación mediante derivadas parciales y su clasificación en ecuaciones elípticas, parabólicas e hiperbólicas, lo que permite comprender su comportamiento y condiciones de solución, para luego abordar métodos numéricos básicos basados en diferencias finitas que aproximan las derivadas mediante expresiones algebraicas, considerando criterios como consistencia, estabilidad y convergencia para asegurar resultados confiables, y finalmente aplicar estos conceptos en problemas representativos de ingeniería y ciencias, integrando los conocimientos del curso y fortaleciendo la capacidad del estudiante para modelar y resolver problemas complejos de forma estructurada.

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

- 1) Analizar problemas matemáticos y computacionales utilizando métodos numéricos adecuados para su solución aproximada.
- 2) Aplicar técnicas numéricas en la resolución de problemas propios de la ingeniería y las ciencias computacionales.
- 3) Evaluar resultados numéricos considerando el error, la estabilidad y la confiabilidad de los métodos empleados.

Reto # (3)

Ecuaciones Diferenciales Ordinarias II

En la Clase 14 vimos métodos de un paso básicos (Euler y Euler mejorado). Ahora avanzamos a métodos de mayor orden, en particular la familia Runge-Kutta, que ofrecen precisión significativamente superior sin aumentar drásticamente el número de evaluaciones de la función $f(t, y)$.

Los métodos de mayor orden para la resolución de ecuaciones diferenciales ordinarias (EDO) se desarrollaron para mejorar la precisión de las soluciones aproximadas, reduciendo los errores inherentes a los métodos de primer orden, como el método de Euler. Estos métodos son fundamentales cuando se requiere una mayor exactitud sin necesidad de reducir excesivamente el tamaño de los pasos de integración, lo cual podría ser computacionalmente costoso.

El método de Runge-Kutta de segundo orden, también conocido como el método de Heun o el método de "promedio de pendientes", es una mejora del método de Euler. En este método, se realiza una predicción de la solución utilizando el método de Euler y luego se corrige esta predicción calculando una segunda pendiente, que es una estimación de la pendiente en el punto medio del intervalo. Esta corrección mejora la precisión del método de Euler al usar una aproximación más precisa para la pendiente en cada paso, y el error en este método disminuye de manera cuadrática con el tamaño del paso, es decir, el error es proporcional a (h^2) , lo que lo convierte en un método de segundo orden. El método de Runge-Kutta de segundo orden es útil cuando se necesita una mejora en la precisión con respecto a los métodos más simples sin un aumento significativo en los cálculos.

Por otro lado, el método de Runge-Kutta de cuarto orden es uno de los métodos más populares y efectivos para resolver EDOs numéricamente debido a su excelente precisión y eficiencia. Este método utiliza un proceso más complejo, calculando cuatro evaluaciones de la función en cada paso para obtener una mejor aproximación de la derivada en el intervalo. A partir de estas evaluaciones, el método de cuarto orden calcula una combinación ponderada

de las pendientes en diferentes puntos dentro del intervalo, lo que permite obtener una aproximación más precisa de la solución. El error en el método de Runge-Kutta de cuarto orden disminuye de manera proporcional a (h^4) , lo que lo convierte en un método de cuarto orden y, por lo tanto, mucho más preciso que los métodos de menor orden como Euler o Heun, incluso con tamaños de paso relativamente grandes.

Métodos de mayor orden

Los métodos Runge-Kutta (RK) son métodos de un paso explícitos de la forma general:

$$y_{k+1} = y_k + h \sum_{i=1}^s b_i k_i$$

donde los k_i son etapas intermedias:

$$k_i = f\left(t_k + c_i h, y_k + h \sum_{j=1}^{i-1} a_{ij} k_j\right), \quad i = 1, \dots, s$$

Los coeficientes se organizan en el butcher tableau:

$c \mid A$

$\mid b^T$

El orden de precisión p se determina por las condiciones de orden de Butcher (matching de los términos de Taylor hasta h^p).

15.1.1 Método de Runge–Kutta de segundo orden

Una familia de métodos RK de orden 2 tiene $s = 2$ etapas. El más conocido es el método de Heun o RK2 clásico (que coincide con Euler mejorado en su forma predictor-corrector):

$0 \mid 0 \ 0$

$1 \mid 1 \ 0$

$\mid 1/2 \ 1/2$

$$\Rightarrow k_1 = f(t_k, y_k) \quad k_2 = f(t_k + h, y_k + h k_1) \quad y_{k+1} = y_k + \frac{h}{2} (k_1 + k_2)$$

Análisis de error: Error local de truncamiento LTE = $O(h^3)$ Error global GTE = $O(h^2)$ (orden de precisión 2).

Teorema: Si f es Lipschitz en y y suficientemente suave (C^3), el método converge con orden 2.

SECOND-ORDER RUNGE-KUTTKA (HEUN'S) METHOD

Visualizing Order of Accuracy for ODE Solvers

VISUALIZING CONCEPT

Butcher Tableau (Heun's

Method

	c	a _j	b _{ij}
0	0		
1	1	1	
1/2	1	2	
1/2	1/2		1/2

FORMULA & EXPLANATION

- $k_1 = f(t_n, y_n)$
 $k_2 = f(t_n + h, y_n + h \cdot k_1)$
- Predictor (k_1): Initial slope
Slope at predicted point
- Final Average = $\frac{h}{2}(k_1 + k_2)$
- Order of Accuracy: $O(h^2)$

KEY INSIGHTS

- ✓ Higher accuracy than Euler
- ✓ Two function evaluations/step
- ✓ Two function evaluations/step accuracy
- ✓ Explicit method
- ✓ Global Error $\sim O(h^2)$



KEY CONCLUSION: The log-log error plot shows Runge-Kutta, less enhanced Euler on a log-log plot. Runge-Kutta, less enhanced Euler by averaging two slopes per step, significantly increases accuracy to $O(h^2)$ without increasing computational cost. It is with computational cost.

Figura 1: Butcher tableau y etapas del método Runge-Kutta de segundo orden (Heun).

Creación propia.

El método Heun mejora el método de Euler al tomar en cuenta la pendiente en dos puntos (el punto actual y el punto predicho), lo que lo hace más preciso. Este método tiene un orden de precisión de $O(h^2)$, lo que significa que el error disminuye más rápidamente con h en comparación con el método de Euler, que tiene un error de $O(h)$.

ACCURACY COMPARISON: EULER VS ENHANCED EULER (RK2)

Visualizing Convergence Rates of ODE Solvers

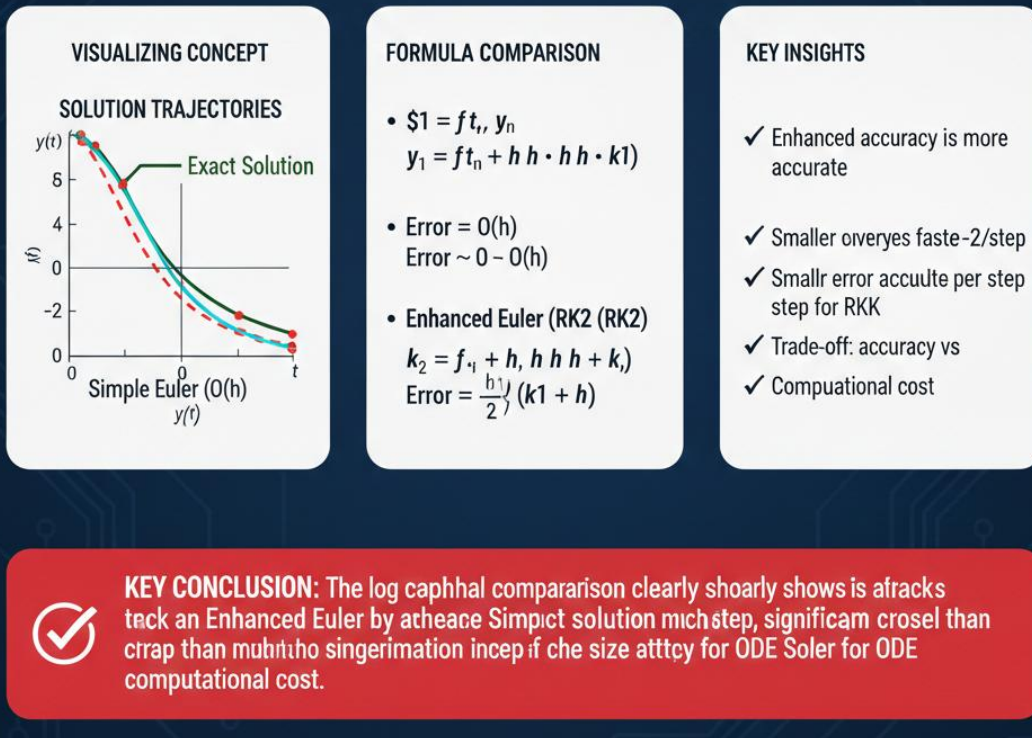


Figura 2: Comparación gráfica: Euler ($O(h)$), Euler mejorado/RK2 ($O(h^2)$) vs solución exacta.

Creación propia.

El método de Euler es un método de primer orden, lo que significa que su error de truncamiento es proporcional a h , es decir, $O(h)$. Este método aproxima la solución mediante una recta tangente a la curva en cada punto, lo que generalmente resulta en una aproximación bastante burda. En la gráfica, la curva obtenida con el método de Euler se aleja rápidamente de la solución exacta a medida que aumenta el paso h . Cuanto mayor es el valor de h , mayor es la desviación del método respecto a la solución exacta.

Método de Runge-Kutta de cuarto orden

El método clásico de Runge-Kutta de orden 4 (RK4) es el más usado de la familia ($s = 4$ etapas).

0	0	0	0	0
1/2	1/2	0	0	0
1/2	0	1/2	0	0
1	0	0	1	0
	1/6	1/3	1/3	1/6

$$\Rightarrow k_1 = f(t_k, y_k) \quad k_2 = f(t_k + h/2, y_k + (h/2)k_1) \quad k_3 = f(t_k + h/2, y_k + (h/2)k_2) \quad k_4 = f(t_k + h, y_k + hk_3) \\ y_{k+1} = y_k + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$

Análisis de error: Error local LTE = $O(h^5)$ Error global GTE = $O(h^4)$ (orden de precisión 4).

Teorema: Si $f \in C^5$ en el dominio, RK4 converge con orden 4. Es el método de orden más alto con solo 4 evaluaciones por paso.

Enlace 1: <https://www.youtube.com/watch?v=6h-KIPNS5BM> Ver en YouTube: Método de Runge-Kutta de segundo orden – Explicación, derivación y ejemplo completo

CLASSICAL FOURTH-ORDER RUNGE-KUTTA (RK4) METHOD

The Industry Standard for ODE Solvers

VISUALIZING CONCEPT

Butcher Tabelau (RK4 Method)

o	o	n	n
0	2	12	1
1	12	12	1
0	0	0	1
16	0	0	14
16	26	28	14

Final Weights

FORMULA & EXPLARISION

- $k_1 = f(t_n, y_n)$
 $k_1 = f(t_1 + h, h \cdot k_1)$
- $k_2 = f(t_1 + \frac{1}{2}h, y_n + \frac{1}{2}h \cdot k_1)$
Error = $0 + 2(h) \cdot h \cdot k_3$
- Final Update Accuracy: $O(h^4)$
 $y_{n+1} = y_n + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4)$

KEY INSIGHTS

- ✓ Very high accuracy (Order 4)
- ✓ Four function evaluations/step
- ✓ Balances accuracy and cost
- ✓ Explicit method ($\sim O(h^4)$)
- ✓ Explicit method

KEY CONCLUSION: The RK4 method achieves a high off accuracy by averaging by sareng four Elceek Sulpes acstar slopes for across slope arengte fowe pronicands, ctapeunad, Oup presicon, and the most widely usecl used ODE Solver for slves for smooth functions.

Figura 3: Butcher tableau clásico de Runge-Kutta de orden 4.

Creación propia.

El método Runge-Kutta de cuarto orden (RK4) es uno de los más precisos para resolver ecuaciones diferenciales de primer orden, ya que tiene un error de truncamiento de orden $O(h^5)$, lo que significa que su precisión mejora rápidamente con un pequeño aumento en el número de pasos n . Es ampliamente utilizado debido a su eficiencia y exactitud, ya que utiliza

varias evaluaciones de la función por paso para obtener una solución precisa sin ser excesivamente costoso en términos de cálculos.

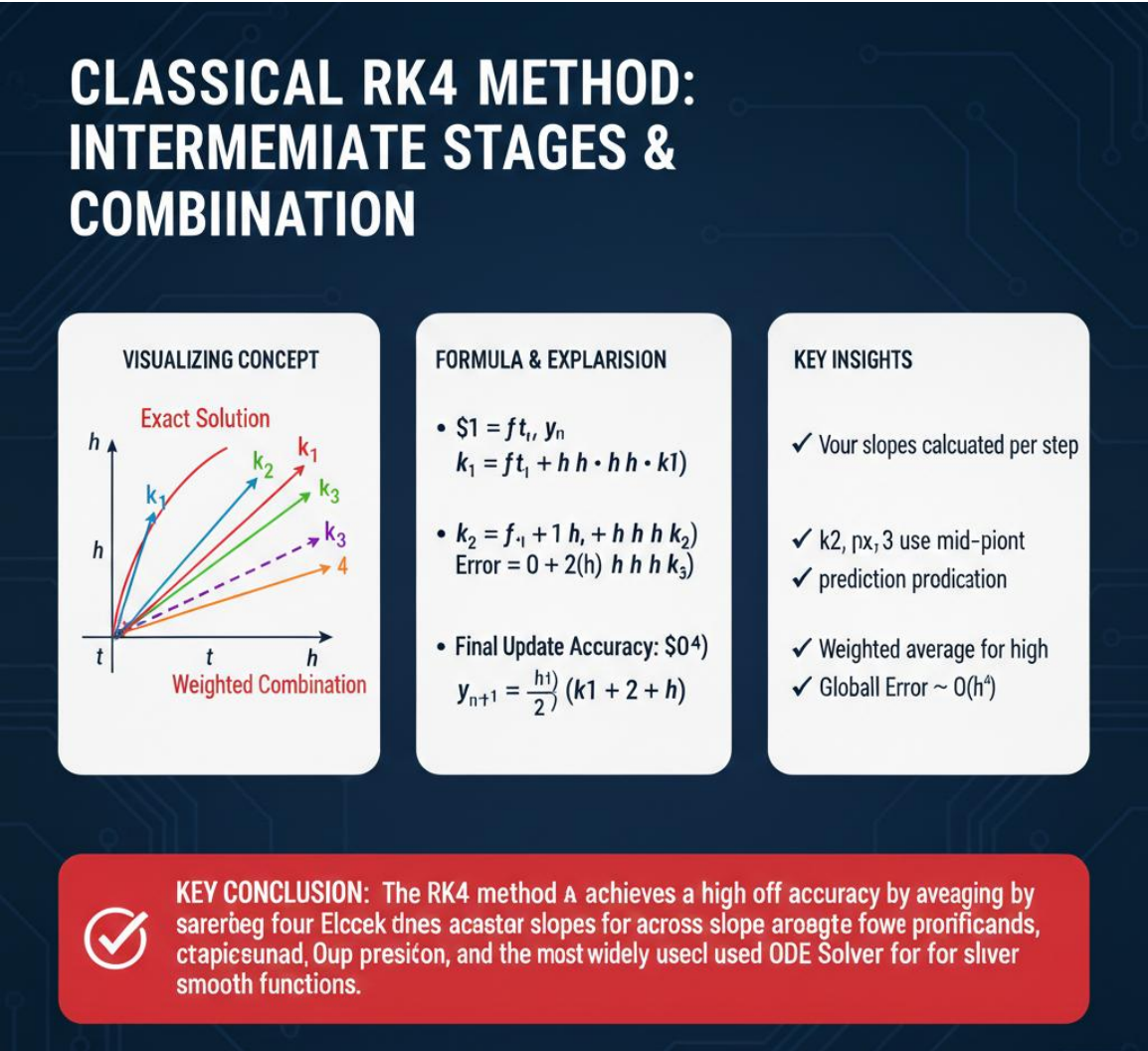


Figura 4: Etapas intermedias k_1, k_2, k_3, k_4 en RK4 y combinación ponderada.

Creación propia.

Comparación de métodos para EDO

Método	Número de evaluaciones por paso	Orden global	Error global	Estabilidad absoluta (región)	Ventajas matemáticas	Desventajas	Aplicación típica en ciberseguridad
Euler	1	1	$O(h)$	Disco de radio 1 centrado en -1	Muy simple, bajo costo	Baja precisión, inestable para rígidas	Simulaciones rápidas y prototipos de propagación
Euler mejorado / RK2	2	2	$O(h^2)$	Mayor que Euler	Buen balance costo/precisión	Aún limitado para	Modelos SIR/SEIR con tasas moderadas

						problemas rígidos	
Runge-Kutta 4 (clásico)	4	4	$O(h^4)$	Mucho mayor que RK2	Alta precisión con pocas evaluaciones	Costo 4× mayor que Euler	Simulaciones precisas de dinámica de amenazas

Tabla 1: En escala log-log (error global vs h), pendiente -1 para Euler, -2 para RK2, -4 para RK4 (convergencia mucho más rápida con RK4).

Creación propia.

Elección:

- Euler para prototipos rápidos o cuando h ya es pequeño.
- RK2 para balance costo/precisión.
- RK4 para máxima precisión con razonable costo computacional.

Introducción a Ecuaciones Diferenciales Parciales y Aplicaciones

Esta clase será más extensa y reflexiva, ya que sirve como cierre y proyección hacia aplicaciones reales. Exploraremos el concepto fundamental, los métodos numéricos básicos más utilizados, aplicaciones relevantes en ingeniería y ciencias (incluyendo ciberseguridad), y finalmente una integración de todos los contenidos vistos, con recomendaciones para tu proyección académica y profesional.

Las ecuaciones diferenciales parciales (EDP) son un tipo de ecuación que involucra derivadas parciales de una o más funciones desconocidas con respecto a varias variables independientes. A diferencia de las ecuaciones diferenciales ordinarias (EDO), que involucran derivadas respecto a una sola variable independiente, las EDP modelan fenómenos más complejos en los cuales las soluciones dependen de múltiples variables.

Para resolver EDPs, especialmente en casos donde no es posible obtener una solución exacta, se recurren a métodos numéricos. Los métodos de diferencias finitas para EDP son una de las técnicas más utilizadas. Estos métodos consisten en aproximar las derivadas parciales de una función utilizando diferencias entre valores de la función en una malla de puntos discretos.

El método de diferencias finitas convierte las ecuaciones diferenciales parciales en un sistema de ecuaciones algebraicas que es más fácil de resolver numéricamente. Este método es sencillo de implementar y adecuado para problemas con geometrías regulares.

El método de elementos finitos, por ejemplo, divide el dominio en "elementos" más simples y resuelve la ecuación de forma aproximada en cada uno de ellos, lo que le otorga gran flexibilidad en la resolución de problemas de ingeniería con geometrías irregulares.

Las ecuaciones diferenciales parciales tienen una amplia variedad de aplicaciones en la ingeniería y las ciencias, donde modelan fenómenos dinámicos y estáticos en sistemas multidimensionales.

Las soluciones numéricas a las EDP, obtenidas a través de métodos como las diferencias finitas, permiten a los ingenieros y científicos obtener predicciones útiles que de otra manera serían difíciles de obtener.

Concepto de ecuaciones diferenciales parciales

Una ecuación diferencial parcial (EDP) es una ecuación que relaciona una función desconocida $u(x_1, x_2, \dots, x_d)$ de varias variables independientes con sus derivadas parciales respecto a esas variables.

Formulación matemática general:

$$F\left(x, u, \frac{\partial u}{\partial x^1}, \frac{\partial u}{\partial x^2}, \dots, \frac{\partial^k u}{\partial x^{1k_1} \dots \partial x_d^{k_d}}, \dots\right) = 0$$

donde F es una función dada, y la ecuación involucra derivadas parciales de orden hasta k .

Clasificación principal según el tipo (para EDP lineales de segundo orden en dos variables):

Sea la EDP general de segundo orden:

$$A\frac{\partial^2 u}{\partial x^2} + B\frac{\partial^2 u}{\partial x \partial y} + C\frac{\partial^2 u}{\partial y^2} + \text{términos de orden menor} = g(x, y)$$

El discriminante es $\Delta = B^2 - 4AC$. Entonces:

- **Hiperbólica** si $\Delta > 0$ (e.g., ecuación de onda: $\frac{\partial^2 u}{\partial t^2} = c^2 \frac{\partial^2 u}{\partial x^2}$, $\Delta = 4c^2 > 0$)
- **Parabólica** si $\Delta = 0$ (e.g., ecuación del calor: $\frac{\partial u}{\partial t} = \kappa \frac{\partial^2 u}{\partial x^2}$, $\Delta = 0$)
- **Elíptica** si $\Delta < 0$ (e.g., ecuación de Laplace/Poisson: $\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = f(x, y)$, $\Delta = -4 < 0$)

Condiciones de contorno y/o iniciales son esenciales para unicidad y existencia:

- Dirichlet: u especificado en la frontera
- Neumann: derivada normal especificada
- Robin (mixta): combinación lineal
- Iniciales (para evolutivas): $u(t=0)$ y posiblemente $\frac{\partial u}{\partial t}(t=0)$

Teorema de existencia y unicidad: depende del tipo y condiciones (e.g., problema bien planteado de Hadamard: existencia, unicidad y dependencia continua de datos iniciales/contorno).

Métodos numéricos básicos para EDP

Los métodos numéricos para EDP discretizan el dominio espacial y temporal (si es evolutiva) y transforman la ecuación continua en un sistema algebraico grande.

Métodos principales:

1. Diferencias finitas (Finite Differences)

- Discretiza derivadas parciales mediante diferencias finitas (forward, backward, centradas).
- Para ecuación del calor 1D: $\partial u / \partial t = \kappa \partial^2 u / \partial x^2 \rightarrow (u_i^{n+1} - u_i^n) / \Delta t = \kappa (u_{i-1}^n - 2u_i^n + u_{i+1}^n) / (\Delta x)^2$ (explícito)
- Esquemas implícitos (Crank-Nicolson) más estables.
- Orden: $O(\Delta t + \Delta x^2)$ típico; Crank-Nicolson $O(\Delta t^2 + \Delta x^2)$.

2. Elementos finitos (Finite Element Method – FEM)

- Divide el dominio en elementos (triángulos, tetraedros, etc.).
- Aproxima u en cada elemento por polinomios locales (base de funciones de forma).
- Lleva a sistema $Ax = b$, donde A es matriz de rigidez (dispersa).
- Ventaja: maneja geometrías complejas y condiciones de contorno naturales.

3. Volúmenes finitos (Finite Volume Method – FVM)

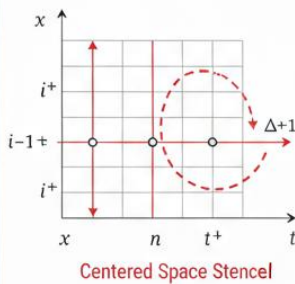
- Conserva cantidades físicas (flujo, masa) en celdas.
- Ideal para problemas de conservación (e.g., ecuaciones de Navier-Stokes).

4. Métodos espectrales y pseudospectrales

- Usan bases globales (Chebyshev, Fourier) para alta precisión en dominios simples.

1D HEAT EQUATION: FINITE DIFFERENCE SCHEME

VISUALIZING CONCEPT



FORMULA & EXPLANATION

- Heat Equation (1D): $u_t = \alpha_{xx}$
Wave Eq: $u_t = \alpha^2 u_{xx}$
- Time Deriv. (Forward Centred) $\Delta t < \Delta t = u_t$
- Space Deriv: $-2u_t^1 + (2u_t^1 + u_{xx})$
- Explicit Scheme: $= u_t$
Uses current time: n to predict $n+1$

KEY INSIGHTS

- ✓ Approximates PDE with linear equations.
- ✓ Explicit scheme to implement.
- ✓ Stability condition: $\alpha \Delta t / 2 \Delta x^2 \leq 1/2$
- ✓ First-order accurate in time, second-order in space (FTCS).



KEY CONCLUSION: The Explicit Finite Difference scheme for the 1D Heat Equation is forward for stability with fineness of space computed for strict stability but is unstable in advection and solving ODEs.

Figura 5: Esquema de diferencias finitas para ecuación del calor 1D: stencil centrado en espacio y forward en tiempo.

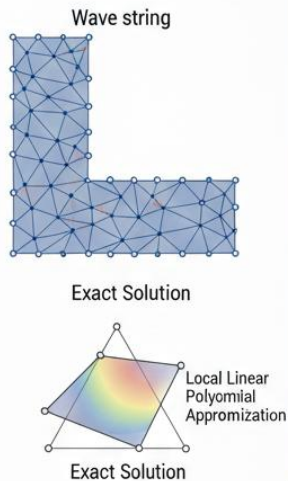
Creación propia.

Enlace 2: <https://www.youtube.com/watch?v=6iS3kM2f6zU> Ver en YouTube: Métodos numéricos para EDP – Diferencias finitas y elementos finitos (introducción) Este recurso muestra esquemas básicos de diferencias finitas para calor y onda, y menciona FEM.

FINITE ELEMENT MESH: TRIANGULAR TRIANGULAR APPROXIMATION

Local Linear Polyomolals for PDE Solutions

VISUALIZING CONCEPT



FORMULA & EXPLAIATION

- Approxitates PDE on comall complex domains)
 - Solution is approx. by simple fct simple fct on each element
 - Local approx: $u(xyy)$
 - Global solution $-c_1x + c_2y + c_3$
 - Global solution $\approx$ continuous
- Appromiationun

KEY INSIGHTS

- ✓ Versatile for complex geomteries.
- ✓ Basis fcts shape fcts)
- ✓ Good for iraguulr domare solutions.
- ✓ Physics dictaer domains.



KEY CONCLUSION: Firite Edment Method Meei vels fcsten pte timeash ofie chteel uss sorward tor B liutas with witureate nls lncaly ardon polyomiss is aotias for nout obu quie tie solution soolurs solution, spowerfuly pxtetfroum solving ODEs.

Figura 6: Malla de elementos finitos triangular: aproximación local por polinomios lineales.

Creación propia.

La figura muestra una malla de elementos finitos triangulares, que es un método de aproximación numérica utilizado para resolver ecuaciones diferenciales parciales (EDP) en geometrías complejas. En esta malla, se divide el dominio de la solución en pequeños triángulos, y en cada uno de esos elementos, se aproximan las soluciones utilizando polinomios lineales.

Aplicaciones en ingeniería y ciencias

Las EDP son el lenguaje matemático de la física y la ingeniería:

- **Ecuación del calor:** difusión térmica, difusión de contaminantes, propagación de información en redes (modelos de difusión de datos o malware).
- **Ecuación de onda:** propagación de señales electromagnéticas, acústica, ondas sísmicas, propagación de ataques DDoS en espacio-tiempo.
- **Ecuación de Laplace/Poisson:** potenciales electrostáticos, flujo de calor estacionario, análisis de balance de carga en servidores seguros, optimización de rutas en redes.

- **Ecuaciones de Navier-Stokes:** dinámica de fluidos (simulación de ataques por inyección de tráfico).
- **Ecuaciones de Schrödinger:** criptografía cuántica y post-cuántica (simulación de sistemas cuánticos).
- **Ecuaciones de reacción-difusión:** propagación de epidemias digitales (malware, phishing), modelos SIR/SEIR adaptados a redes.

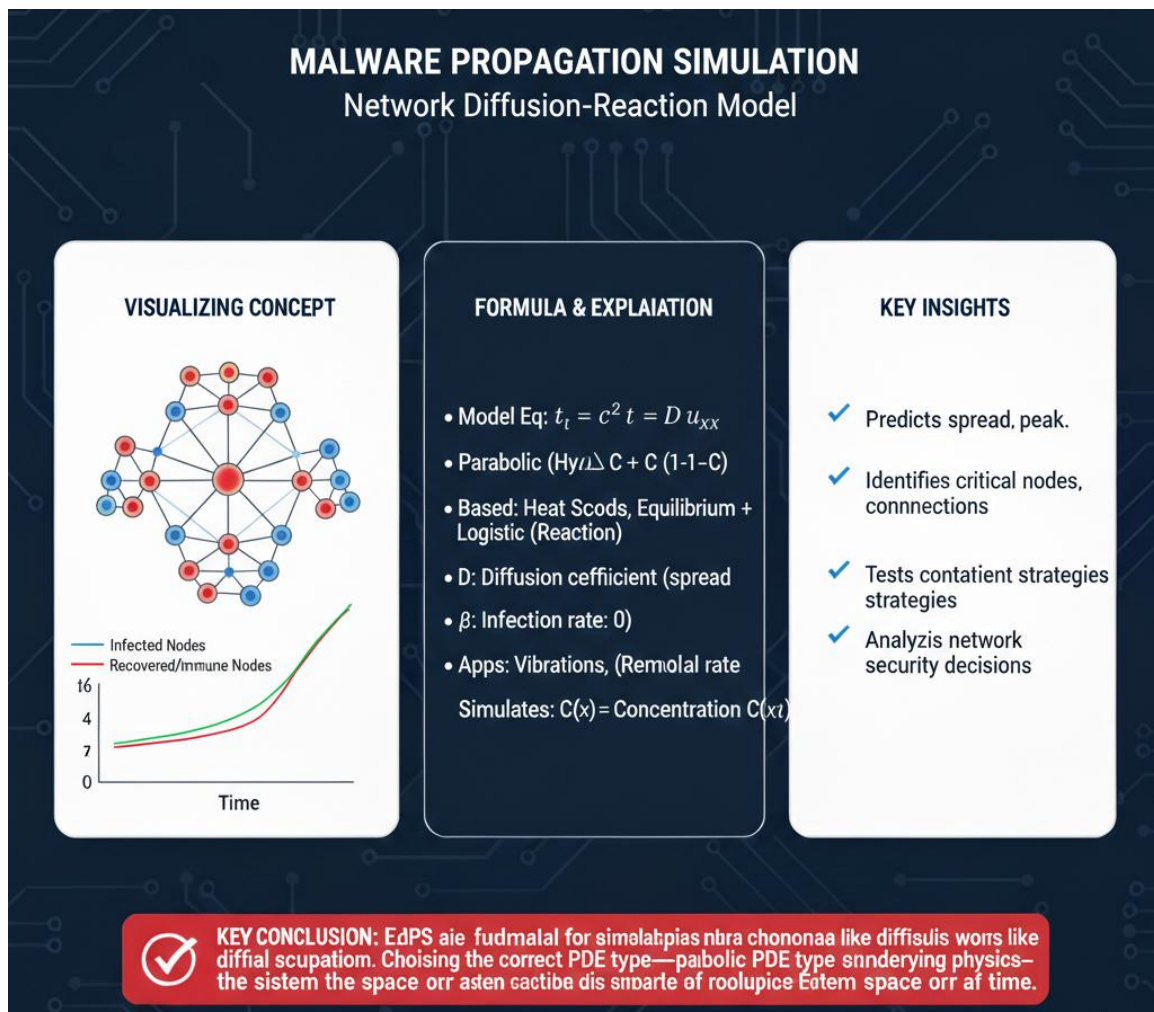


Figura 7: Simulación de propagación de malware en red usando ecuación de difusión-reacción (modelo adaptado).

Creación propia.

Este tipo de ecuaciones es común en modelos dinámicos que describen la propagación de sustancias, señales o incluso enfermedades, y puede adaptarse para simular la expansión de malware a través de una red de computadoras o dispositivos interconectados.

El modelo se basa en una ecuación de difusión-reacción, que es una combinación de dos términos fundamentales: uno que describe cómo el malware se propaga (difusión) y otro que describe cómo el malware interactúa con el sistema o se elimina (reacción).

Integración de contenidos y proyección académica

Esta asignatura ha cubierto un espectro completo de métodos numéricos aplicados a computación y ciberseguridad:

- a) Errores y precisión → base para toda implementación segura.
- b) Ecuaciones no lineales → criptoanálisis (Coppersmith, lattices).
- c) Sistemas lineales → resolución en lattices y regresión side-channel.
- d) Interpolación y aproximación → reconstrucción de funciones de leakage.
- e) Integración numérica → correlación en DPA/CPA, Monte Carlo para riesgos.
- f) EDO → dinámica temporal de amenazas y propagación.
- g) Introducción a EDP → propagación espacial-temporal, side-channel EM, modelado avanzado.

Proyección académica y profesional:

- Continúa con cursos avanzados: EDP numéricas (FEM, FVM), métodos espectrales, EDP no lineales/rígidas.
- Herramientas recomendadas: Python (SciPy, FEniCS, Dedalus), MATLAB, Julia (DifferentialEquations.jl).
- Proyectos sugeridos:
 - Simulador de propagación de malware en red usando ecuación de difusión-reacción.
 - Implementación de DPA con integración numérica de correlación.
 - Modelado de side-channel EM usando ecuación de onda.
- Certificaciones complementarias: OSCP, CEH, o cursos especializados en criptografía numérica y ML para seguridad.

RE FE REN CIAS

Glosario:

- Creswell, J. W., & Creswell, J. D. (2018). Research Design: Qualitative, Quantitative, and Mixed Methods Approaches (5th ed.). SAGE Publications.
- Booth, W. C., Colomb, G. G., & Williams, J. M. (2008). The Craft of Research (3rd ed.). University of Chicago Press.
- Montgomery, D. C., & Runger, G. C. (2018). Applied Statistics and Probability for Engineers (7th ed.). Wiley.
- Knuth, D. E. (1984). Literate Programming. The Computer Journal, 27(2), 97–111.
- Kelleher, J. D., & Tierney, B. (2018). Data Science. MIT Press.



síguenos en:



www.pucevirtual.puce.edu.ec